Towards a reference architecture for integrated knowledge networks

by

Johannes Hendrik Gous

Dissertation presented for the degree of Doctor of Philosophy
in the Faculty of Engineering at Stellenbosch University

Prof C.S.L. Schutte

Department of Industrial Engineering, Stellenbosch University

Prof A. Gerber

Centre for Artificial Intelligence Research, Meraka CSIR

December 2014

DECLARATION

By submitting this dissertation electronically, I declare that the entirety of the work contained therein is my own, original work, that I am the sole author thereof (save to the extent explicitly otherwise stated), that reproduction and publication thereof by Stellenbosch University will not infringe any third party rights and that I have not previously in its entirety or in part submitted it for obtaining any qualification.

Date: ....

Opsomming

Hierdie tesis ondersoek die ontwikkeling van geïntegreerde kennisnetwerke (GK's) aan die hand van 'n verwysingsargitektuur. Die doel van die studie is om 'n beter begrip van die onderliggende ontwerpbeginsels van hierdie klas kollaborasie netwerke te kry. Alhoewel beide GK's en ondernemingsingenieurswese beskou word as belowende benaderings tot die uitdagings van die Inligtingsera, bestaan beduidende uitdagings steeds in die ontwikkeling van GK's.

Ons geglobaliseerde, gekommersialiseerde samelewing word tans gekenmerk deur beide buitengewone kompleksiteit en buitengewone tempo-verandering. Ondernemingsingenieurswese word beskou as 'n belowende benadering om ondernemings toe te rus met die eienskappe wat in aanvraag is in die moderne ekonomie, insluitend aanpasbaarheid en vlugheid. Ondernemingsargitektuur dra by tot hierdie poging deur 'n hoëvlak ontwerp van die onderneming te voorsien wat geïntegreerde ontwikkeling van die onderneming toelaat.

Vanuit 'n kommersiële oogpunt word dit ruim aanvaar dat die vermoë om te innoveer en nuwe kennis te ontwikkel deur die ontwikkeling van nuwe produkte, dienste en prosesse 'n kernfaktor in die oorlewing van ondernemings is. Die jongste benadering in innovasiebestuur toon aan dat die innovasieproses nie meer slegs in een onderneming uitgevoer word nie. Dit lei dan tot interorganisasie-innovasienetwerke. Die belangrikheid van kennis as 'n dinamiese instaatsteller van hierdie netwerk-innovasiebenadering word verder beklemtoon. Dit het gelei tot die ontstaan van GK's waarin kennis tot stand gebring en gedeel word tussen netwerk belanghebbendes om sodoende volhoubare innovasie te bevorder.

Die toenemende tempo in verandering beteken dat ondernemings, insluitende GK's, toenemend minder tyd het om op markveranderings en -geleenthede te reageer. Die klem val daarom op die ondernemingsingenieurswese- en ondernemingsargitektuur-dissiplines as hulpmiddels om by die dinamiese landskap van die Inligtingsera aan te pas. Tans word die vermoë om omvattende ondernemingsingenieurswese in GK's te beoefen, gekniehalter deur die tekort aan hulpbronne wat grondige konstruksie-beginsels vir hierdie netwerke beskryf. Die fokus van hierdie studie is daarom die toepassing van die ondernemingsingenieurswese-dissipline op GK's deur die ontwikkeling van 'n verwysingsargitektuur.

Die verwysingsargitektuur vir GK's word ontwikkel deur ontwerpwetenskapnavorsing binne 'n pragmatiese en kwalitatiewe navorsingstrategie. Die navorsingsprobleem word eers geïdentifiseer en gemotiveer. Verskeie oplossingsdoelwitte word vervolgens bepaal. Hierna geskied die ontwerp en ontwikkeling van die verwysingsargitektuur deur middel van die vier herhalende ontwerpsiklusse. Die verwysingsargitektuur vir GK's word gedemonstreer en geëvalueer deur 'n reeks beeldende scenario's, waarna die bruikbaarheid, nuutheid en ontwerpstrengheid van die artefak gekommunikeer word.

Dit is bevind dat die verwysingsargitektuur konstruksiebeginsels in die ontwikkeling van GK's voorsien en sodoende die ontwerp, werk en navorsing in hierdie klas kollaborasie netwerke moontlik maak. Dié studie neem 'n eerste tree in die rigting om die konsep van ondernemingsingenieurswese tot GK's uit te brei. Dit maak die groter aanpasbaarheid van hierdie netwerke by die dinamiese omgewing van die Inligtingsera moontlik.

Summary

This thesis has as its focus the engineering of integrated knowledge networks (IKNs) through the use of a reference architecture. The purpose of the study is to gain a deeper understanding of the constructional principles underlying this class of collaborative networks. Although IKNs and enterprise engineering are both seen as promising approaches to the challenges of the Information Age, significant challenges still exist in the engineering of IKNs.

Our globalised and commercialised society may currently be characterised by both extreme complexity and extreme rates of change. Enterprise engineering is seen as a promising approach to equip enterprises with the characteristics that are desirable in the modern economy, including flexibility and agility. Enterprise architecture contributes to this endeavour by providing a high-level design of the enterprise that allows for integrated engineering of the enterprise.

From a commercial point of view, it has been widely recognised that the ability to innovate and generate new knowledge through the development of new products, services and processes is a key factor in the survival of enterprises. The latest trends in innovation management, however, show that the innovation process is no longer one that is executed inside a single enterprise, giving rise to the development of inter-organisational innovation networks. Furthermore, the importance of knowledge as a dynamic enabler of this networked innovation approach is highlighted. This has lead to the emergence of IKNs in which knowledge is created and shared between network stakeholders in order to foster sustainable innovation.

The increasing rate of change means that enterprises, including IKNs, have progressively less time to react to market changes and opportunities. The emphasis is therefore on the potential of the EE and EA disciplines as tools to adapt to the dynamic landscape of the Information Age. At present, the ability to apply comprehensive enterprise engineering to IKNs is hampered by the lack of resources that describes sound constructional principles for these networks. The focus of this study is therefore on the application of the enterprise engineering discipline to IKNs through the development of a reference architecture.

The reference architecture for IKNs is developed through design science research within a pragmatic and qualitative research strategy. The research problem is first identified and motivated. Various solution objectives are subsequently defined. This is followed by the design and development of the reference architecture through four iterative design cycles. A qualitative systematic review is conducted and serves as the foundation for the development of various reference models. The reference architecture for IKNs is demonstrated and evaluated through a series of illustrative scenarios, after which the utility, novelty and design rigour of the artefact is communicated.

It was found that the reference architecture provides constructional principles in the engineering of IKNs, thus enabling the design, operation and research of this class of collaborative networks. The study therefore takes a first step toward extending the concept of EE to IKNs, and collaborative networks in general. This enables the greater adaptability of these networks to the dynamic environment of the Information Age.

Acknowledgements

I would like to express my sincere gratitude and appreciation to the following people whose support, assistance and contributions made this study possible:

- To **Prof Corne Schutte**, for your guidance, inputs, collaborative spirit and support throughout this research project. You introduced me to this field nearly 10 years ago, and have been a mentor ever since.

- To **Prof Aurona Gerber**, for your supervision, invaluable support and motivation during this study. Thank you for investing in this thesis from day one and providing me with the opportunity to dedicate my time to this study.

- To the **CSIR Meraka Institute** and **CAIR**, for making this study financially feasible and giving me the opportunity to pursue this dream.

- To the representatives of the **RLabs, Agri and MediaLab/MIH** innovation networks, for volunteering their time and insight during the evaluation of the study.

- To the **personnel** at the **Department of Industrial Engineering** at Stellenbosch University, for your support and encouragement.

- To **HTWG Konstanz** and the **NITIM** network, for broadening my horisons and exposing me to different perspectives on research.

- To **John A. Zachman**, for your inputs and inspiration.

Dan ook:

- Aan my **ouers**, vir jul liefde en vertroue in my oor die laaste 13 jaar. Sonder julle sou hierdie reis vir seker ook nie moontlik gewees het nie.

- Aan my vrou **Denéle**, dankie ook vir jou liefde, ondersteuning, inspirasie, geduld en vertroue. Niemand anders verstaan hierdie paadjie beter as jy nie.

Table of Contents

List of Figures

List of Tables

Abbreviations

ARCON	A Reference Model for Collaborative Networks
CIMOSA	Computer Integrated Manufacturing Open System Architecture
CN	Collaborative Networks
CNO	Collaborative Networked Organisations
DSR	Design Science Research
EA	Enterprise Architecture
ECOLEAD	European Collaborative Networked Organisations Leadership Initiative
ERA	Enterprise Reference Architecture
GERAM	Generalised Enterprise Reference Architecture and Methodology
GRAI-GIM	GRAI Integrated Methodology
ICT	Information and Communication Technology
IKN	Integrated Knowledge Network
IE	Industrial Engineering
IS	Information Systems
KN	Knowledge Networks
PERA	Purdue Enterprise Reference Architecture
R&D	Research and Development
TOGAF	The Open Group Architecture Framework
VERA	Virtual Enterprise Reference Architecture

Definitions

An alphabetical summary of definitions that are either adopted or developed for concepts in this study.

Architecture:

> *An architecture is the fundamental organisation of a system embodied in its components, their relationships to each other, and to the environment, and the principles guiding its design and evolution.* (IEEE STD 1471-2000)

Architecture description:

> *A collection of artefacts (e.g. models and descriptions) that describe the architecture of an entity.*

Architecture instantiation:

> *The manifestation of the architecture of an entity as an instantiated object.*

Architecture framework:

> *An architecture framework is a tool. It should describe a method for designing an information system in terms of a set of building blocks, and for showing how the building blocks fit together. It should contain a set of tools and provide a common vocabulary. It should also include a list of recommended standards and compliant products that can be used to implement the building blocks.* (The Open Group 2011)
>
> *[A] generic architecture that can be used as the starting point to derive an enterprise's architecture* (Giachetti 2010), specifically a generic reference architecture in this study.

Artefact:

> *An object made by a human being, typically one of cultural or historical interest.*
> (Oxford Dictionaries n.d.)

Collaborative Networked Organisation (CNO):

> *A collaborative network possessing some form of organisation in terms of structure of membership, activities, definition of roles of the participants, and following a set of governance principles and rules.* (Camarinha-Matos & Afsarmanesh 2012)

Enterprise architecture (EA):

> *An Enterprise Architecture describes the structure of an enterprise, its decomposition into subsystems, the relationships between the subsystems, the relationships with the external environment, the terminology to use, and the guiding principles for the design and evolution of an enterprise.* (Giachetti 2010)

Enterprise reference architecture (ERA):

> *An enterprise-class reference architecture.*

Framework:

> *[A] structure for supporting or enclosing something else. In the modeling area, a framework can be seen as an "envelope" that might include a number of (partial) models, collections of templates, procedures and methods, rules, and even tools (e.g. modeling languages).* (Camarinha-Matos & Afsarmanesh 2008a)

Innovation:

> *Innovation is not just one simple act. It is not just a new understanding or the discovery of a new phenomenon, not just a flash of creative invention, not just the development of a new product or manufacturing process; nor is it simply the creation of new capital and markets.*
>
> *Rather innovation involves related creative activity in all these areas. It is a connected process in which many and sufficient creative acts, from research through service, are coupled together in an integrated way for a common goal.* (Salvendy 2001)

Integrated Knowledge Network:

> *A Knowledge Network that spans all domains, communities, and trust relationships with the goal of fostering sustainable innovation that will continue to promote the competitiveness of its users.* (Du Preez, Louw & Lutters 2008a)

Knowledge Network:

> *A number of people, resources and relationships among them, who are assembled in order to accumulate and use knowledge primarily by means of knowledge creation and transfer processes, for the purpose of creating value.* (Seufert et al. 1999)

Model:

> *A model is an abstraction of a (real or language-based) system allowing predictions or inferences to be made.* (Kühne 2006)

Reference model:

> *A reference model is a generic abstract representation for understanding the entities and the significant relationships among those entities of some area, and for the derivation of other specific models for particular cases in that area.*

Reference architecture:

> *[A] generic architecture that can be used as the starting point to derive an enterprise's architecture* (Giachetti 2010), specifically a partial reference architecture in this study.

Publication

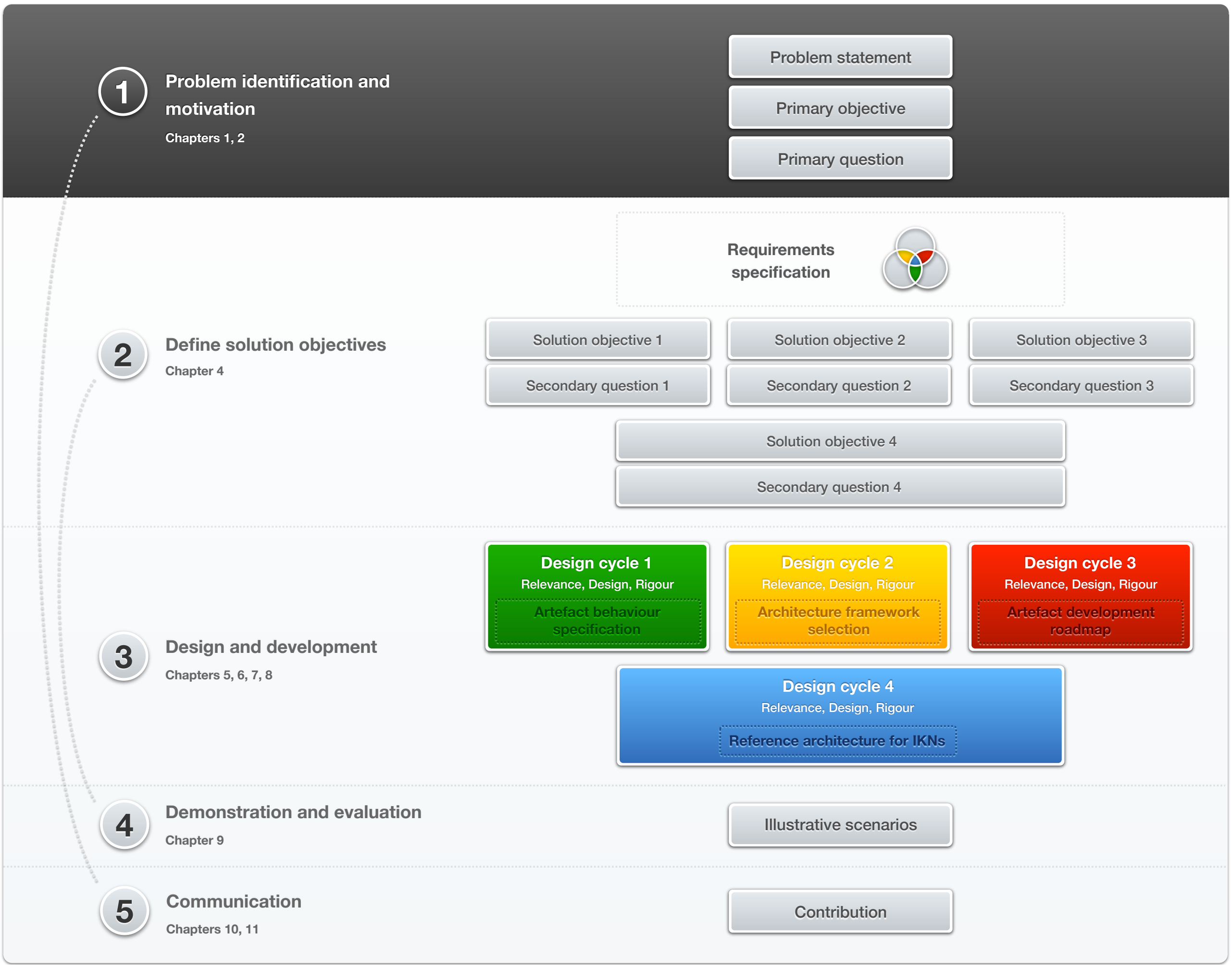
It should be noted that this study already produced a published conference proceeding prior to the final compilation of this thesis:

Gous, H. et al., 2011. Business architecture for inter-organisational innovation networks: A case study comparison from South Africa and Germany. In 17th International Conference on Concurrent Enterprising. Aachen, Germany: IEEE.

Part 1 - Problem identification and motivation

This document consists of five parts, each containing a number of chapters. The structure of these parts is based on the research design that is employed in the study, as depicted in the document layout diagram below. Each part therefore has a specific focus and function in terms of the research design. This research design is presented in chapter 3.

The first part of the document contains two chapters (chapters 1 and 2), and presents the first activity of the research design employed in this study. This activity aims at identifying and motivating the research problem which the study will pursue. Chapter 1 introduces the focus, rationale, objectives and scope of the study, before chapter 2 elaborates on the identified research problem by presenting relevant literature. In Part 2 of the document, solution objectives for an artefact that aims to solve this research problem are defined.



1. Introduction

1.1 Introduction

This thesis has as its focus the engineering of integrated knowledge networks (IKNs) through the use of a reference architecture. The purpose of the study is to gain a deeper understanding of the constructional principles underlying this class of collaborative networks. This multi-disciplinary study falls within the industrial engineering discipline, specifically within enterprise engineering, and contributes a reference architecture for IKNs.

Three phenomena of modern society are embedded in the above focus statement of the thesis, namely 1) enterprise engineering, 2) enterprise architecture and 3) integrated knowledge networks. In order to provide some context for the study, a brief introduction to these phenomena is provided first.

In 1980, Alvin Toffler described the transition of society from the Industrial Age, or “Second Wave”, to the Information age, which he termed the “Third Wave” (Toffler 1990). The Information Age is associated with a shift from the traditional, mechanised industry brought on by the industrial revolution, to an economy based on computerised information. In the about 35 years since the first publication of Toffler’s *The Third Wave*, modern society has continued, and indeed undergone accelerated development, along this trend. Innovations like the internet, mobile devices and ubiquitous computing have fundamentally altered the fabric of the world of the 21st century. At present, our globalised, commercialised society is characterised by:

- extreme complexity, and
- extreme rates of change.

For enterprises attempting to accommodate the extreme complexity of the Information Age, along with the extreme rates of change within this complexity, EE is seen as a promising approach (Liles et al. 1995; Towill 1997; Rouse 2004; Saenz et al. 2009). EE is concerned with the design of the enterprise in order to enable a range of characteristics that are desirable in the modern economy, including flexibility and agility (Giachetti 2010). Enterprises, however, are seldom designed in a single project, with EE more likely being conducted through a range of smaller projects throughout the life cycle of the enterprise. In order for these EE efforts to produce an integrated design, a high-level design of the enterprise is required. EA fulfils this role by specifying an enterprise-wide view of the processes, information and organisation of the enterprise, including how these aspects are integrated (Kappelman 2011).

From a commercial point of view, it has been widely recognised that the ability to innovate and generate new knowledge through the development of new products, services and processes is a key factor in the survival of enterprises in the Information Age and its associated knowledge economy (Krogh et al. 2001;

Drucker 2007). The latest trends in innovation management, however, show that the innovation process is no longer one that is executed inside a single enterprise, giving rise to the development of inter-organisational innovation networks (Powell et al. 1996). The importance of knowledge as a dynamic enabler of this networked innovation approach is furthermore highlighted (Seufert et al. 1999). This has lead to the emergence of IKNs in which knowledge is created and shared between network stakeholders in order to foster sustainable innovation.

The increasing rate of change means that enterprises, including IKNs, have progressively less time to react to market changes and opportunities. Therefore, enterprises need to adapt in order to become custom products, *assembled to order* rather than *made to order* (Kappelman & Zachman 2013). In this approach, independent enterprise components, rather than complete enterprise implementations, are created for storage. Any combination of these enterprise components may then be assembled in a multitude of different ways through late-binding to construct an enterprise that reacts to a change or opportunity in the market. The emphasis is therefore on the potential of the EE and EA disciplines as tools to adapt to the dynamic landscape of the Information Age. The focus of this study is on extending the discipline of enterprise engineering to IKNs through the development of a reference architecture that describes the constructional components of these networks.

Section 1.2 provides additional theoretical background to define the domain of collaborative networks and IKNs, as well as the field and discipline of enterprise engineering. Section 1.3 discusses the rationale for the study by articulating the research problem and the purpose of the study. The research objectives and questions that drive the study are presented in section 1.4, before the research strategy to achieve the research objective is outlined in section 1.5. Section 1.6 discusses the scope and delineations of the study, and section 1.7 provides an overview of the contributions of the study. Section 1.8 concludes this chapter by presenting the structure of this document.

1.2 Background

The purpose of this section is to provide the theoretical background that forms the context for the discussion of the rationale for the study in section 1.3.

1.2.1 Integrated knowledge networks

Varying levels of collaboration of inter-organisational networks have been recorded for many centuries (Camarinha-Matos & Afsarmanesh 2011). The phenomenon of collaborative networks is therefore not exclusively linked to the Information Age. Recent advances in the ICT domain, however, have revolutionised the state of collaboration, with virtual collaboration amongst geographically dispersed actors and entities enabling the emergence of a host of new organisational forms. Collaborative networks are currently observed in diverse forms and exhibiting a variety of behavioural patterns.

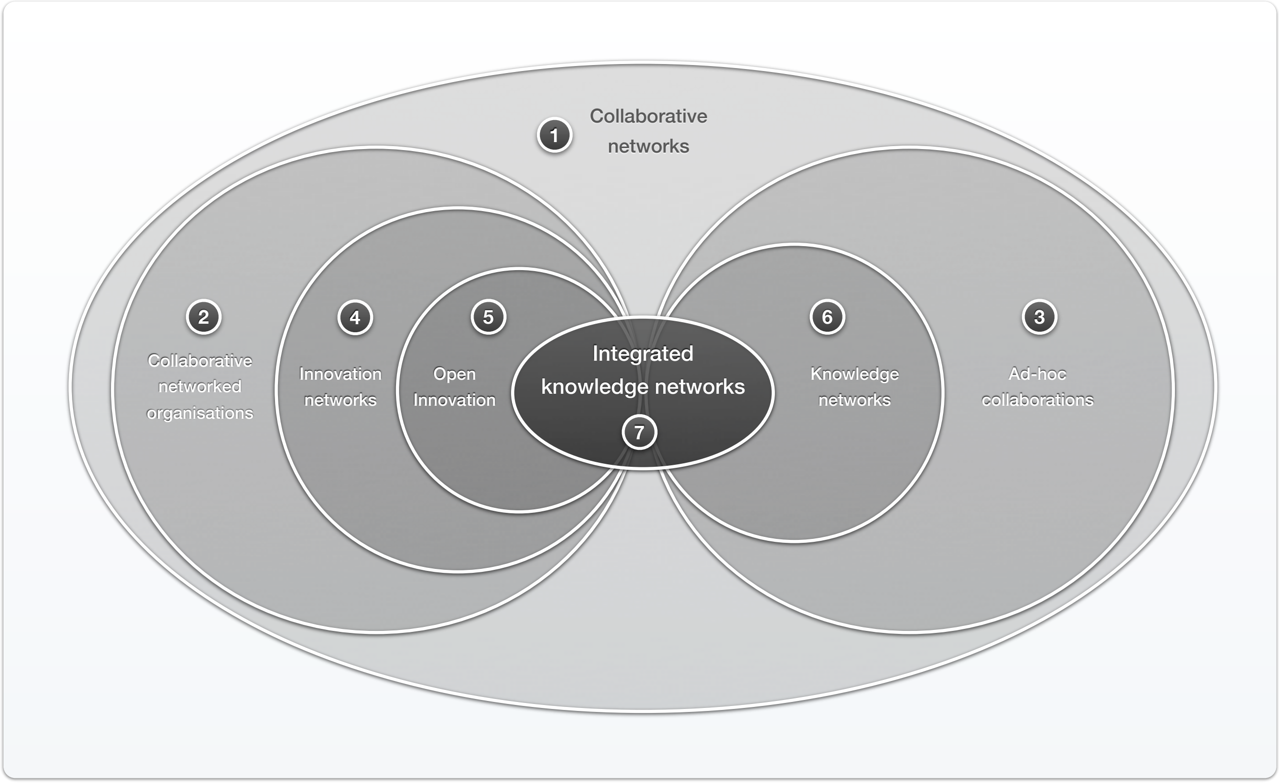


Figure 1.1: Organisational landscape of collaborative networks

Some collaborative networks (refer to Figure 1.1) (1) exhibit organisation of the activities of their members, e.g. identifying roles for the participants and implementing some form of governance. These networks are referred to as collaborative networked organisations (CNOs) (2), while collaborative networks that lack these traits are classified as ad-hoc collaborations (3). Amongst the wide variety of CNOs, it is possible to highlight those that are created with the mission to boost innovation, the so-called “innovation networks” (Berasategi et al. 2011) (4). Within this group of networks, there are those that embrace the concept of Open Innovation through the systematic integration of external inputs at different stages during the innovation process (Chesbrough 2003) (5).

Research is increasingly highlighting the fact that knowledge management is a key requirement for effective innovation management (Pérez-Bustamante 1999; Pittaway et al. 2004). Significant utilisation and exploitation of knowledge is, however, only enabled when networks are available to support it, and knowledge networking is therefore seen as an effective and holistic approach to knowledge management (Seufert et al. 1999). It could be argued that these knowledge networks are mostly ad-hoc collaborations, given the types of organisational forms that fall into this category (Anklam 2007) (refer to Figure 1.1) (6).

It is however recognised that to enable innovation within a knowledge network, this network needs to support the flow of knowledge through the knowledge supply chain (Du Preez, Louw & Lutters 2008a). This knowledge supply chain is equivalent to a knowledge generation value chain in which learning occurs between various domains (Gemünden et al. 1992; Gemünden et al. 1996; Geels 2004; Chesbrough & Prencipe 2008). The knowledge supply chain indicates how discovering new knowledge, making the knowledge transferable, transferring that knowledge through documentation and from person to person, and finally applying that knowledge, all support the innovation process. Such a network typically includes stakeholders from the public domain (academia and government), private domain (enterprises, including competitors and suppliers) and user domain (market).

Collaborative networks that support the knowledge supply chain through an inter-organisational knowledge network are viewed as *integrated knowledge networks* (IKNs) (7). These networks are hybrid forms of collaborative networks that combine traits from CNOs, especially innovation networks and the principles of open innovation, and the ad-hoc nature of knowledge networks. IKNs facilitate the creation and transfer of knowledge between network stakeholders, with the goal of fostering sustainable innovation which will continue to promote their competitiveness.

1.2.2 Enterprise engineering and enterprise architecture

In order to discuss the concept of EE in this section, it should first be noted that, for the purpose of this study, the following definition of an enterprise is adopted from Giachetti (2010). An enterprise is:

> *A complex, socio-technical system that comprises interdependent resources of people, information, and technology that must interact with each other and their environment in support of a common mission.* (Giachetti 2010:4)

The term “enterprise” therefore encompasses all types of organisations, including private companies, government, non-profits, supply chains, virtual enterprises, as well as parts of a company such as a division or program (Giachetti 2010). It could be argued that inter-organisational networks such as IKNs also subscribe to the definition above, and may therefore also be described as enterprises.

Three emerging disciplines currently contribute towards enterprise design and alignment with the aim of enabling enterprises to adapt to the challenges of the Information Age. These are 1) enterprise engineering (EE), 2) enterprise architecture (EA) and 3) enterprise ontology (EO). Limited literature is available on EO, but a number of publications exist on EE and EA. There is however a lack of shared meaning in terms of the *theoretical foundations*, *definitions* and *business benefits* of these approaches (Kappelman 2011; Lapalme 2012). This creates significant challenges in searching for relevant literature and advancing the EE and EA disciplines (Kappelman 2011; Lapalme 2012; De Vries 2012).

EE is not a new field, having originally developed as a sub-discipline of the systems engineering domain (Giachetti 2010), but is also not yet regarded as a discipline in the same vein as electrical or civil engineering. EE is currently largely practice-based and aims at studying enterprises in a multi-disciplinary and engineering-driven way, but often without much scientific foundation (Dietz 2006). There have, however, been recent attempts to establish the theories, goals, and fundamentals of EE as a discipline (Saenz et al. 2009; Dietz et al. 2013).

With regard to EA, the most common understanding of the term “architecture” for an enterprise is as a collection of artefacts (e.g. models and descriptions) that describe the as-is model of the enterprise (Kappelman 2011). This collection of artefacts serves to create a shared language to discuss and document important aspects of the enterprise, and also forms the basis for integrating and changing the enterprise toward a to-be model (The Open Group 2011; Kappelman & Zachman 2013). Other scholars, however, also associate EA with the process of defining these enterprise standards and creating as-is models (Bernard 2012). The presence of a multiplicity of definitions suggests that EA is a highly complex and dynamic construct that encapsulates botwh technical and social dimensions, the present and future, as well as the logical and physical aspects of the enterprise (Sidorova & Kappelman 2010).

1.2.3 Engineering enterprises as systems

EE inherits various concepts from its history within systems engineering, including the definition of enterprises as socio-technical systems (refer to section 1.2.2). In systems theory, and according to Dietz (2006) in EE as well, it is possible to differentiate between teleological systems and ontological systems. The teleological system notion describes the function and external behaviour of a system, and corresponds to the “black-box” model type. A “black-box” model essentially consists of the relation between a set of input variables and a set of output variables, called a transfer function. These types of models are adequate for the purpose of using or controlling a system, and is the dominant system concept in the social and organisational sciences (Dietz & Hoogervorst 2008).

An ontological system notion, however, needs to be adopted for the purpose of building and changing systems. The ontological system notion, with its associated “white-box” model, describes the construction and operation of a system, and is therefore the dominant system concept in the engineering sciences (Dietz & Hoogervorst 2008). There exists a fundamental relationship between the teleological and ontological system notions in that the function and behaviour of a system are brought about and explained by its construction and operation.

When designing a system, both the teleological (functional) and ontological (constructional) system definitions are relevant (Dietz & Hoogervorst 2008). The system design process starts with the need of a using system (US) for an object system (OS) (refer to Figure 1.2) (1). Once this need is understood, the requirements for the OS is determined through function design. These requirements for the OS by nature relate to the function and behaviour of the OS, and are thus articulated in terms of a “black-box” model of the OS (2).

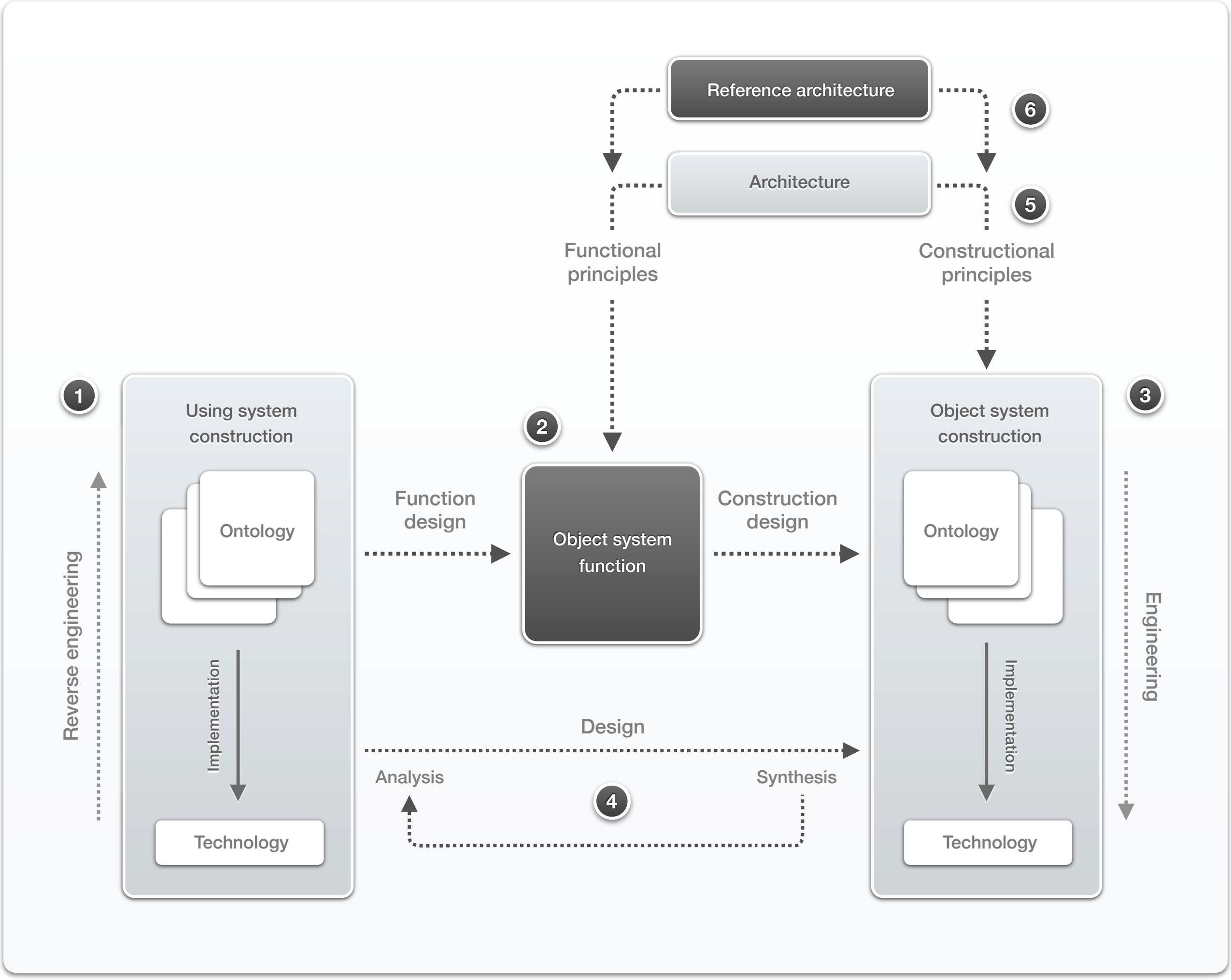


Figure 1.2: The generic system development process

(adapted from Dietz & Hoogervorst 2008)

The next step in the system design process is to specify the construction and operation of the OS in terms of a “white-box” model of the OS, i.e. construction design (3). The steps of function design and construction design correspond to analysis and synthesis (Alexander 1964) (4). This system design process, however, is an iterative one, with the end result being a balanced compromise between reasonable requirements and feasible specification.

In the iterative system design process discussed above, architecture influences both function and construction design in order to produce an integrated system. The role of architecture in the design of systems may be understood as “the normative restriction of design freedom” (Dietz & Hoogervorst 2008) that is required to arrive at an integrated system (5). It may be argued that these restrictions are expressed in terms of a consistent and coherent set of design principles embodied in the artefacts that comprise the architecture. Among these design principles, it is possible to distinguish between functional principles and constructional principles (Dietz & Hoogervorst 2008).

It is furthermore possible to define a reference architecture that holds for many systems, typically for a class of similar systems (Giachetti 2010) (6). Through this mechanism of reference architectures, it is possible to impose sound functional and constructional principles to ensure the integrated design of an entire class of systems. This study is concerned with the development of such a reference architecture that may be used to engineer IKNs as a class of enterprise systems.

1.3 Rationale for the study

This section presents the rationale for the study from both practical and scientific points of view, culminating in an articulation of the research problem that the study addresses.

IKNs are a phenomenon that have emerged due to the need to innovate in order to survive and compete in the rapidly changing environment brought on by the Information Age (Du Preez, Louw & Lutters 2008a). IKNs therefore need to be able to accommodate constant change, leading to the necessity for the engineering of these networks. Within such an environment, IKNs are assembled to order, rather than manufactured to order. This, however, requires an understanding of the fundamental components of these networks in order to enable the rapid assembly of composite structures (Kappelman & Zachman 2013).

However, a series of personally conducted exploratory interviews with individuals involved with the governance of operational IKNs revealed a number of issues related to the engineering of IKNs as described above. During these interviews, a number of recurring points of discussion confirmed the following issues in the current state of engineering of IKNs:

- Networks are designed “from scratch” and network leaders have to “figure things out as they go along”.
- This results in the wheel having to be reinvented for each successful network implementation, as there is no architectural reference available.
- Many network implementations, however, are unsuccessful due to a lack of engineering or the knowledge of how to construct an IKN from an architectural point of view.
- When trying to duplicate a successful network, they had great difficulty developing models that reflected their existing network’s architecture.
- When trying to improve their network, they had a rough sense of which elements to target but very limited architecture descriptions to base their efforts on and were therefore unsure about the required extent or implications of changes.
- Uncertainty about how to evolve and develop IKNs reduces their agility, i.e. their ability to react to changes in their environment.
- The concept of assembling an IKN to order, rather that manufacturing to order is a promising solution, but that the resources to achieve this do not exist.

From the above observations, it is possible to derive that practitioners find it difficult to engineer their networks, as it is a significant challenge to develop implementation-specific architecture descriptions. There is also evidence that the lack of a reference artefact to provide inputs to the development of these architecture descriptions and models in the engineering of IKNs is one of the major issues contributing to this problem as experienced in practice.

This description of the problem facing the engineering of IKNs in practice was verified by surveying the range of resources that are available to engineer networks. A large body of knowledge has been developed that describe the CNO phenomenon in general, and this knowledge has been consolidated through the ECOLEAD research project that contributed a reference model for CNOs in general (Camarinha-Matos & Afsarmanesh 2008a). Some partial reference models exist that more accurately describe certain manifestations of CNOs, e.g. virtual enterprises (Vesterager et al. 2003). These reference models have provided the foundations for more sustainable development in the CNO domain, enabling research toward improving the efficacy and efficiency of these organisational forms.

While deemed to relate to the CNO domain (refer to Figure 1.1), IKNs show significant differentiation from the generic form of CNOs, especially through their focus on the innovation process and incorporation of concepts from ad-hoc collaborations, for example knowledge networks (Schutte 2010, refer to Figure 1.1). This leads to the general reference model for CNOs not accurately describing the construction IKNs, while it could be argued that none of the other existing partial reference models for specific CNO classes address their specific nature either.

From the ad-hoc collaborative point of view (refer to Figure 1.1), various resources are available to describe knowledge networks, e.g. the Knowledge Network framework contributed by Back et al. (2005). This framework, however, primarily focuses on intra-organisational knowledge networks and does not explicitly support inter-organisational cases where the network is organised around innovation projects, as is the case with IKNs (refer to 1.2.1). Schutte (2010) built on the work done by Back et al. (2005) by deriving a methodology for the development of IKNs, including phases of network design, implementation, operation and refinement, as well as phase-out. Relating these contributions to Figure 1.2, they, however, both focus on providing functional principles to the design of knowledge networks and IKNs. These resources therefore primarily enable the development of the “black-box” models of knowledge networks and IKNs that are required to use and control the networks.

Given the discussion above, it may be argued that there is substantial evidence for the lack of an architectural resource that contributes constructional principles to IKNs as a class of systems. It is also possible to motivate that the lack of such an artefact is one of the major issues contributing to the problems experienced with the engineering of IKNs in practice. These initial findings support the identification of a need for a reference architecture in the engineering of IKNs, which is the focus of this study. This identified lack of a reference architecture that supports the engineering of IKNs leads to the following articulation of the *research problem* that this study will address:

No reference architecture exists for use in the engineering of IKNs.

The purpose of the study is to solve this identified problem through the development of a reference architecture for use in the engineering of IKNs. The contribution of such an artefact will extend the concept of enterprise engineering to the domain of IKNs, which enables much-needed adaptability of these networks to their environment. As is the case with CNOs in general, a reference architecture for IKNs will furthermore help to provide the foundations for more sustainable development in the IKN domain.

1.4 Research objectives and research questions

The purpose of this section is to state the primary research objective and research question considered in the study. Additional solution objectives and secondary research questions are identified in chapter 4, in which their derivation is also discussed in full.

Based on the rationale for the study as discussed in the previous section, the research objective (RO) of the study is:

To develop a reference architecture for use in the engineering of IKNs.

In order to achieve this primary research objective, the following primary research question (PRQ) is considered:

How can a reference architecture for use in the engineering of IKNs be developed?

A number of additional solution objectives and secondary research questions are derived and identified in chapter 4 through the application of a comprehensive approach to requirements specification. For the sake of completeness, these objectives and questions are presented here in Table 1.1 along with RO and PRQ.

Table 1.1: Objectives and research questions

Code	Research/solution objective	Code	Research question
RO	**To develop a reference architecture for use in the engineering of IKNs.**	**PRQ**	**How can a reference architecture for use in the engineering of IKNs be developed?**
SO1	To exhibit the desired behaviour of a reference architecture in the engineering of IKNs.	**SRQ1**	What is the desired behaviour of a reference architecture in the engineering of IKNs?
SO2	To employ an architecture framework that is	**SRQ2**	Is an architecture framework that is suitable

Code	Research/solution objective	Code	Research question
	suitable to the engineering of IKNs.		to the engineering of IKNs available?
SO3	To implement the selected architecture framework in a way that enables the artefact to function as a reference architecture.	**SRQ3**	How can the selected architecture framework be implemented in a way that enables the artefact to function as a reference architecture?
SO4	To function as a reference architecture in the engineering of IKNs through the implementation of a suitable architecture framework.	**SRQ4**	How can an artefact that functions as a reference architecture in the engineering of IKNs be developed through the implementation of a suitable architecture framework?

1.5 Research strategy

The purpose of this section is to provide an overview of the research strategy that is adopted in order to achieve the identified primary research objective. The methodology and research design adopted in the study are discussed in detail in chapter 3. The research design which is described in this section is also used to structure this document, as discussed in section 1.8.

The study is conducted within the philosophical paradigm of pragmatism, which is concerned with the generation and application of knowledge for action and change (Mead & Morris 1938). Within pragmatism, knowledge is developed through inquiry processes inspired by problem situations (Goldkuhl 2012), e.g. the rationale for this study as described in section 1.3.

Within this pragmatic paradigm, the study adopts a qualitative approach to the above-mentioned inquiries with the aim to uncover patterns that enable a better understanding of the construction of IKNs. Design science research (DSR) is employed as a research method, with its mission of solving problems by building artefacts that are complementary to pragmatism (Hevner & Chatterjee 2010b; Goldkuhl 2012).

The DSR methodology presented by Peffers et al. (2008) is adapted and employed in five steps in this study to construct a reference architecture for IKNs (refer to Figure 1.3):

1. **Problem identification and motivation.** The research problem which is the basis for the study is identified and motivated from both practical and theoretical points of view.

2. **Definition of solution objectives.** A comprehensive approach to requirements specification is employed to rationally infer objectives for the solution artefact accompanied by secondary research questions.

3. **Design and development.** The artefact is constructed in four iterative design cycles that each consider the artefact from a different point of view in order to achieve a solution objective and answer the accompanying secondary research question. Each of these cycles features design reasoning, including an awareness, a suggestion and development of a version of the artefact.

    3.1. **First design cycle.** The first design cycle, indicated in green in Figure 1.3, produces the first version of the artefact in the form of a specification of the desired behaviour of a reference architecture in the engineering of IKNs.

    3.2. **Second design cycle.** The second design cycle (yellow) produces the second version of the artefact in the form of the selection of an architecture framework that is suitable to the engineering of IKNs.

    3.3. **Third design cycle.** The third design cycle (red) produces the third version of the artefact in the form of a roadmap that specifies how the selected architecture framework may be implemented to enable the artefact to function as a reference architecture.

    3.4. **Fourth design cycle.** The fourth design cycle (blue) produces the fourth and final version of the artefact by constructing the reference architecture for IKNs. This artefact functions as a reference architecture in the engineering of IKNs through the implementation of a suitable architecture framework.

4. **Demonstration and evaluation.** The way in which the reference architecture for IKNs attempts to achieve the solution objectives defined in the second step is demonstrated, and its success in doing so is evaluated.

5. **Communication.** The contribution made by the reference architecture for IKNs in solving the problem identified in the first step is reflected upon. Furthermore, the utility and novelty of the artefact, the rigour of its design, and its effectiveness are communicated to both practical and scientific audiences.

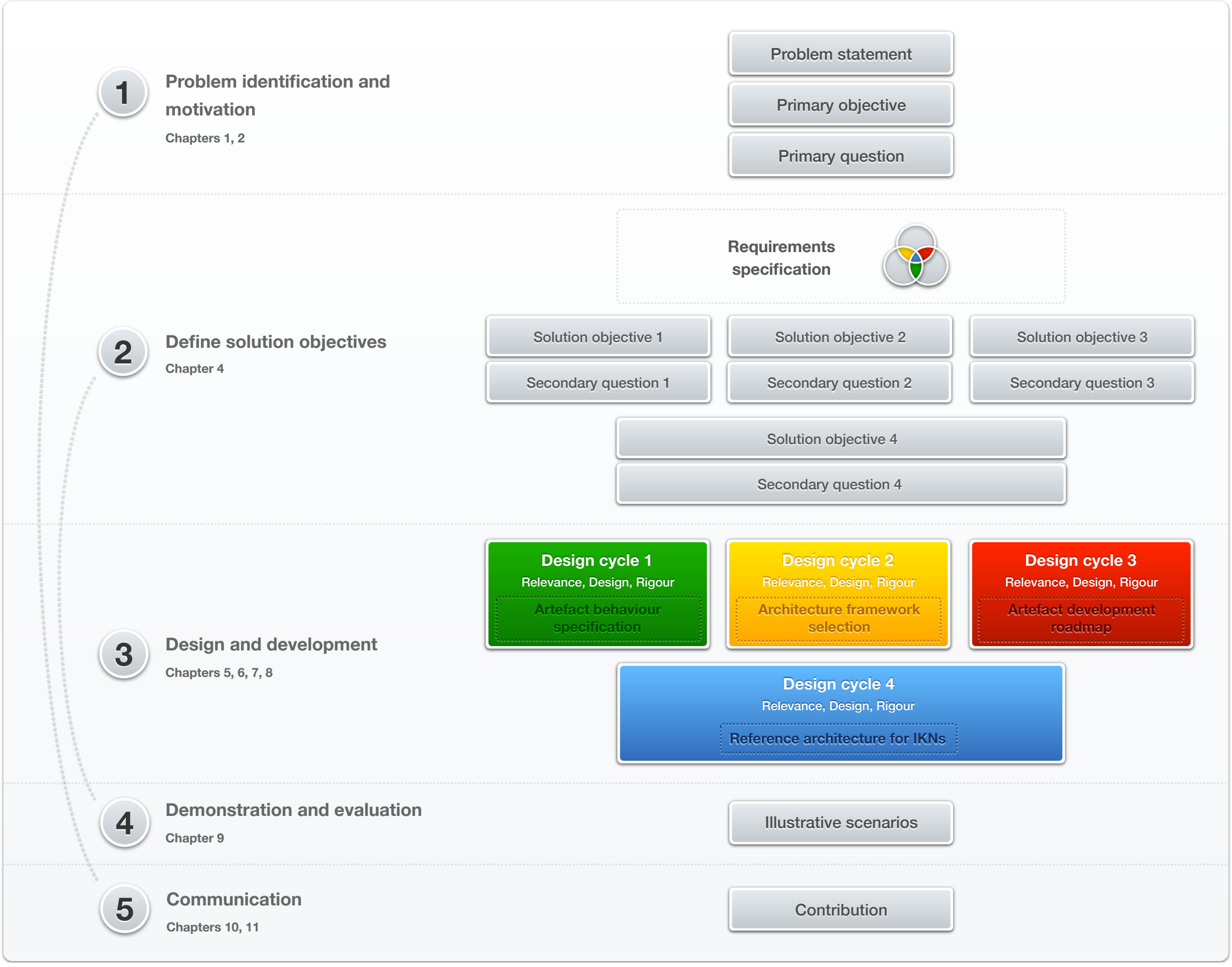


Figure 1.3: Research design

1.6 Scope and delineations

The purpose of this section is to clearly specify the scope of the study through various delineations. This is done in order to manage expectations regarding the findings of the study.

The primary objective of the study is to develop a reference architecture for use in the engineering of IKNs. Given this primary objective, the scope of the study is outlined as follows:

- The study investigates underlying constructional patterns present in IKNs in order to compile an artefact that captures and represents these patterns. Since the patterns will undoubtedly differ in CNO forms that fall outside the bounds of IKNs as presented in section 2.5, the scope of the study is limited to the investigation of IKNs. This does not rule out limited applicability of the patterns represented in the reference architecture for IKNs to CNOs in general, but the in-depth investigation of CNOs is beyond the scope of the study.

- The study focuses on the engineering of IKNs and investigates the constructional (“white-box”) principles of IKNs (refer to Figure 1.2). The investigation of the functional (“black-box”) principles of IKNs is therefore beyond the scope of the study, and this aspect has been researched in the study conducted by Schutte (2010).

- Artefacts such as the ARCON reference model for CNOs (Camarinha-Matos & Afsarmanesh 2008a) describe both the endogenous and exogenous environments of their modelling subject. However, since the reference architecture for IKNs focuses on the provision of constructional principles to the engineering of IKNs, the study is limited to an endogenous view on these networks.

- The reference architecture for IKNs may be described as an ontology that describes the nature of being for IKNs by employing the Zachman framework (Zachman 2011; Kappelman & Zachman 2013) to structure the knowledge in the domain. From a functional point of view, the artefact provides constructional principles to the development of particular IKNs as systems, as discussed in section 1.2.3 and illustrated in Figure 1.2. The reference architecture for IKNs does therefore not include a methodology and is seen as complementary to the methodology presented by Schutte (2010). The in-depth research of an optimised version the systems development process for IKNs is therefore beyond the scope of the study.

- In order to achieve a holistic ontological view on IKNs, the investigation into the underlying constructional patterns of these networks is seen as “enterprise-wide”. No element of the construction of the networks, e.g. roles, processes, life cycles or data is therefore either emphasised or excluded.

- As is motivated and discussed in chapter 7, the information included in the reference architecture for IKNs is limited to a strategic and business level. This is primarily due to the feasibility of generalising the constructional patterns that are present at these levels, versus those that are present at more technical levels. Although the feasibility of reference models at a system logic level could be argued for, the extension of the reference architecture for IKNs to this level is beyond the scope of the study.

- The content of the reference models included in the reference architecture for IKNs reflect the current understanding of various topics in the IKN domain. As the scientific understanding of these complex phenomena deepens in the future, the content of certain reference models may require updating. The structure of the artefact, however, ensures that such updates to the reference architecture will not compromise the integrity of the artefact as a whole.

- The DSR methodology elected as part of the research strategy for the study, includes three underlying cycles, namely relevance, design and rigour (Hevner 2007). The relevance cycle is of particular importance in a pragmatic paradigm, since the developed solution artefact has to be initiated by a real-world problem (refer to section 1.3). It is also important that the usefulness of the artefact in solving the problem is demonstrated and evaluated in practice. The timeframe afforded to the study, however, did not allow for extensive field-testing of the reference architecture for IKNs through case studies in order to determine the effect of the artefact on organisations in which it is implemented. Such field-tests would typically take several years to generate reliable data, and involve significant risk to the organisations that are involved. The evaluation of the reference architecture for IKNs in this study is therefore limited to a series of illustrative scenarios in which the utility of the artefact is rigorously investigated (refer to chapter 9). This delineation is acknowledged in the title of the thesis: “Towards a reference architecture for integrated knowledge networks”.

1.7 Significance and contribution

This section describes the scientific significance of the study, including the contributions that it produces. This is done in order to motivate the value of performing the study.

The primary contribution of the study is a solution to the problem described in section 1.3 in the form of a *reference architecture for* IKNs that contributes towards enhancing the current state of IKN engineering and development. This artefact provides constructional principles to the development of these networks, which had previously been lacking.

From a practitioner’s perspective, the reference architecture for IKNs improves the *ability of practitioners to build and change networks*. IKNs that are built based on the inputs and principles provided by the reference architecture are constructionally sound, require a shorter roll-out time and are more replicable. In terms of changing networks, the reference architecture for IKNs provides reference models against which practitioners may benchmark the construction of existing networks. The constructional principles provided by the reference architecture also allow for the design of refined versions of networks and are a valuable resource in change management.

In terms of the scientific significance of the study, the development of the reference architecture for IKNs takes a first step toward *extending the concepts of enterprise engineering and enterprise ontology to the domain of CNOs*. The study complements existing resources that focus on contributing functional principles to the domain, e.g. the methodology for IKNs (Schutte 2010). Having both constructional and functional perspectives available (refer to Figure 1.2), means that IKNs may now be comprehensively designed, operated, and indeed studied as systems. The reference architecture for IKNs provides the foundation for more sustainable development in the IKN domain, since it completes a comprehensive systems view of this class of networks.

The reference architecture for IKNs employs the Zachman framework to structure its content internally; therefore, the study is an example of the use of the framework to structure the body of knowledge in a certain domain.

The secondary contributions of the study include:

- A description of the desired behaviour of a reference architecture in the engineering of IKNs,
- The use of the Zachman framework to describe non-hierarchical enterprises,
- The use of the Zachman framework as an architecture framework in the development of reference architectures,
- A visual representation format for the Zachman framework, and
- A research design for use in design science research.

1.8 Structure of the document

This document is structured according to the research design presented in section 1.5 (refer to Figure 1.3). The document therefore consists of five parts containing a total of eleven chapters. The scope and purpose of each part of the document are determined by the research design adopted by the study. The number of chapters included in each part varies, and each part is accompanied by a unique visual cue, as indicated below.

Part 1 - Problem identification and motivation

The research problem, which is the basis for the study, is identified and motivated from both practical and scientific points of view.

Chapter 1	*Introduction* Introduces the research problem and provides an overview of the study.
Chapter 2	*Background* Provides background information from literature that elaborates on the identification and motivation of the research problem in chapter 1.

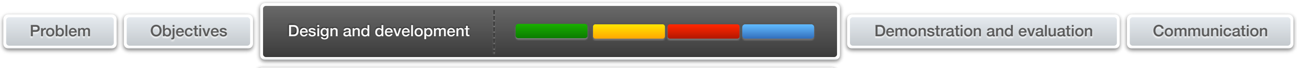
Chapter 3	*Methodology and research design* The chapter presents a detailed discussion of the methodology, including the

research design, adopted in the study to solve the problem that was identified and motivated in Part 1. Since the document structure is based on the research design presented in this chapter, the chapter itself does not belong to any specific part of the document.

Part 2 – Definition of solution objectives

Chapter 4	*Solution objectives* Defines various solution objectives and secondary research questions through the implementation of a comprehensive approach to requirements specification.

Part 3 – Design and development



The artefact is constructed in four iterative design cycles that each consider the artefact from a different point of view in order to achieve a solution objective and answer the accompanying secondary research question.

Chapter 5	*Artefact behaviour specification* Presents the first design cycle (green) and constructs the artefact behaviour specification as the first version of the solution artefact.
Chapter 6	*Architecture framework selection* Presents the second design cycle (yellow) and selects a suitable architecture framework as the second version of the solution artefact.
Chapter 7	*Artefact development roadmap* Presents the third design cycle (red) and compiles an artefact development roadmap as the third version of the solution artefact.
Chapter 8	*Reference architecture for IKNs* Presents the fourth and final design cycle (blue) and constructs the reference architecture for IKNs as the fourth version of the solution artefact.

Part 4 – Demonstration and evaluation

Chapter 9 *Demonstration and evaluation*

Demonstrates the way in which the reference architecture for IKNs attempts to achieve the solution objectives defined in Part 2, and evaluates its success in doing so.

Part 5 – Communication

The contribution made by the reference architecture for IKNs in solving the problem identified in Part 1 is reflected upon. The utility and novelty of the artefact, the rigour of its design, and its effectiveness is communicated to both practical and scientific audiences.

Chapter 10 *Contribution*

Reflects on the primary contribution made by the reference architecture for IKNs in solving the problem, as well as various secondary contributions.

Chapter 11 *Conclusion*

Concludes the document by providing an overview of the study that communicates the value of the research that was done.

2. Background

2.1 Introduction

This chapter presents background information from literature that contextualises and reinforces the research problem identified in section 1.3.

The current landscape of collaborative networked organisations is discussed in section 2.2. Among these organisational forms, an emergent collaborative network form is identified, namely inter-organisational innovation networks that are driven by knowledge sharing. These organisations may be described as IKNs, i.e. multi-domain knowledge-sharing networks aimed at fostering sustainable innovation that promote the competitiveness of their members. In order to characterise IKNs, the concepts of knowledge and innovation are subsequently discussed.

Section 2.3 therefore presents an overview of the knowledge concept, including a discussion of various types of knowledge, knowledge generation and knowledge work processes, as well as knowledge networks. Another key concept in understanding IKNs, namely *innovation*, is discussed in section 2.4. The phenomenon of innovation is defined, distinguished from other related terms, and types of innovation are identified. The innovation life cycle is discussed, followed by a discussion of various generations of innovation models. Particular attention is paid to recent models describing innovation networks and open innovation, leading to a discussion of IKNs.

IKNs are discussed in section 2.5, starting with an overview of the relationship between knowledge and innovation in these networks. The knowledge supply chain is discussed as a key concept in the definition of IKNs. The existing body of knowledge on IKNs, which amount to a functional view on these networks, is then presented. The need for a constructional view on IKNs is identified, which leads to a discussion of EE and EA in the following section.

EE is discussed in section 2.6, adopting a systems view on enterprises. This is followed by a discussion of the enterprise life cycle and the design of enterprises as systems. The role of EA to ensure the design of integrated enterprises through EE is discussed in section 2.7, along with a discussion of the reference architecture mechanism. Several architecture frameworks are reviewed, namely the Zachman framework, PERA, GERAM and VERA, TOGAF and ARCON, before section 2.8 concludes this chapter.

Concise literature reviews are also included in sections 5.3, 6.3 and 7.3. Those sections present focussed literature that specifically inform the problem awarenesses for the design cycles discussed in the respective chapters.

2.2 Collaborative Networked Organisations

This section introduces the field of collaborative networks and collaborative networked organisations. In section 2.2.3, IKNs are identified as being part of this domain.

Collaborative networks have an extensive history, with some instances being dated to the ancient Silk Road and more recently to Dutch and Spanish navigation ventures (Camarinha-Matos & Afsarmanesh 2011). The Industrial Revolution of the late 1700s and early 1800s affected and reformed the collaborative networks of the day, which existed mainly in the form of global supply chains (Dekkers 2010).

During the 1970s and 1980s various Japanese manufacturing concepts such as “just-in-time” and “co-makership” attracted research interest due to their reliance on interactions among autonomous but cooperating entities (Bevan 1987). In the 1990s the drive toward lower costs, greater efficiency and responsiveness put the paradigm of core competencies in the spotlight, and led to the consequent move toward outsourcing (Barthélemy 2003). The adoption of co-makership and alliances, the search for flexibility, as well as greatly enhanced and technologically enabled manufacturing capabilities have resulted in a shift toward increasingly interconnected industrial entities. These industrial networks are perceived as solutions to the required flexibility and agility in response to changes in market demands (Camarinha-Matos & Afsarmanesh 2005).

Recent advances in the ICT domain, however, have again revolutionised the state of collaboration, with virtual collaboration among geographically dispersed actors and entities enabling the emergence of a host of new organisational forms (Choudary et al. 2013). Collaborative networks are currently observed in such diverse forms and exhibit such a variety of behavioural patterns that it leads to difficulties both in characterisation of the domain, as well as communication among experts (Camarinha-Matos & Afsarmanesh 2012).

2.2.1 Collaborative Networks

The term “collaborative networks” is inspired by the *networking* metaphor as the actors in these networks, be it organisations or individuals, resemble a distributed collection of nodes (Anklam 2007). These nodes are connected through links that represent interactions or collaborative relationships.

Three collaborative network topologies appear frequently in literature, namely chain topology, star topology and general network topology (Katzy et al. 2005). In a chain topology (refer to Figure 2.1) (1) the node interaction patterns mainly follow a value chain, as is the case in supply chains in the manufacturing industry. In a star topology (2) nodes interact with a central hub or dominant member, which is typically the case in the automotive industry. In a general network topology (3) entities share multiple relationships without hierarchy, as in project-oriented and knowledge-based industries.

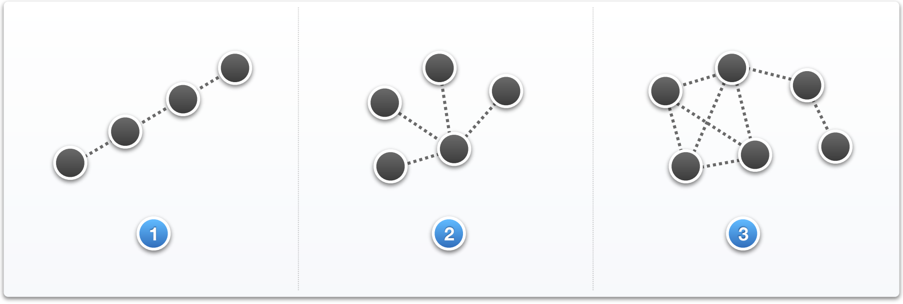


Figure 2.1: Topologies of collaborative networks

(adapted from Katzy et al. 2005)

The network metaphor, however, does not fully capture the dynamic reconfiguration of links, relationships and interactions that involve several nodes at once. Furthermore, relationships exist between nodes (network stakeholders) and the network itself, representing notions such as sharing and belonging (Camarihna-Matos & Afsarmanesh 2012). This perspective on the collaboration is better captured by the *community* metaphor, which is considered implicitly in the notions of “group identity”, “co-creation” and “co-innovation”.

The term “collaborative network” should be understood as inheriting ideas from both the networking and community metaphors. In order to capture the essence of these perspectives and to provide a theoretical framework for future development of the domain, collaborative networks (CNs) have emerged as a new scientific discipline (Camarinha-Matos & Afsarmanesh 2005).

2.2.2 From networking to collaboration

The notion of collaboration is often confused with that of cooperation, with the terms often erroneously being used interchangeably (Denise 1999; Pollard 2005). The situation becomes more ambiguous when terms such as networking, communication and coordination are considered part of the same discussion (Grosz & Kraus 1996; Himmelman 2001).

Four particular terms, namely *networking*, *coordinated networking*, *cooperation* and *collaboration* are now defined and discussed in relation to one another.

2.2.2.1 Networking

Networking is defined as:

> *A process involving communication and information exchange among participants for mutual benefit.* (Camarinha-Matos & Afsarmanesh 2006)

Networking commonly occurs in situations in which individuals share their experiences with a specific tool to their mutual benefit, without there necessarily being a common goal or structure to the networking activities (Anklam 2007). Contemporary social networks such as Facebook and Twitter are also examples of networking.

2.2.2.2 Coordinated networking

Coordinated networking is defined as:

> *Coordination is a framework used to ensure that otherwise disparate forces will all pull in harness. It is therefore a process that, in addition to communication and exchanging information, involves aligning or altering activities so that more efficient results are achieved.*
> (Denise 1999; Camarinha-Matos & Afsarmanesh 2006)

Coordination entails acting together harmoniously, e.g. when entities with their own resources and objectives adjust the timing of certain activities to maximise their impact on a common goal (Pollard 2005).

2.2.2.3 Cooperation

Cooperation is defined as:

> *A process that involves not only information exchange and adjustments of activities, but also sharing resources for achieving compatible goals. Cooperation is achieved by division of some labor (not extensive) among participants.* (Denise 1999; Pollard 2005)

Participants in traditional supply chains cooperate by mostly working individually on specific tasks, with these tasks representing the decomposition of a larger process (Mentzer et al. 2001). The management of this larger process may require intermittent synchronisation and interaction.

2.2.2.4 Collaboration

Collaboration is defined as:

> *A process in which entities share information, resources and responsibilities to jointly plan, implement, and evaluate a program of activities to achieve a common goal.*
> (Grosz & Kraus 1996)

The collaboration process is one of shared creation and entails that a group of entities enhance one another’s capabilities. Collaboration “implies sharing risks, resources, responsibilities and rewards, which, if desired by the group, can also present an outside observer with the image of a joint identity. Collaboration involves mutual engagement of participants to solve a problem together, which implies mutual trust and thus takes time, effort and dedication. In collaboration, parties are more closely aligned in the sense of “working together” to reach the desired outcome, rather than that outcome being achieved through “individualistic” participation constrained by contextual factors such as those imposed by client-supplier or subcontracting relationships” (Camarihna-Matos & Afsarmanesh 2012).

Each of these four concepts is incorporated and implied in the next one (refer to Figure 2.2), with collaboration being the most demanding concept, requiring a high level of participation among participants (Grosz & Kraus 1996; Denise 1999; Himmelman 2001; Pollard 2005). Progression from networking to collaboration entails increased common goal-oriented risk-taking, commitment and joint investment, leading to the different levels of interaction that represent “collaboration maturity levels” (Camarinha-Matos & Afsarmanesh 2008b).

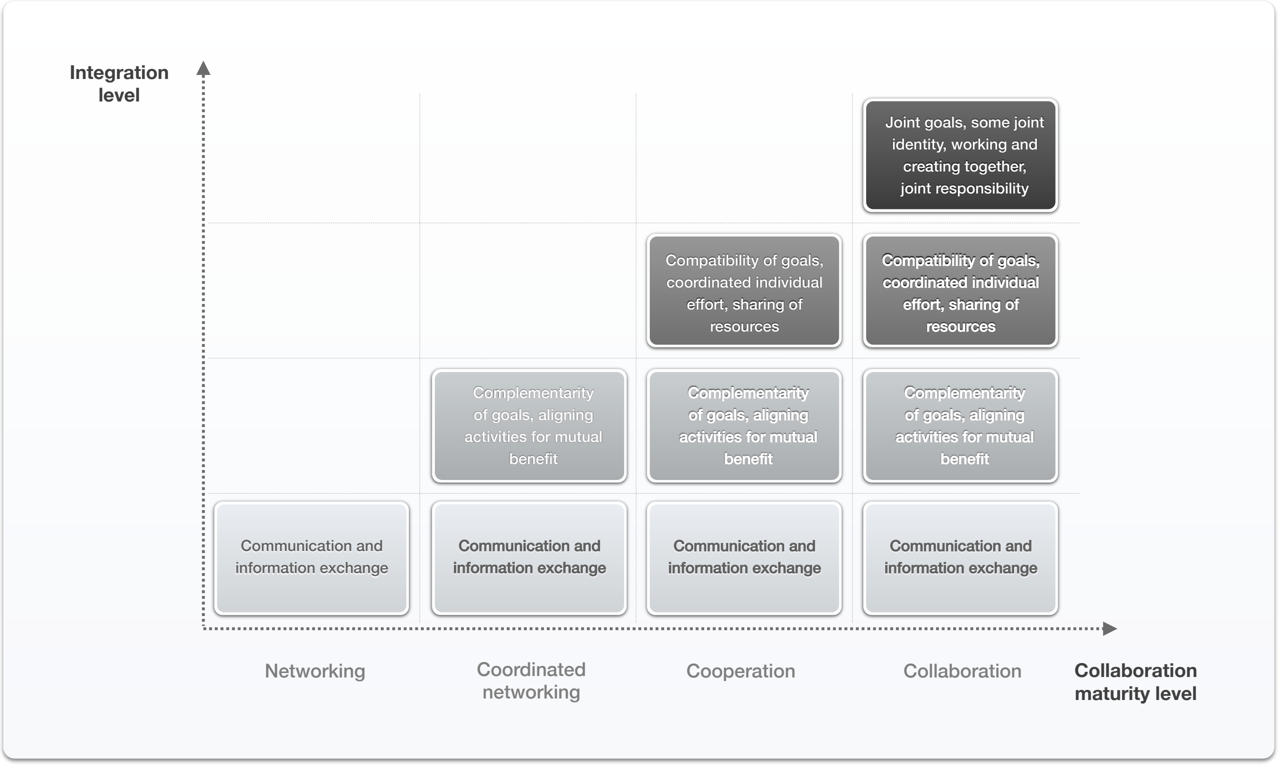


Figure 2.2: Building blocks of collaboration

(adapted from Camarinha-Matos et al. 2008)

In practice, levels of interaction between participants in collaborative networks ebb and flow, with phases of intense collaboration being interspersed between phases limited to cooperation (Camarinha-Matos & Afsarmanesh 2008b). Since the understanding of collaboration also implies understanding of the less demanding concepts, most research in this particular domain focuses on collaborative networks.

2.2.3 Base organisational forms

A wide array of collaborative network forms can be observed in society, with the internet, in particular, enabling a host of new organisational forms. These various forms of collaborative networks are spread over various application domains, leading to different terminologies. Two of the leading authors on collaborative networks, Camarinha-Matos and Afsarmanesh, contributed an open taxonomy of collaborative networks (refer to Figure 2.3) that describes the various forms observable today, but also allows for the inclusion of emergent forms (Camarinha-Matos & Afsarmanesh 2012).

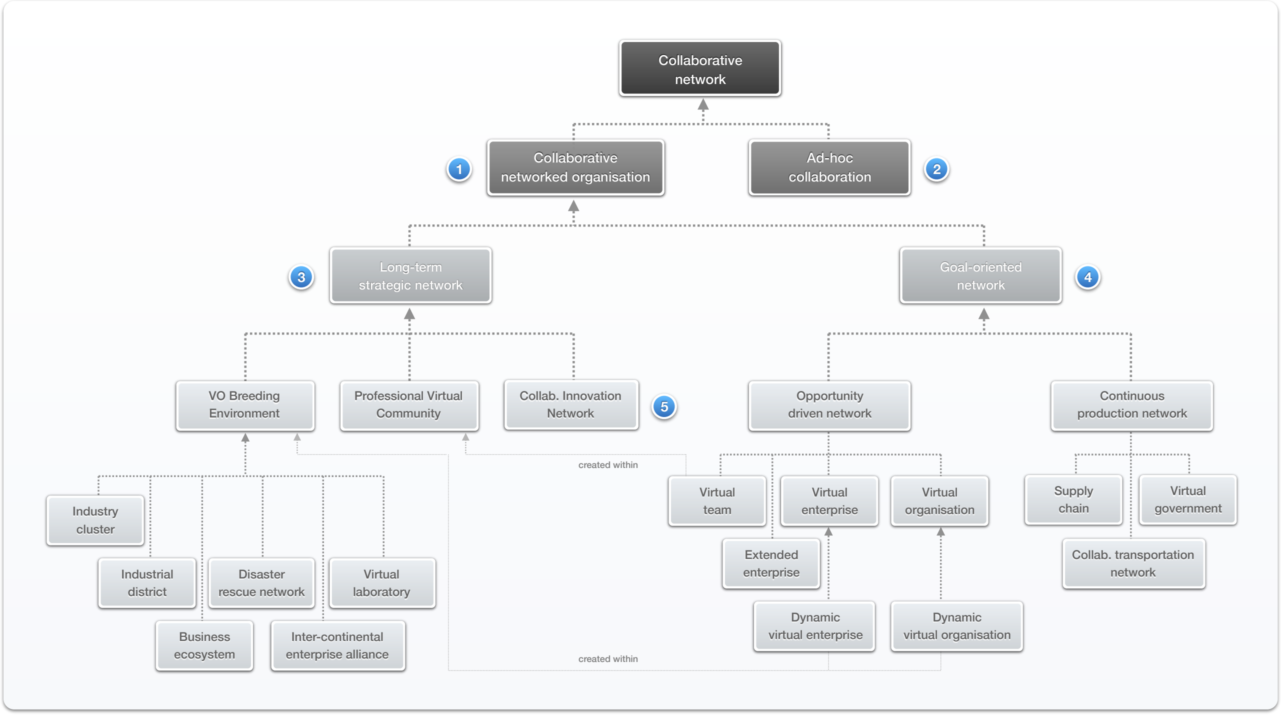


Figure 2.3: Taxonomy of Collaborative Networks

(adapted from Camarinha-Matos & Afsarmanesh 2012)

A number of collaborative networks exhibit some form of organisation of the activities of their members, identifying roles for the participants, implement some form of governance, and are subsequently referred to as collaborative networked organisations (CNOs) (refer to Figure 2.3) (1). Other collaborative networks lack these traits and are classified as ad-hoc collaborations (2).

Among networks that are classified as CNOs, a further distinction is made between long-term strategic networks (3) and fleeting, goal-oriented networks (4). Each of these classes is further divided into distinct organisational forms. Definitions for all of these base forms of CNOs are presented in Table 2.1.

However, it should be noted that hybrid forms of CNOs are commonly observed, while network instances may exhibit the characteristics of various base organisational forms (Camarinha-Matos & Afsarmanesh

2012). Another common phenomenon is the creation of goal-oriented networks in the context of long-term strategic networks, e.g. the creation of VEs and VOs as products of VBEs (Ermilova & Afsarmanesh 2007).

Table 2.1: Taxonomy of Collaborative Networks

(Camarinha-Matos & Afsarmanesh 2012)

Organisational form	Definition
Collaborative network (CN)	A network consisting of a variety of entities (e.g. organisations, people and intelligent machines) that are largely autonomous, geographically distributed and heterogeneous in terms of their operating environment, culture, social capital and goals. These entities, however, collaborate to better achieve common or compatible goals, and their interactions are supported by a computer network.
Collaborative networked organisation (CNO)	A CN possessing some form of organisation in terms of structure of membership, activities, definition of roles of the participants, and following a set of governance principles and rules.
Ad-hoc collaboration	A “spontaneous” form of collaboration without a precise structure or pre-defined organisation.
Long-term strategic network	A strategic alliance established with the purpose of being prepared for participation in collaboration opportunities, and in which the interaction between members is weighted toward cooperation, rather than collaboration. These alliances are aimed at offering the conditions and environment to support rapid and fluid configuration of collaborative networks when opportunities arise.
Virtual organisation breeding environment (VBE)	An association of organisations and a number of related supporting institutions, adhering to a base long-term cooperation agreement, and adopting common operating principles and infrastructures. The main goal of a VBE is to increase the preparedness of its members for rapid configuration of temporary collaborative alliances in VEs and VOs.
Industry cluster	Consists of a group of companies, typically located in the same geographic region and operating in a common business sector, that maintain some links to each other in order to increase their general competitiveness in the larger area.
Industrial district	A term mostly used in Italy that refers to a concept similar to an industry cluster. It can be focused on one single sector or cover a number of sectors in a given region.
Business ecosystem	Similar to a cluster or industry district, although not limited to one sector, but rather tends to cover the key sectors within the geographical region. A business ecosystem is inspired by the mechanisms of the biological ecosystems and attempts to preserve local specificities, tradition and culture, and frequently benefits from (local) government incentives.

Organisational form	Definition
Inter-continental enterprise alliance	A special case of a VBE involving sub-networks of enterprises located on different continents.
Disaster rescue network	A local or global strategic alliance of (non-)governmental organisations specialising in rescue operations in case of disasters, aimed at facilitating a rapid and well-coordinated response.
Virtual laboratory (VL)	An alliance of autonomous research organisations, each having their own resources (equipment, tools, data, etc.), enabling their researchers, located in different geographically-spread centres to be recognised and considered for taking part in potential opportunistic problem-solving collaborations.
Professional virtual community (PVC)	An alliance of professional individuals, which provides an environment to facilitate the agile and fluid formation of temporary coalitions of experts in Virtual Teams (VTs), based on the needs of a particular business opportunity.
Collaborative innovation network (CoIN)	A collaborative organisation, similar to a PVC, comprising a group of self-motivated individuals with a collective vision, enabled by the internet and ICT tools, to collaborate in creating a new trend (innovation) by sharing ideas, information, knowledge and work. The main focus in a CoIN is on pursuing innovation through collective intelligence.
Goal-oriented network	A CN in which intense and well-focused cooperation and/or collaboration, toward a common goal or a set of compatible goals, are/is practiced among their partners.
Opportunity driven network	A CN driven by the aim of grasping a single collaboration opportunity and that dissolves after the goal is accomplished.
Virtual enterprise (VE)	A temporary alliance of enterprises that come together to share skills or core competencies and resources in order to better respond to business opportunities, and whose collaboration is supported by computer networks.
Virtual organisation (VO)	A concept similar to a virtual enterprise, comprising a set of (legally) independent organisations that share resources and skills to achieve its goal, but that is not limited to an alliance of profit-oriented enterprises. A VE is therefore a particular case of a VO.
Dynamic virtual enterprise / organisation	A VE/VO that is established in a short time to respond to a competitive market opportunity, and has a short life cycle, and dissolves when the short-term purpose of the VE/VO is accomplished.
Extended enterprise (EE)	A concept typically applied to a networked organisation in which a dominant enterprise "extends" its boundaries to all or some of its suppliers. An EE can be seen as a particular case of a VE (in case of a temporary and goal-oriented EE) or of a supply chain (in the case of a long-term structure).

Organisational form	Definition
Virtual team (VT)	A similar structure to a VE but formed by individuals, not organisations, comprising a temporary group of professionals or knowledge workers that work together toward a common goal, e.g. realising a consultancy job, establishing a joint project, and using computer networks as their main interaction environment.
Continuous production network	A CN driven by or oriented to continuous production/service provision activities.
Supply chain	A stable long-term network of enterprises each having clear roles in the manufacturing value chain, covering all steps from initial product design and the procurement of raw materials, through production, shipping, distribution and warehousing until a finished product is delivered to a customer.
Virtual government	An alliance of governmental organisations (e.g. city hall, tax office, and civil infrastructures office) that combine their services through the use of computer networks to provide integrated services to the citizen through a common front end.
Collaborative transportation network	A long-term CN involving a diversity of actors such as road management entities, logistic operators, parking management entities, gas stations and banks in order to provide integrated transportation services.

Over and above the organisational forms and domain applications presented in Table 2.1, collaborative networks are emerging as promising approaches in a number of new domains. Amongst other examples, Camarinha-Matos and Afsarmanesh (2012) identify “open innovation networks” as one such new application of the collaborative network concept. These “open innovation networks" share some concepts with the collaborative innovation networks (CoINs, refer to Figure 2.3 (5)) defined in Table 2.1. Both of these network forms are examples of innovation networks driven by knowledge sharing (Schutte 2010; Berasategi et al. 2011). In some of these networks individuals are primarily involved (Von Hippel 2005; Von Hippel 2007b), while in other instances inter-organisational collaboration is enabled (Powell et al. 1996). The integrated knowledge network (IKN) concept (refer to section 2.5), addresses this entire domain by describing multi-domain knowledge-sharing networks aimed at fostering sustainable innovation that promote the competitiveness of their members (Du Preez, Louw & Lutters 2008a; Schutte 2010).

The knowledge (refer to section 2.3) and innovation (refer to section 2.4) concepts that define IKNs are now presented to provide the necessary context for a discussion of the rationale and characteristics of IKNs themselves (refer to section 2.5).

2.3 Knowledge

This section presents fundamental concepts regarding knowledge as context for the discussion of IKNs in section 2.5.

2.3.1 Definition and philosophy

Epistemology is a branch of philosophy that deals with the theory of knowledge, especially with regard to its nature, methods, validity and scope, and the distinction between justified belief and opinion (Oxford Dictionaries n.d.).

A classic epistemological definition of knowledge reads as follows:

> 1. *Facts, information, and skills acquired through experience or education; the theoretical or practical understanding of a subject.*
>
>   1.1 *The sum of what is known.*
>
>   1.2 *Information held on a computer system.*
>
>   1.3 ***True, justified belief**; certain understanding, as opposed to opinion.*
>
> 2. *Awareness or familiarity gained by experience of a fact or situation.*
>
> (Oxford Dictionaries n.d.)

This definition shows that knowledge can be viewed as beliefs that are justified as truth. Nonaka et al. (2000), however, particularly object to this Western view on epistemology that sees knowledge as “justified true belief”, and contends that it “fails to address the relative, dynamic and humanistic dimensions of knowledge”. Nonaka et al. (2000) therefore rather proposes the following definition for knowledge:

> *A dynamic human process of justifying personal belief toward the ‘truth’.*

This view on knowledge holds that knowledge is dynamic, as it is created in social interactions amongst individuals and organisations (Nonaka et al. 2000). Knowledge is also context-specific, and depends on a particular time and space (Hayek 1945; Nonaka et al. 2000). Knowledge is furthermore related to human action and has a subjective nature represented by terms like ‘belief’, which is rooted in personal value systems (Nonaka et al. 2000).

Based on these definitions of knowledge, Bornemann et al. (2003) identify three basic characteristics of knowledge.

- Knowledge is created dynamically through changes to cognitive structures.

- Knowledge is intrinsically linked to people.
- Knowledge is a prerequisite for human action.

Considering these characteristics of knowledge, it is clear that managing knowledge as an organisational asset requires a different approach to those employed for other, more tangible assets. Indeed, Bornemann et al. (2003) state that “knowledge management is not the management of knowledge itself, but rather the management of the organisation with a particular focus on knowledge”.

2.3.2 Types of knowledge

Based on the characteristics of knowledge presented in the previous section, various distinctions can be made between different types of knowledge. For instance, based on knowledge being linked to people, one way to categorise knowledge would be according to the ease with which it can be **articulated** (Polanyi 1962). Nonaka et al. (2000) formulated particularly insightful descriptions of explicit and tacit knowledge according to this distinction:

- **Explicit knowledge** can be expressed in formal and systematic language and shared in the form of data, scientific formulae, specifications, manuals and such like. It can be processed, transmitted and stored relatively easily.
- **Tacit knowledge** is highly personal and hard to formalise. Subjective insights, intuitions and hunches fall into this category of knowledge. Tacit knowledge is deeply rooted in action, procedures, routines, commitment, ideals, values and emotions. It ‘indwells’ in a comprehensive cognisance of the human mind and body. It is difficult to communicate tacit knowledge to others, since it is an analogue process that requires a kind of ‘simultaneous processing’.
  (Nonaka et al. 2000)

These descriptions can be summarised by stating that tacit knowledge resides in the mind of the knower, while explicit knowledge has been codified in some form.

Knowledge, however, may be categorised via two further dimensions (Bornemann et al. 2003):

- As knowledge is a prerequisite for human action, **knowledge psychology** differentiates between **declarative** and **procedural** knowledge. Declarative knowledge refers to facts and objects (“know-what”), while procedural knowledge concerns the way in which cognitive processes and actions are performed (“know-how”).
- Knowledge is linked to people, and a distinction can therefore be made according to the **knowledge holder**, with **individual** and **collective** knowledge identified as classes. Individual knowledge is held by one person and is therefore independent of context, while collective knowledge is intrinsically linked to a particular context or group of individuals.

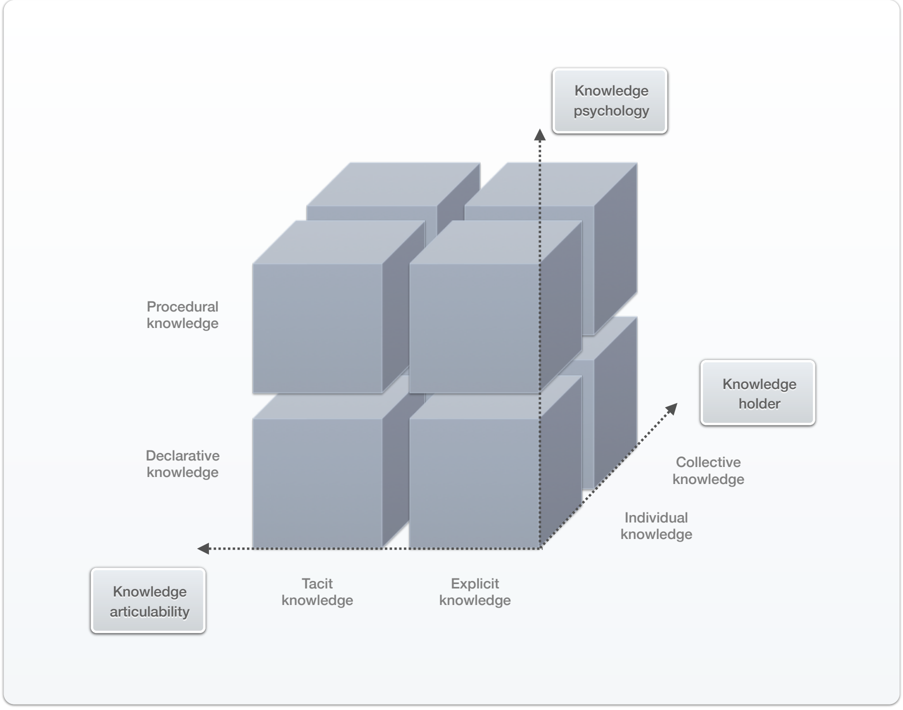


Figure 2.4: Knowledge categorisation framework

(adapted from Bornemann et al. 2003)

Bornemann et al. (2003) combine these different dimensions of knowledge classification into a knowledge categorisation framework, as illustrated in Figure 2.4.

2.3.3 Knowledge generation process

The ability to continuously generate new knowledge and thereby move on to new products and services is a key enabler of long-term competitive advantage (Schutte 2010). This ability to generate new knowledge is based on an organisational competence in knowledge-seeking and knowledge-creation (Seufert et al. 1999).

Recent developments in the ICT domain greatly improve the capture and distribution of explicit knowledge. These developments have skewed the approach to knowledge generation toward knowledge that is easy to articulate (refer to Figure 2.4). This is due to the significant challenges involved in exploiting tacit knowledge with its dependence on the knower and the context in which it was created (Orlikowski 2002).

Nonaka et al. (2000) outline a knowledge generation process that includes both explicit and tacit knowledge. This process consists of four modes for knowledge conversion through which tacit and explicit knowledge expands in both quality and quantity (refer to Figure 2.5). These four conversion modes are also the base for the name of the process, namely the SECI model.

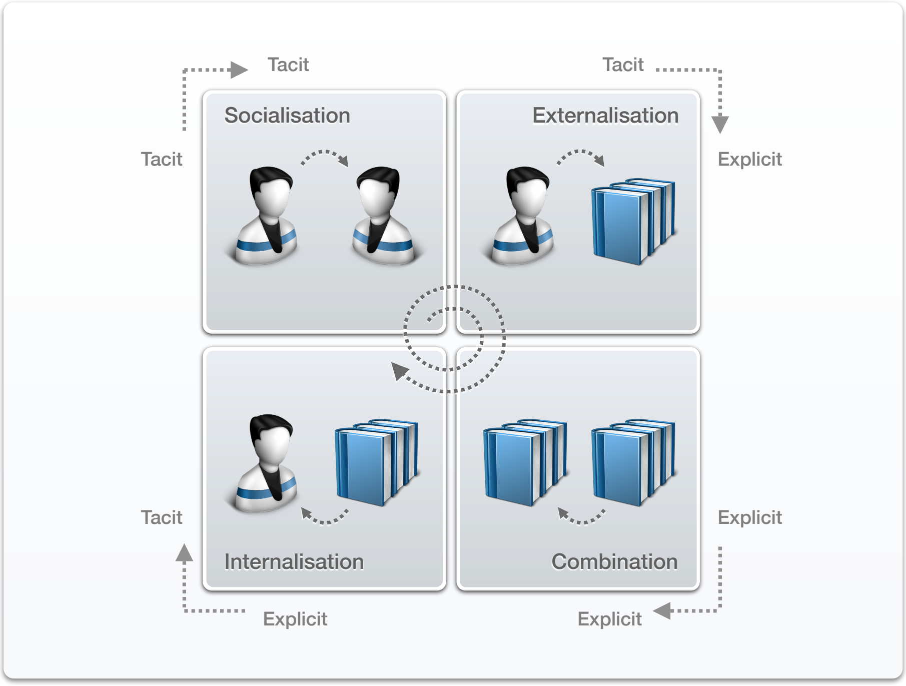


Figure 2.5: Knowledge conversion modes of the SECI model

(adapted from Nonaka et al. 2000)

The four conversion modes are (with reference to Nonaka et al. 2000 throughout):

- **Socialisation** is the process of generating new tacit knowledge through shared experiences. Tacit knowledge is difficult to formalise and is often context specific, and can therefore only be acquired through shared experience. This shared experience can be in the form of informal social meetings or by working together on projects, in which tacit knowledge such as world views, mental models and mutual trust can be created and shared. The socialisation process is not limited to organisational boundaries, but generally requires physical proximity, with organisations taking advantage of the tacit knowledge residing with customers or suppliers by interacting with them on a personal basis.

- **Externalisation** is the process of articulating tacit knowledge into explicit knowledge, thereby crystallising it and allowing it to be shared with others as the basis of new knowledge. During externalisation, tacit knowledge is expressed and translated with the help of tools that make it understandable to others, e.g. metaphors, analogies, concepts, diagrams, models and prototypes.

- **Combination** is the conversion of explicit knowledge into more complex and systematic sets of explicit knowledge. In order to achieve this, explicit knowledge is collected from both inside and outside the organisation and then combined, and edited or processed to form new knowledge. The combination of explicit knowledge is done through media such as documents, meetings and computerised communication networks, which enable the dissemination of new explicit knowledge among the members of the organisation.

- **Internalisation** is the process of embodying organisation-wide explicit knowledge into the tacit knowledge of the individual. This, however, requires that the individual should have the means to recognise explicit knowledge that is personally relevant to him or her. Internalisation is related to "learning by doing" and is used to broaden, extend and reframe the tacit knowledge of individuals that are part of the organisation.

Once explicit knowledge has been internalised by an individual, it becomes part of that individual's unique tacit knowledge base and is a valuable asset. This individual tacit knowledge can initiate a new spiral of knowledge creation when it is shared with others through socialisation or codified through externalisation (refer to Figure 2.6).

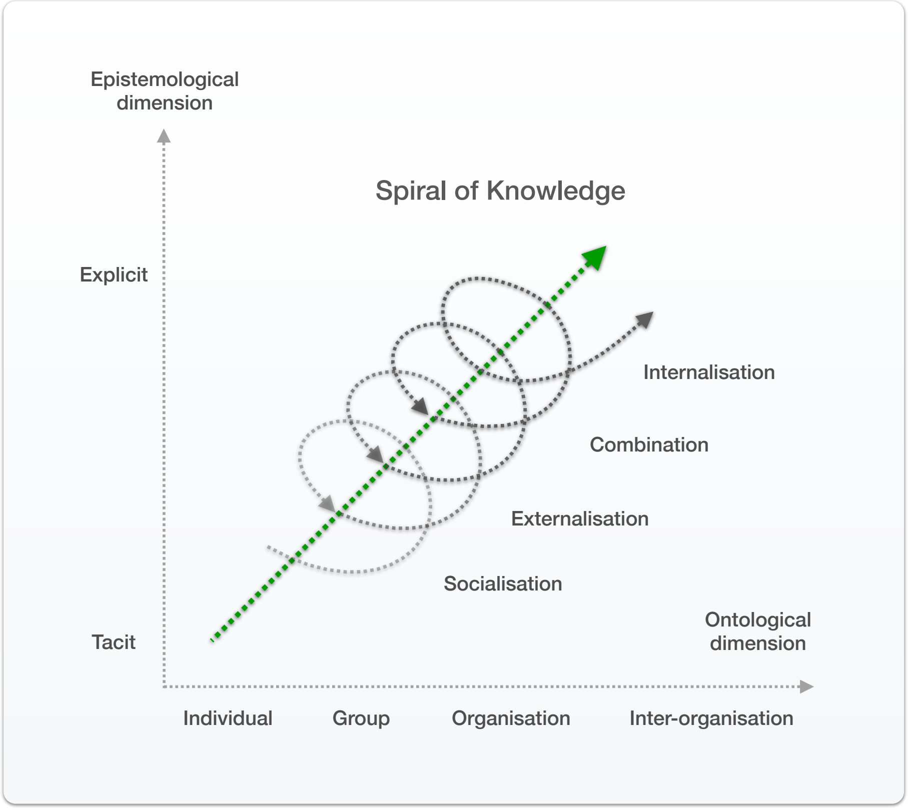


Figure 2.6: Knowledge generation process as a spiral

(adapted from Nonaka & Takeuchi 1995)

Nonaka et al. (2000), however, point out that progression through the knowledge generation process forms a spiral, and not a circle. In the spiral of knowledge creation, the interactions between tacit and explicit knowledge are amplified through the four conversion modes discussed above. This spiral therefore grows as it moves up through various ontological levels, with iterations of the spiral expanding horizontally and vertically across organisations.

The knowledge generation process starts at an individual level and expands as it moves through communities of interaction, transcending departmental and even organisational boundaries. The knowledge generation spiral therefore takes place both inside and between organisations, with knowledge being transferred beyond organisational boundaries, and knowledge originating in different organisations interacting to create more new knowledge (Badaracco 1991; Nonaka 1991; Wikström et al. 1994). Through dynamic interaction between organisations knowledge created by an organisation can trigger the mobilisation of knowledge residing with outside constituents such as universities, organisations that share the same domain, as well as customers (Nonaka et al. 2000).

The knowledge generation process itself therefore provides the foundation of the integrated knowledge network concept discussed in section 2.5.

2.3.4 Knowledge work processes

In order to enable the knowledge generation process described in the previous section, several authors identify a range of work processes that can be performed in the knowledge domain. These actions provide the mechanisms for the various knowledge conversions.

Back et al. (2005) categorise these processes as follows:

- **Locating** and **capturing** knowledge.
- **Transferring** and **sharing** knowledge.
- **Creating** knowledge.

The underlying principle of these categories, however, remains the **application** of existing or generated knowledge to create value. As it is not knowledge itself, but rather the organisation benefiting from knowledge that is managed, the application of knowledge is at the centre of these knowledge work processes (Back et al. 2005) (refer to Figure 2.7).

Most organisations start the evolution of their knowledge activities by focusing on locating and capturing the knowledge that already exists inside the organisation. Over time the focus, however, shifts to transferring and sharing this existing knowledge in order to create new knowledge assets. Finally, organisations that achieve high levels of maturity in their knowledge activities will create new knowledge, thereby achieving a distinct competitive advantage (Back et al. 2005).

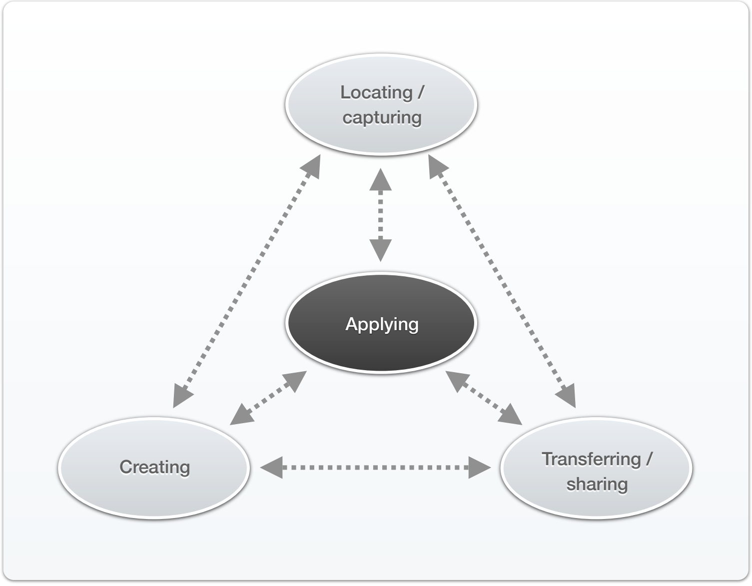


Figure 2.7: Knowledge work processes

(adapted from Back et al. 2005)

2.3.5 Knowledge-based network organisation

Bornemann et al. (2003) state that “the main requirement for effective knowledge management is an organisational framework that supports the optimal acquisition and networking of knowledge.” As previously discussed, knowledge is inherently linked to people and hence their location and proximity plays an important role in managing the organisation for effective knowledge work. In this sense, the view of the organisation corresponds to the traditional “topographical” perspective in which the organisation is described as the sum of its locations and departments (Buchanan 2003).

A different view of the organisation, however, is achieved when viewing it from the knowledge perspective. In this sense the organisation appears as a network of knowledge domains. In this view of the organisation, knowledge domains are seen as social systems that concern themselves with a common area of interest, acting as virtual departments (Castells 2010). These virtual departments, however, require the necessary organisational and technological support to overcome any form of dispersion that may hamper their knowledge work. Knowledge domains commonly span beyond the traditional boundaries of organisations, and may for instance include researchers from academia, as well as employees of customers and suppliers integrated in value chain processes (Bornemann et al. 2003; Wenger 1998).

2.3.6 Knowledge networks

The discussion of knowledge generation and knowledge work showed that significant utilisation and exploitation of knowledge are only enabled when networks are available to support it, and that networks are inherent to the nature of knowledge work.

When used in this context, the term ”networks” refers to the networks that exist between individuals, groups or organisations, as well as between collectives of organisations (Seufert et al. 1999). The social relationships between these actors can be described according to contents (e.g. products or services), form (e.g. duration) and intensity (e.g. frequency of communication) (Seufert et al. 1999). Such network relationships are, however, typically characterised by a mixture of form and contents, i.e. the relationships between actors are of various forms, and consist of diverse contents being exchanged (Seufert et al. 1999).

While ICT-based networks have proved to be valuable in the dissemination of explicit knowledge, they are a limited means for disseminating tacit knowledge (Nonaka 1995). The importance of human networks to enable knowledge diffusion and integration in knowledge intensive organisations have therefore been emphasised (Grant 1996; Liebeskind et al. 1996; Swan et al. 1999; Okhuysen & Eisenhardt 2002).

Seufert et al. (1999) subsequently define **knowledge networks** as:

> *A number of people, resources and relationships among them, who are assembled in order to accumulate and use knowledge primarily by means of knowledge creation and transfer processes, for the purpose of creating value.*

With regard to the development of knowledge networks, a differentiation is made between emergent and intentional networks (Seufert et al. 1999; Back et al. 2005). Intentional knowledge networks are seen as networks that are constructed from scratch, whereas emergent knowledge networks already exist but need to be cultivated in order to achieve high levels of performance. Knowledge networks are regarded as dynamic structures due to the continuous augmentation of knowledge through learning, and are supported and transformed by ICTs (Seufert et al. 1999).

Networks in general, including knowledge networks, can be classified at a high level according to their various purposes (Anklam 2007):

- **Personal growth and support networks** are informal and exist at an individual level. These personal networks are useful when we need, for example, advice, companionship and entertainment.
- **Idea networks** are based on creative exchange in which ideas build on each other, with no preconceived notions of what the outcome may be.
- **Learning networks** focus on growing the capacity of an individual or group in a particular area of skill or knowledge.
- **Mission networks** are generally directed at the social greater good with focus on arts and culture, education, environmental issues and religion commonly found.
- **Business networks** have goals linked to production and growth in terms revenue, profit and returns to stakeholders through growth in market reach, product offering and knowledge.

Apart from the general network purposes listed above, it has recently been noted that a class of networks have emerged that have the explicit purpose of fostering **innovation** between their members (Schutte 2010; Camarinha-Matos & Afsarmanesh 2012). These networks facilitate the exchange of various types of knowledge from different individuals, which is seen as a prerequisite for innovation (Eisenhardt & J. A. Martin 2000; Seufert et al. 1999). These networks commonly extend beyond the boundaries of single organisations in order to obtain access to knowledge from external sources (Powell et al. 1996; Chesbrough, Vanhaverbeke & West 2006b).

2.4 Innovation

This section presents fundamental concepts regarding innovation as context for the discussion of IKNs in section 2.5.

2.4.1 Innovation defined

Joseph Schumpeter first outlined the notion of innovation in his 1934 work entitled *The Theory of Economic Development*. He developed his ideas on innovation in various subsequent publications, but the concept remained vague and indeed still does to some extent today. Schumpeter (1939) defined innovation as encompassing an entire process, starting from a kernel of an idea, continuing through all the steps required to reach a marketable product that has an impact on the economy.

In the 21st century it has been widely acknowledged that the ability to innovate and generate new knowledge through the development of new products, services and processes is a key factor in the survival of organisations in the knowledge economy (Krogh et al. 2001; Drucker 2007). The term “innovation” is, however, still inappropriately and ambiguously used, and also commonly confused with other terms such as “invention” (Tidd & Bessant 2011). A wide range of definitions exist for innovation, with Salvendy (2001) providing the following comprehensive version:

> *Innovation is not just one simple act. It is not just a new understanding or the discovery of a new phenomenon, not just a flash of creative invention, not just the development of a new product or manufacturing process; nor is it simply the creation of new capital and markets.*
>
> *Rather innovation involves related creative activity in all these areas. It is a connected process in which many and sufficient creative acts, from research through service, are coupled together in an integrated way for a common goal.*

Essmann (2009) interprets this definition by stating that two characteristics of innovative initiatives emerge:

- A novelty or newness associated with innovation activities, and
- The presence of an inherent process.

The first of these aspects is discussed in section 2.4.2, which differentiates between different types of innovation based on the innovation subject, as well as section 2.4.3, which discusses versions of novelty. Sections 2.4.4 and 2.4.5 address the second aspect stated above by respectively discussing the innovation life cycle and various innovation process models. Section 2.4.7 focuses on the open innovation process model, and section 2.4.8 considers the Fugle innovation process model.

2.4.2 Types of innovation

Various types of innovation are defined and discussed in literature, including innovation in terms of, for example, products, services, processes, strategy, business models, marketing and value. Overlaps in the definitions of these types of innovation also regularly occur.

Schumpeter (1939) distinguished between five types of innovation and sources of competitive advantage, namely new products, new methods of production, new sources of supply, the exploration of new markets, and new ways to organise business.

This classification, however, may be simplified into three basic types based on the primary objective of the initiative (Hamel & Prahalad 1994):

- Product innovation.
- Process innovation.
- Strategy innovation.

It is, however, also possible that a given instance of innovation is best described as a combination of the above-mentioned types.

2.4.2.1 Product innovation

In this context the term “product” refers to any organisational output delivered, conveyed or served to a customer and may be tangible or intangible in nature (Essmann 2009). Innovations in services and product/service combinations are therefore also included in this category. Product innovations are furthermore not limited to those introduced to the market by commercial enterprises, with innovations in the offerings of non-profit organisations also included.

For a product initiative to constitute an innovation, the product itself does not need to be entirely new. Product innovation considered from the perspective of an enterprise encompasses a “change in, or an addition to the entities that comprise its product line” (Rothberg 1981). Product innovations are sources of competitive advantage by responding to an identified market need and claiming a portion of a current market, or by asserting an unidentified or untapped market (Essmann 2009).

2.4.2.2 Process innovation

The term “process” refers to “any course of action, procedure, technique, practice or modus operandi that can be established and executed within an organisation in an effort to transform or support the transformation of resources” (Essmann 2009). Such processes may be manual or automated, and of high-level managerial (Hamel 2006) or detailed operational nature.

As with product innovation, process initiatives need not be new in their entirety to constitute a process innovation, with changes in or additions to existing processes also included. Process innovations create competitive advantage in the form of organisational improvements regarding quality and time-to-market, and may or may not be readily perceivable to customers.

2.4.2.3 Strategy innovation

Strategy innovations impact the higher echelons of organisational governance, i.e. the positioning and direction of the organisation, including its mission and vision, policies and business models (Essmann 2009). Strategy innovation is also commonly referred to as business concept innovation (Hamel 1996; Baker 2002).

As with product and process innovation, strategic innovation need not be novel in its entirety and may include changes to existing strategies and business concepts. Strategy innovation creates competitive advantage through a new organisational direction and positioning that serves to create long-term differentiation, and/or result in the innovation of products or services (Essmann 2009). Such differentiation on a strategic level can create new markets, anticipate future markets, or revitalise old markets so that an organisation can pre-emptively position itself for competitiveness (Hamel 1996; Baker 2002).

2.4.3 Incremental vs. radical innovation

Of the various dimensions used to characterise innovation, a commonly accepted means is to place it on a scale ranging between the extremes of incremental and radical innovation (Hamel 1996; Cooper 1998; Baker 2002). This distinction focuses on the extent of the newness or novelty of the innovation, and can be made based on the “degree of strategic and structural change that the firm must undergo to accommodate the innovation in question” (Cooper 1998).

Innovations that represent a relatively small change in an existing product, process or strategy are considered to be incremental (Abernathy & Utterback 1978; Tushman & Anderson 1986). Basic improvements and optimisation associated with regular operational activities, however, are excluded, as they are not seen as novel.

Radical innovations are found at the other end of the newness spectrum (Abernathy & Utterback 1978; Tushman & Anderson 1986). Radical innovations represent “advances from the norm of such consequence” (Essmann 2009) that “revolutionary alteration of the organisation and its support networks must occur to accommodate and implement change” (Cooper 1998). Radical innovations can revolutionise established markets and spawn the growth of new markets (Tidd & Bessant 2011).

2.4.4 Innovation life cycle

Earlier reference was made to the distinction between invention and innovation (refer to section 2.4.1), without a definition for invention being provided. Having discussed some aspects of innovation, the following high-level definitions are in order: invention is the generation of newness or novelty, while innovation extends to the derivation of value from that novelty (Szmytkowski 2005).

Berth (1993) presented evidence that few inventions, however, reach the point where they are regarded as innovation. This effect leads to the innovation funnelling paradigm (refer to Figure 2.8), in which the number of ideas successfully commercialised is significantly lower than the original number of ideas. According to Fagerberg et al. (2005), this presents evidence that innovation may be characterised by a process that bridges the gap between invention and innovation.

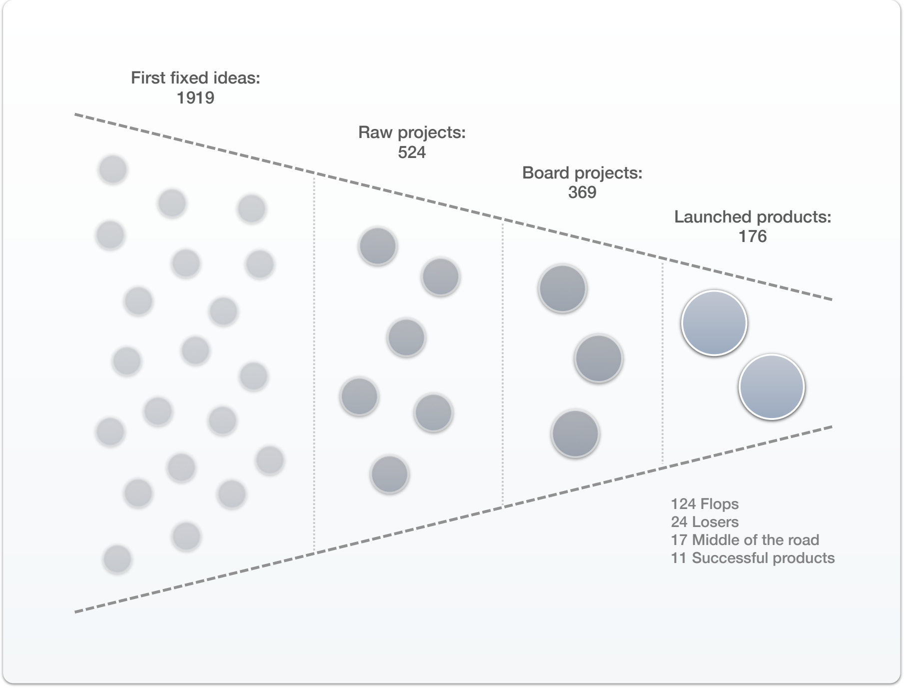


Figure 2.8: The innovation funnelling paradigm

(adapted from Berth 1993)

By definition, a process requires time, resources, capabilities, knowledge, and structure to be executed and to ensure sound output (Essmann 2009). A process may furthermore be represented to a life cycle of phases requiring execution, all of which should be executed in order to ensure the desired output. A basic and generic representation of the innovation life cycle includes the following phases: invention, feasibility, implementation, operation and disposal (Van Zyl 2006) (refer to Figure 2.9).

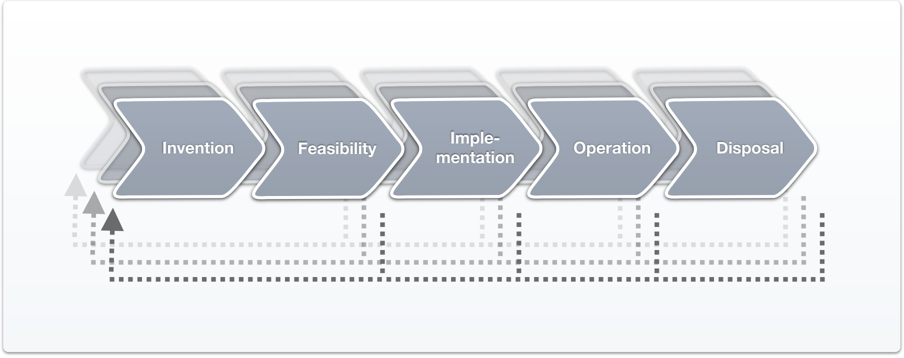


Figure 2.9: The basic innovation life cycle

(adapted from Essmann 2009)

The invention phase includes opportunity identification and idea generation, supported by general creative activity. The idea that triggers the innovation may relate to products, processes and/or strategy. The feasibility of this idea then needs to be determined through rigorous testing and screening. The specification, functional analysis and initial design of the innovation are also performed in this phase. The detailed design and manifestation of the innovation occurs in the implementation phase.

The operation phase is initiated once a commercially viable output has been delivered, and includes activities such as production and quality control, and the monitoring and optimisation of processes. Once the desired utilisation of the innovation is achieved, the innovation life cycle draws to a close during the disposal phase. This marks the conclusion of the innovation, and focuses on reflecting and learning from the process and fulfilling final (e.g. legal and environmental) obligations (Essmann 2009). This disposal, however, does not refer to the conclusion of the innovation process, but rather to the conclusion of a particular innovative initiative. The innovation process should continue in order to sustain and exploit and competitive advantage that results from previous initiatives (Moore 2008).

Learning forms an integral part of the activities in all innovation life cycle phases, and there are opportunities to learn from the successes and failures of each phase at its conclusion. The innovation life cycle is furthermore of such a nature that phases may be revisited in order to re-execute certain activities or to refine certain aspects of the innovation (Essmann 2009). Multiple innovation life cycles may be executed concurrently in an enterprise, in line with the definition of innovation provided by Salvendy (1992), which states that innovation constitutes “many creative acts”.

The innovation life cycle discussed in this section represents a basic and generic depiction of the activities involved in innovative initiatives, which allows for the development of innovation process models. The dynamics and complexities of innovation, however, do not allow for a full discussion in this study. It should therefore be noted that the description of the innovation life cycle provided in this section is a over-simplification of the phenomenon. In reality, the complexity involved in executing the innovation process and the delicate support systems and structures necessary to facilitate the process, do not permit a linear approach (Essmann 2009).

2.4.5 Innovation models

Innovation models generally describe an innovation process including four main phases that have to be managed in order to turn ideas into reality (Perry & Uys 2010; Tidd & Bessant 2011):

1. **Scanning:** Scan and search the environment (both internal and external) to identify and process opportunities for innovation.
2. **Selecting:** Strategically select those things from the set of potential triggers for which the organisation will commit resources to turn into reality.
3. **Planning:** Having chosen an option, resources must be provided, either through research and development, or acquisition, and allocated.
4. **Implementing:** The innovation has to be implemented, growing it from an idea through various stages of development to the final launch, either as a new product or service in the external market place or a new process or method within the organisation.

Various models have been developed to analyse and understand the nature of the innovation process beyond the simple (and incomplete) terms listed above, each focusing on different areas that were dominant in the period during which they were conceived. Rothwell (1992) organised these innovation models into five generations, with the addition of sixth and seventh generations proposed by Du Preez and Louw (2008) (refer to Table 2.2).

The first (1930s) and second (1960s) generation innovation models are linear and explain innovation being either pushed by progress in science and technology, or pulled by market needs. The third generation models date from the 1970s and feature coupling, which recognises the simultaneous influence of both technological capabilities and market needs. Although these coupling models contain feedback loops, they remain sequential models with limited functional integration (Du Preez & Louw 2008).

To improve on the lack of functional integration in the earlier linear models, the fourth generation innovation models (1980s) introduces an interactive approach. This approach views the innovation process as parallel activities across organisational functions, but still does not explain the entire innovation process (Du Preez & Louw 2008).

Table 2.2: Seven generations of innovation models

(Du Preez & Louw 2008, adapted from Rothwell 1992)

Generation	Model	Characteristics
First	Technology push	Simple linear sequential process, emphasis on R&D and science.
Second	Market pull	Simple linear sequential process, emphasis on marketing; the market is the source of new ideas for R&D.
Third	Coupling models	Recognising interaction between different elements and feedback loops between them, emphasis on integrating R&D and marketing.
Fourth	Interactive models	Combinations of push and pull models, integration within firm, emphasis on external linkages.
Fifth	Network models	Emphasis on knowledge accumulation and external linkages, systems integration and extensive networking.
Sixth	Open innovation	Internal and external ideas, as well as internal and external paths to market can be combined to advance the development of new technologies.
Seventh	Integrated knowledge networks	Combining network models and open innovation.

The fifth generation innovation models depicted in Table 2.2 originated in the late 1980s to early 1990s, and describe innovation networks. The concept of innovation networks as a class of CNOs (refer to section 2.2) have, however, only in recent years been subjected to more intense academic scrutiny (Berasategi et al. 2011). These models show an evolution in the nature of the innovation process towards complex models that require high levels of integration, both on intra- and inter-organisational levels (Lundvall 1988; Freeman 1991; Nohria & Eccles 1992; Rothwell 1992). More recently, Chesbrough (2003) complemented this vision by coining the term “Open Innovation” describing the systematic integration of external inputs at different stages of the innovation process (refer to section 2.4.7). Some scholars argue that in some sense Open Innovation represents a sixth generation innovation model, as it clearly departs from previous closed models (Du Preez, Louw & Lutters 2008a).

The networks described by the fifth generation innovation models are discussed in more detail in the next section. This is followed by a discussion of open innovation as described by sixth generation models in section 2.4.7. The Fugle innovation model discussed in section 2.4.8 may be seen as a seventh generation innovation process model. This leads up to the discussion of IKNs in section 2.5.

2.4.6 Innovation networks

While no unanimously accepted reference framework for innovation networks has yet been achieved (Oliver & Ebers 1998; Berasategi et al. 2011), important advances have been made in characterising the networked innovation process (Pittaway et al. 2004). The informal and multi-dimensional nature of the process has been highlighted (Kalthoff et al. 1998), with various of its social aspects also gaining attention (Pyka 2002; Taatila et al. 2006). There is consensus that innovation networks serve to promote creativity, increase the capacity for invention and act as a catalyst for innovation (Powell et al. 1996; Hagedoorn 2002; Prahalad & Ramaswamy 2004b; Berasategi et al. 2011; Galanakis 2006; Vanhaverbeke & Cloodt 2006).

There have been many attempts to categorise innovation networks. Dussauge et al. (1992) performed an early classification according to strategic objectives, with Tidd (2006) proposing the identification of types of innovation networks according to radicalism of innovation and partner similarity. A more recent classification that sheds light on the development of a networked innovation model is based on criteria related to the sources of innovation (Berasategi et al. 2011). In this schema, there can be distinguished between three types of collaborative innovation:

- External collaborations with other organisations (Gulati et al. 2000; Contractor & Lorange 2002; Dilk et al. 2008).
- Co-innovations through customers (Cox & Mowatt 2004; Surowiecki 2005; Von Hippel 2005).
- Collaborations via innovation market platforms (“innomediaries”) (Huston & Sakkab 2006; Piller 2008).

Various approaches to the management of these networks have been documented, e.g. Jones et al. (1999), Nooteboom (2000), Bullinger et al. (2004), Douthwaite (2006), Dooley and O’Sullivan (2007) and Cowan et al (2007). Chesbrough and Prencipe (2008) propose an approach in which the dynamics of technology development reflects the dynamics of an inter-organisational network. In this approach the innovation network is managed in an explorative mode during the early phases of technology development, with the emphasis on building connections with research centres and universities. Once the interactions between component types of the new technology are better understood, codified, modularised and shared, more exploitative networks featuring suppliers and customers may be better suited to exploit the current technology. The transition phase between these explorative and exploitative modes is well suited to building ties with start-ups and new entrants, as these firms experiment with alternative design configurations that exploit the underlying technology (Chesbrough & Prencipe 2008).

Despite this contribution by Chesbrough & Prencipe (2008), most efforts to describe the management of innovation networks are conceptual and descriptive, rather than being based on experience. This is indicative of the emerging nature of the domain of innovation networks. There is a significant lack in literature regarding holistic, systematic and integrated models. Berasategi et al. (2011), however, does make a contribution in this regard by proposing a comprehensive framework for collaborative networked innovation that features an activity model and an actor model.

2.4.7 Open innovation

Chesbrough (2003) defines open innovation as follows:

> *Open Innovation means that valuable ideas can come from inside or outside the company and can go to market from inside or outside the company as well. This approach places external ideas and external paths to market on the same level of importance as that reserved for internal ideas and paths to market during the Closed Innovation era.*

Models depicting open innovation (refer to Figure 2.10) emerged during the early 2000s with the advent of Web 2.0 technologies and were already in use in various major enterprises before being studied academically. Lafley and Charan (2008) note that it was envisioned in 2001 at Proctor & Gamble that roughly half of the organisation’s new ideas would come from outside the organisation. By 2006 Huston & Sakkab confirmed that this figure had indeed grown to above 35% (Huston & Sakkab 2006).

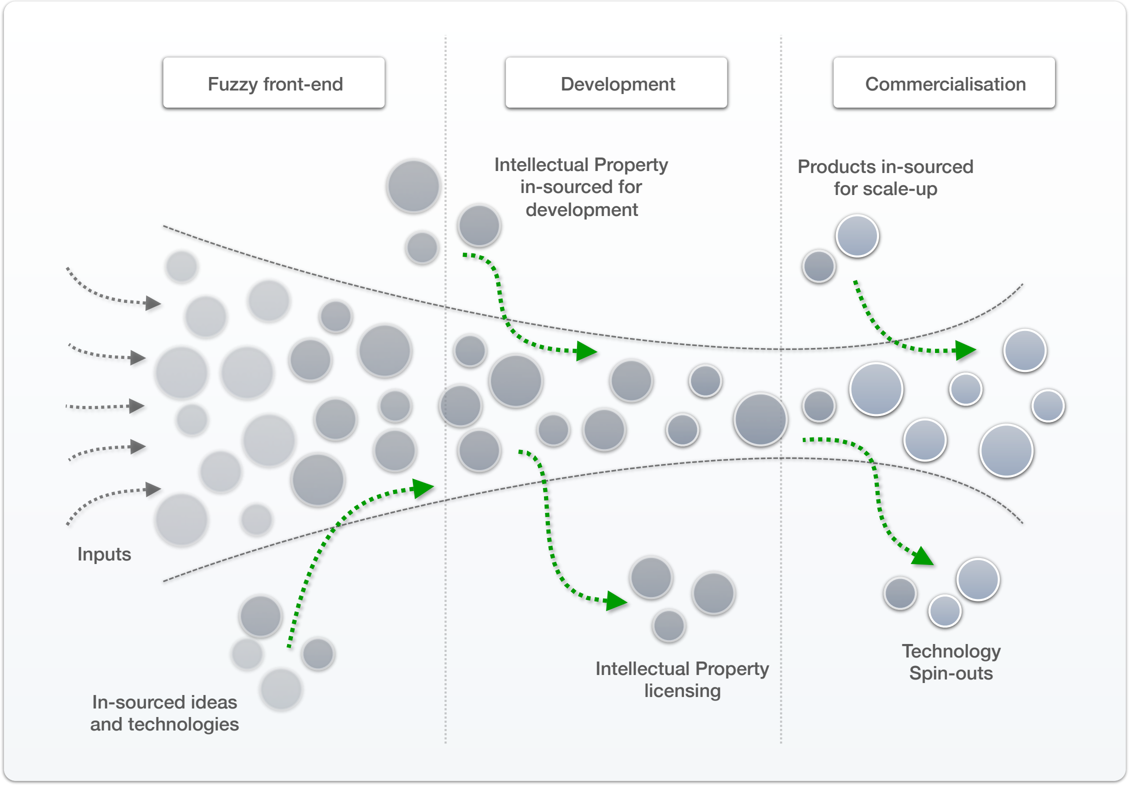


Figure 2.10: Open innovation model

(adapted from Docherty 2006)

The above definition of open innovation points out that open innovation clearly stands in contrast to closed innovation, in which an organisation only relies on internal R&D, idea generation and problem solving. Conceptually, Open innovation implies a methodology and mindset in which an organisation has well-defined innovation structures, and makes use of individuals and/or organisations outside the organisation's boundaries to contribute to these structures (Marais 2010).

An obvious benefit of open innovation is the much larger base of ideas and technologies from which the enterprise can draw to drive internal growth (Docherty 2006). Leading companies have also identified open innovation as a strategic tool to explore new growth opportunities at lower risk levels (Docherty 2006). Open innovation has changed the innovation environment by adding a new dimension to the networking and collaboration introduced by the earlier network models (refer to section 2.4.6). Open innovation calls for a new logic, which puts openness and collaboration right at the centre, enabled by networked and web communities.

2.4.8 Fugle innovation model

Du Preez & Louw (2008) contributed an innovation model that not only focuses on the innovation funnel, but also includes the implementation, operation and exploitation phases of innovation initiatives. The model revolves around a generalised innovation process that combines the convergent innovation funnel (refer to section 2.4.4) with a divergent exploitation and deployment "bugle". The "funnel" and "bugle" terminology of these concepts are combined to name this model, namely the Fugle innovation process model (Du Preez & Louw 2008) (refer to Figure 2.11).

The innovation process described by the Fugle model operates internally in the enterprise, but with all stages of the process linked to the external environment. This emphasises the networking aspect, especially with regard to open innovation, as all stages could have external influence up to the point of being collaboratively executed by a network of participants. The innovation process is guided and supported at the top by the enterprise's strategies, its people and culture, organisational structure and processes, as well as information and knowledge (Du Preez & Louw 2008). The innovation process described by the Fugle model consists of a number of stages, includes several gates and filters, and a number of iterative loops are also possible. The main stages of the innovation process described in the model are:

- Idea generation/identification,
- Concept definition,
- Concept feasibility and refinement,
- Portfolio management,
- Deployment,
- Refinement and formalisation, and
- Exploitation.

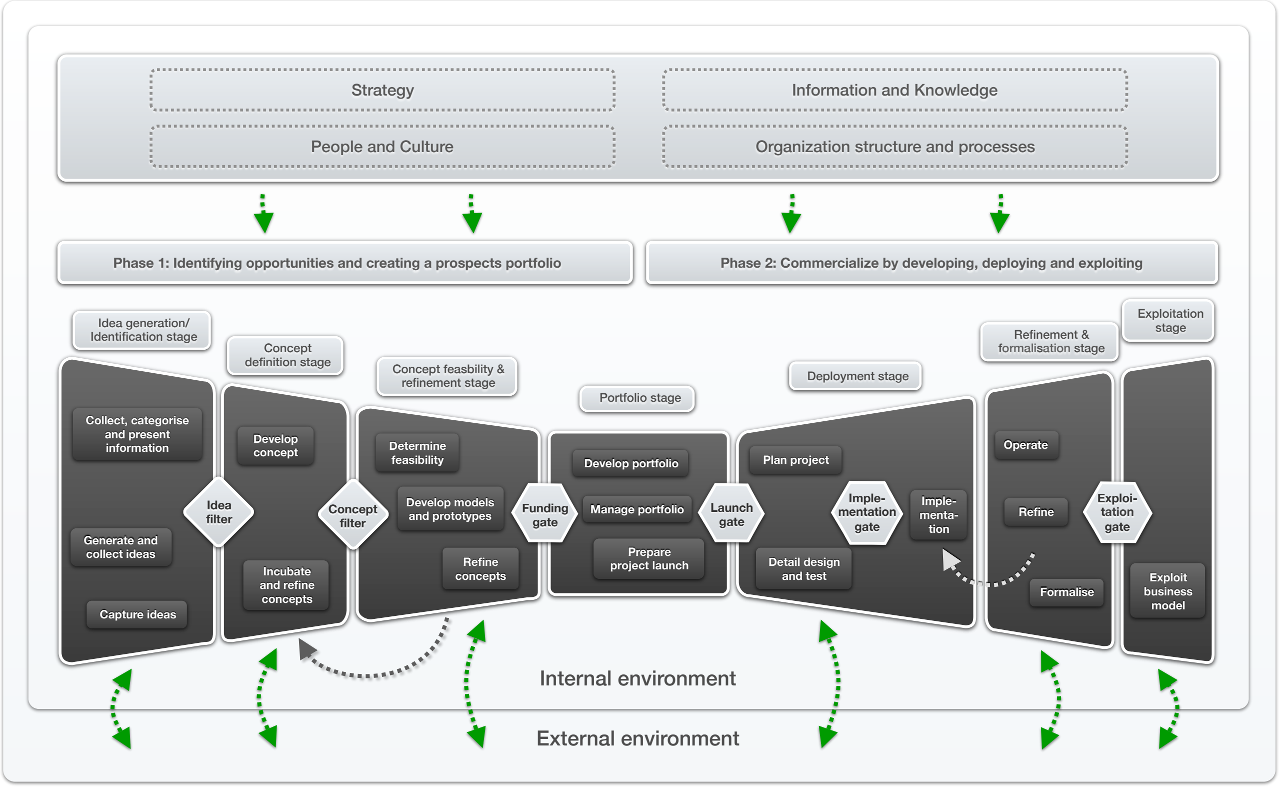


Figure 2.11: Fugle innovation process model

(adapted from Du Preez & Louw 2008)

Although the fifth and sixth generation innovation models (refer to sections 2.4.6 and 2.4.7) are still relatively new, it may be argued that the Fugle model may already be identified as a seventh generation innovation model. Seventh generation innovation models emphasise open innovation within the fifth generation network models to describe the integrated environment in which innovation has been executed in recent years (Du Preez, Louw & Lutters 2008a; Schutte 2010).

2.5 Integrated knowledge networks

When seventh generation innovation models such as the Fugle model (refer to section 2.4.8) are implemented, they are supported by integrated knowledge networks (IKNs), which exploit the concepts of open innovation (Du Preez, Louw & Lutters 2008a; Schutte 2010). The current understanding and rationale for IKNs is discussed in greater detail in this section.

2.5.1 Knowledge and innovation

This section discusses the relationships between knowledge and innovation as context for the definition of IKNs in section 2.5.4.

Research increasingly highlights the fact that knowledge management is a key requirement for effective innovation management (Johannessen et al. 1999; Pérez-Bustamante 1999; Carneiro 2000; Burgelman et al. 2009; Cowan et al. 2007; Pittaway et al. 2004). How knowledge is used, spread and stored by an organisation's employees determines whether this organisation has a culture of stimulating or restraining innovation (Schutte & Du Preez 2008).

Innovation is achieved through the novel exploitation of knowledge that resides both internal and external to the organisation (Fischer & Varga 2002; Leiponen 2006). Therefore, in order to innovate effectively and sustainably, existing knowledge should not only be captured, but also shared and integrated in context (Clark & Fujimoto 1991; Iansiti & Clark 1994; Okhuysen & Eisenhardt 2002). Through knowledge sharing, inefficient redundancies in learning and innovation may be alleviated (Szulanski 1996; Erickson & Jacoby 2003), while knowledge integration allows for the exploitation of commonalities between explicit knowledge assets (Henderson & Cockburn 1996; Chesbrough, Vanhaverbeke & West 2006a; Von Hippel 2005).

However, to achieve sustainable competitive advantage, the knowledge-based activities and resources must be heterogeneous and not easily replicable by competitors (Henderson & Cockburn 1996). Innovation is a key process that underlies the creation of these kinds of unique capabilities to support sustainable advantage, and knowledge integration that enables innovation is therefore of utmost strategic importance (Leiponen 2006; Esterhuizen et al. 2012).

Only focusing on explicit knowledge and thereby taking a rather narrow view on innovation involves the danger of erecting various kinds of barriers to innovation (Seufert et al. 1999). Knowledge networking, however, provides a potential solution to these barriers by cross-linking islands of knowledge to stimulate the evolution, dissemination and application of knowledge (Swan et al. 1999). This integrated approach to knowledge management yields great benefits, with the openness and richness of networks believed to foster a fertile environment for the creation of new knowledge, while accelerating the rate of innovation (Seufert et al. 1999). Furthermore, Powell et al. (1996) demonstrated a ladder effect, showing that successful organisations position themselves at the centre of overlapping networks, thereby stimulating valuable research collaborations with their various partner-organisations.

Seufert et al. (1999) make a compelling case for the transformative effects of knowledge networking on participating organisations. They state that "the presence of a dense network of collaborative ties may alter participants' views on competition". Inside a densely connected environment, organisations must adjust to a novel perspective in which it is no longer necessary to have exclusive ownership of an asset in order to profit from it. In a particularly apt analogy, Seufert et al. (1999) state that "[s]ince a competitor on one project may become a partner on another, the playing field resembles less a horse-race and more a rugby match, in which players frequently change the color of their jerseys."

A further facet of the correlation between knowledge and innovation is described by Du Preez et al. (2006) in a discussion of the correlation between the knowledge life cycle and the innovation life cycle (refer to Figure 2.12).

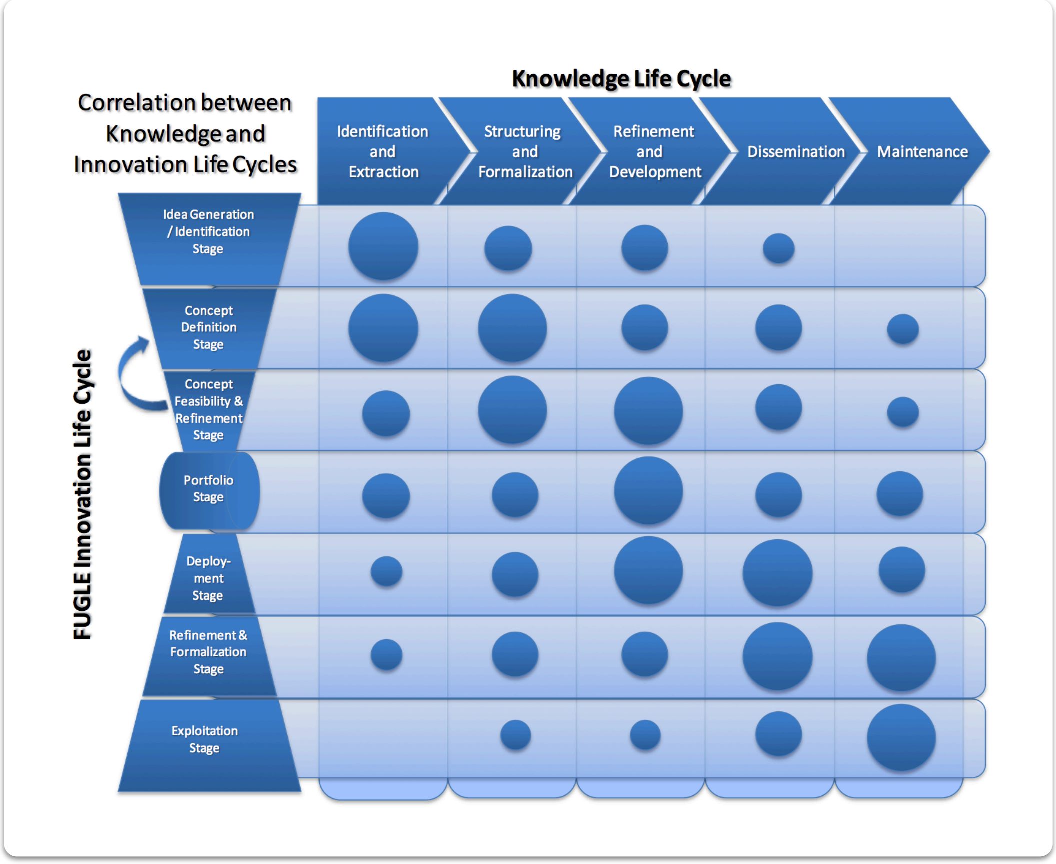


Figure 2.12: Correlation between innovation and knowledge life cycles

(Schutte 2010; adapted from Du Preez et al. 2006)

The knowledge life cycle consists of the following phases (Du Preez et al. 2008b):

- Identification and extraction – knowledge is identified and extracted from other sources.
- Structuring and formalisation – knowledge is structured and formalised in the selected knowledge management tools.
- Refinement and development – knowledge is analysed, refined and further developed.
- Dissemination – the distribution of applicable knowledge to the people that require it.
- Maintenance – maintaining the knowledge, to ensure it remains up to date and applicable to the domain.

An innovation project will typically incorporate more than one knowledge life cycle. During such a knowledge life cycle, knowledge is repeatedly captured, refined, disseminated and maintained, depending on the progress and success of each phase of the innovation project.

Given the above arguments, there is consensus that sustainable innovation is dependent on innovators' ability to use knowledge management tools and techniques to perform the following activities (Perry & Uys 2010):

- Analyse market needs, trends and opportunities.
- Capture the outputs of innovation projects to preserve "corporate memory" for analysis and future use.
- Re-use the outputs from previous projects or other groups, to accelerate the current innovation efforts with the co-operative knowledge captured before.
- Link innovation project members together and collaborate with other groups so as to expand the participating community, therefore expanding the ability to learn from others and innovate faster.

2.5.2 Learning and innovation

Following the discussion of the relationships between knowledge and innovation in the previous section, this section presents the theoretical case for inter-organisational knowledge sharing in order to enable innovation. This concept serves as context for the discussion of the knowledge supply chain in section 2.5.3.

Early generations of innovation process models (refer to section 2.4.5) describe the process as a linear journey from research to the market, with a clear beginning and end. The model of the innovation process contributed by Kline and Rosenberg (1986), however, shows an iterative nature and considers innovation to be a learning process between research activities performed both inside and outside the boundaries of the knowledge-creating organisation (refer to Figure 2.13). This process accesses diverse scientific, technological and commercial knowledge assets as needed (Pérez-Bustamante 1999).

The links between these knowledge-creation activities and the various areas in the organisation are activated in different phases of the innovation chain. The process may in fact start at any point of the innovation chain and gains insight and incorporates the advances in scientific research, commercial and market information and knowledge generated in the production area through learning-by-doing, before it reaches the market - if at all (Pérez-Bustamante 1999). This iterative innovation model shows an appreciation for the multitude of inter-connections among all the activities of the innovation process, and that diverse knowledge bases contribute to a successful market implementation of a given innovation (Pérez-Bustamante 1999; Schutte 2010).

The synthesis of scientific and technological knowledge, insights gained through market research, as well as tacit, experiential knowledge available in the enterprise, drive the central innovation chain (Bogers & West 2012) (refer to Figure 2.13). This chain covers the phases of invention, innovation and diffusion of technological knowledge, and incorporates feedback from the various activities (Pérez-Bustamante 1999).

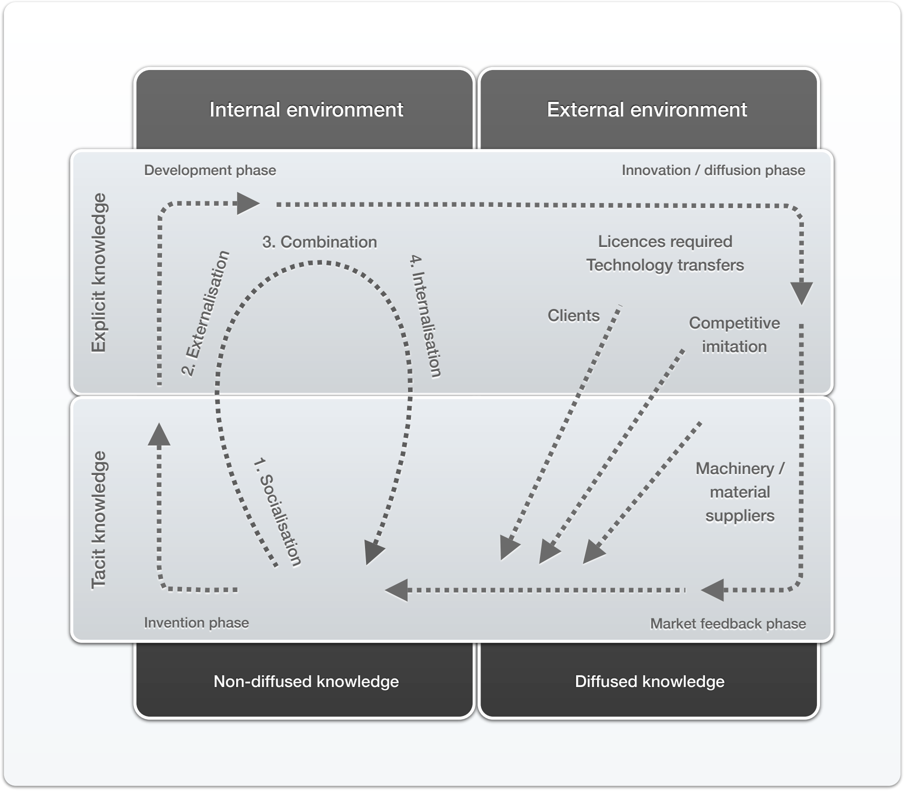


Figure 2.13: Framework model for knowledge innovation

(adapted from Kline & Rosenberg 1986)

In addition to the partial feedback between activities in the innovation chain, the commercial diffusion of an innovation provides market feedback to the stakeholders in the innovation process. This feedback helps to identify opportunities for further research aiming to enhance those characteristics of the innovation that may better satisfy the diverse market needs (Pérez-Bustamante 1999). In this sense, the market enables the creativity of the innovators involved.

Pérez-Bustamante (1999) describes the innovation process as a “locus of learning with outcomes varying according to the phase in which the learning activity is undertaken.” In the first part of the innovation model, the “invention locus”, learning processes imply the acquisition of knowledge associated with natural laws. Despite this initial scientific focus, the purpose of the learning process, however, remains to obtain commercial products based on the generated knowledge. The development area features innovation and research processes undertaken to discover the desired characteristics of the innovation that will satisfy market expectations, and will facilitate the production process. These development activities, however, are only brought to fruition through access to commercial knowledge that incorporates the design of the innovation with the characteristics that are most valued and demanded by the market (Pérez-Bustamante 1999).

To culminate these development activities, it is essential to have access to commercial knowledge reservoirs or research activities, since they will incorporate the product design activities information about those characteristics that are most valued and demanded by the market. This, however, results in an organisational demand for specialised individuals who are capable of assessing and valuing the knowledge which is present in the scientific, engineering, managerial and commercial milieus.

Pérez-Bustamante (1999) summarises the above description of innovation by stating that “innovation is a learning process that takes place between scientific research and the market, through which the organisation uses scientific and engineering knowledge bases to develop products with the characteristics demanded by the market.”

2.5.3 Knowledge supply chain

This section discusses the knowledge supply chain as a precursor for the definition of IKNs in section 2.5.4.

The previous sections established a view on innovation in which it is considered to be a flux of knowledge, characterised by a continuous flow of information within innovative activities that are executed either internally or externally to the enterprise. This consideration supports the existence of an important correlation between innovation and knowledge management (Pérez-Bustamante 1999).

Du Preez et al. (2008) identify a number of the role players in this view on innovation by characterising a knowledge supply chain. This knowledge supply chain is equivalent to a knowledge generation value chain in which learning occurs between various domains (Gemünden et al. 1992; Gemünden et al. 1996; Geels 2004; Chesbrough & Prencipe 2008). It indicates how discovering new knowledge, making the knowledge transferable (from tacit to explicit), transferring that knowledge through documentation and from person to person, and finally applying that knowledge, all support innovation in the material supply chain (refer to Figure 2.14).

Some interpretations of the effects of the knowledge supply chain are highlighted (Du Preez et al. 2008a):

- The combination of public and private domain information result in an abundance of knowledge.
- The innovation process that must support the material supply chain is much too complex to be addressed by a single team in a single organisation.
- The extensive interaction between public domain activities and private domain development work is imperative.

The benefits of collaboration and networking between various domains to enable knowledge creation and transfer have been discussed in previous sections (refer to section 2.3.6). The importance of a holistic approach to knowledge networking that includes both explicit and tacit knowledge was stressed. The

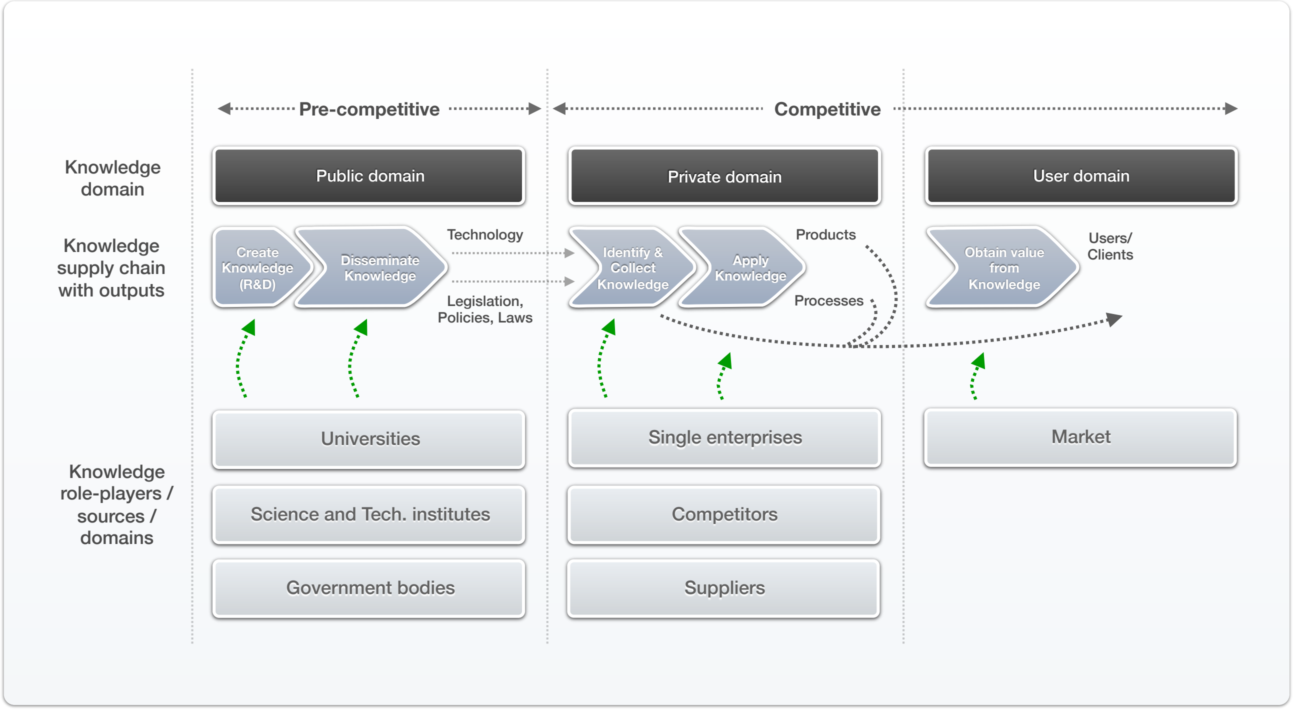


Figure 2.14: Components of the knowledge supply chain

(adapted from Du Preez & Louw 2007)

benefits of open innovation, which allows for both internal and external ideas paths to market, were also emphasised.

Applying these principles to the challenge of managing the knowledge supply chain, provides the foundations for the concept of an integrated knowledge network (IKN).

2.5.4 Definition of IKNs

This section defines IKNs, the primary organisational subject of this study.

Du Preez et al. (2008) define an IKN as follows:

> *A Knowledge Network signifies a number of people and resources, and the relationships between them, that are able to capture, transfer and create knowledge for the purpose of creating value. An* ***Integrated Knowledge Network*** *spans all domains, communities, and trust relationships with the goal of fostering sustainable innovation that will continue to promote the competitiveness of its users.*

Open innovation assumes that organisations in a shared domain have a joint need to innovate, and that they therefore share their innovative abilities toward a common cause. This emphasises the inter-

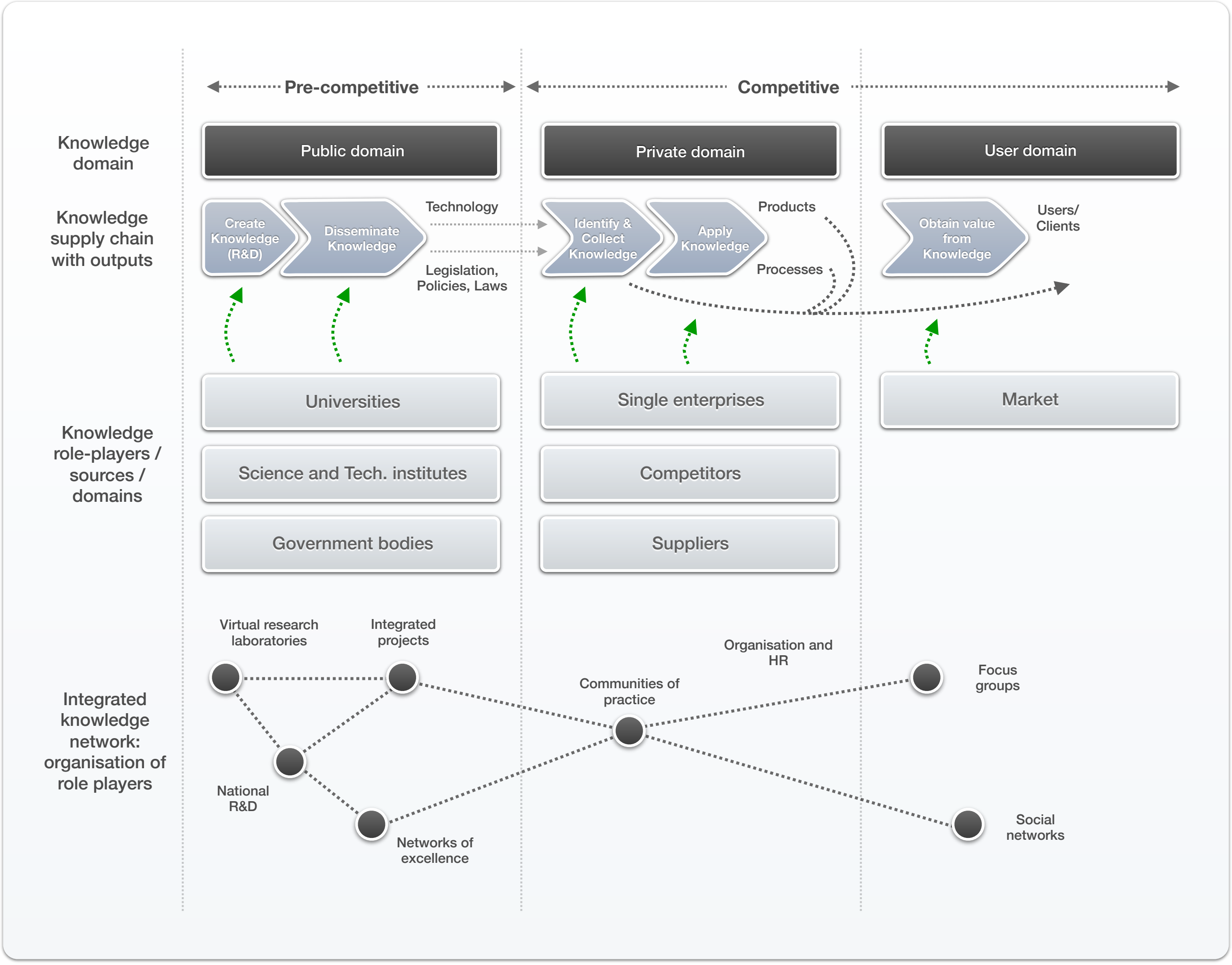


Figure 2.15: The components of an IKN

(adapted from Du Preez & Louw 2007)

organisational aspects of knowledge networking and is the main distinguishing characteristic of an IKN (Schutte 2010) (refer to Figure 2.15).

An IKN establishes the essential links between the market and institutions that focus on R&D that is required in the learning view on innovation (Pérez-Bustamante 1999). An IKN furthermore exploits the diverse capabilities of its constituent members to supply the human resources required to assess and value the knowledge that is present in the scientific, engineering, managerial and commercial milieus that exist along the knowledge supply chain.

The main conceptual aspects of an IKN are (Schutte 2010):

- A joint research interest,
- Inter-organisational collaboration,
- Formal alignment of systems and processes, and

- Knowledge networking.

These aspects represent a functional view on IKNs, which is expanded upon in the next section through the existing research on knowledge networks.

2.5.5 Functional view on IKNs

This section expands on the introduction to knowledge networks that was provided in section 2.3.6 in order to complete the functional view on IKNs.

IKNs are viewed as inter-organisational manifestations of knowledge networks, with the purpose of fostering innovation between network stakeholders. It may therefore be argued that existing research on the functional view of knowledge networks also mostly apply to IKNs.

2.5.5.1 Framework for knowledge networks

Seufert et al. (1999) proposed a framework for knowledge networks comprising the following components:

- Actors (individuals, groups and organisations),
- Relationships between actors, which may be characterised by form, content and intensity,
- Resources, which may be used by actors within their relationships, and
- Institutional properties (structure, culture, rules, processes and communication plans).

The framework furthermore consists of three building blocks (Seufert et al. 1999):

- **Facilitating conditions:** These are the network's internal structural and cultural dimensions in which knowledge work processes take place. Facilitating conditions define the enabling or inhibiting environment for knowledge creation and transfer, with the organisational structure, management systems or network culture being aspects to take into account.
- **Knowledge work processes:** These are social interaction and communication processes on an individual and group level, which advance knowledge evolution to an organisational and inter-organisational level. These processes can be conceptualised as a knowledge spiral consisting of dynamic conversion actions between explicit and tacit knowledge (refer to section 2.3.3, Figure 2.6). Knowledge networks can furthermore be characterised according to the most prolific knowledge work processes as either experiencing (where socialisation is dominant), materialising (externalisation), systematising (combination) or learning (internalisation) networks.
- **Knowledge network architecture:** The tool set used in social relationships, including both organisational tools and information and communication tools. These tools go beyond a collection of modular tools to being integrated as "solution frameworks" with the intention of linking architectural designs at both organisational and ICT level to support knowledge networking.

2.5.5.2 Methodology for IKNs

Schutte (2010) contributed a methodology for the development of IKNs, incorporating concepts from Back et al. (2005) and Anklam (2007). A high-level version of this methodology is illustrated in Figure 2.16.

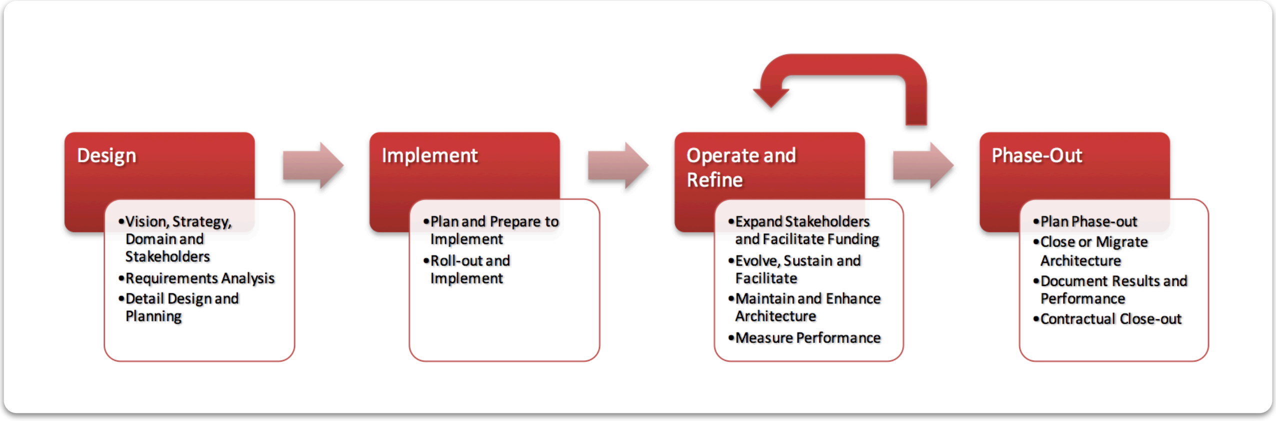


Figure 2.16: Methodology for IKNs

(Schutte 2010)

The methodology represents actions to be performed throughout the life cycle of knowledge networks and consists of four main phases, each including a number of sub-processes:

- **Design** includes the identification of a vision, strategy, domain and inter-organisational stakeholders, requirements analysis, and detail design and planning.
- **Implementation** includes planning and preparing the implementation, as well as the eventual network roll-out and implementation.
- **Operation and refinement** include the expansion of inter-organisational stakeholders and the facilitation of funding, the evolution, sustenance and facilitation of network activities, the maintenance and enhancement of the network architecture, and performance measurement.
- **Phase-out** includes planning of the phase-out, closure and migration of the network architecture, documentation of results and performance, and contractual closeout.

This methodology is a significant enhancement compared to other methodologies in that it enhances the inter-organisational aspects of other methodologies and addresses various other weaknesses (Schutte 2010) (refer to sections 2.5.5.3 and 2.5.5.4).

2.5.5.3 Facilitating conditions and key success factors

The performance of a knowledge network and the assessment of its value is a complex matter due to the number of impacting factors both inside and outside the community (Raimann 2000). These impacting factors serve as enablers or motivators when present and as inhibitors or barriers when they are not. Thus, if these factors are insufficiently fulfilled or not present at all, the performance of the knowledge network will most likely be impacted negatively. A range of such impacting factors, or knowledge management barriers, are illustrated in Figure 2.17.

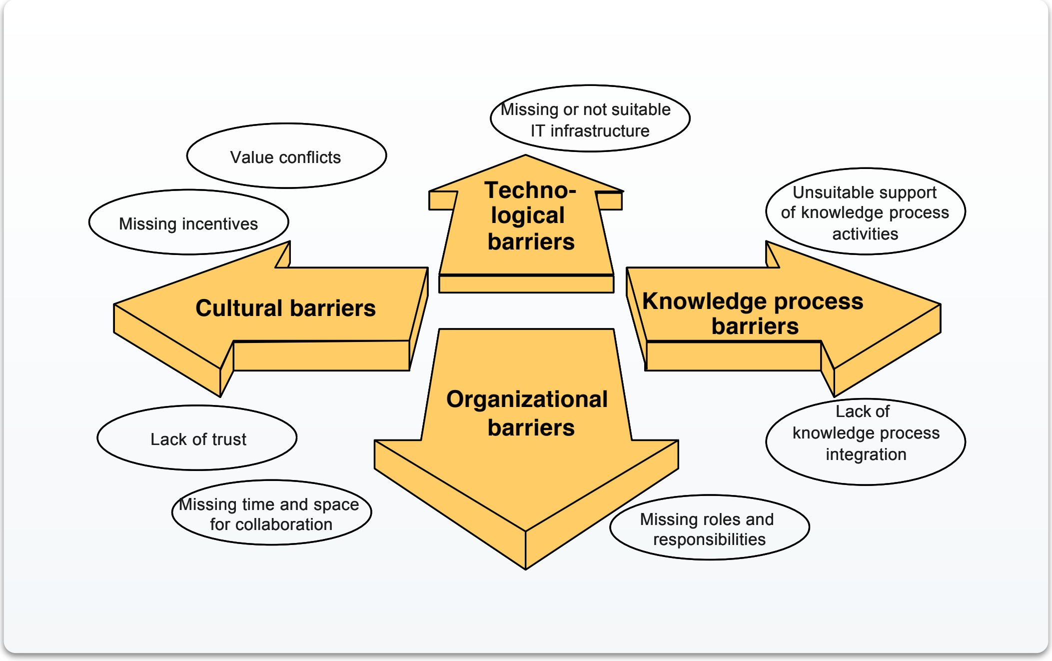


Figure 2.17: Knowledge management barriers

(Raimann 2000)

Various key success factors for the implementation and operation of knowledge networks have furthermore been identified (Forfás 2004), and include:

- **Clear need:** It should be clear to network stakeholdersthat the network can achieve goals that they cannot achieve on their own.
- **Objectives:** These should reflect the needs of the network stakeholders.
- **Leadership and vision:** Leaders convey the network objectives and transform them into plans of action that are tangible to network stakeholders.
- **Early successes:** These convince network stakeholders that there will be returns on their investment in both the short and long term.
- **Trust:** A change in business logic to accept reliance on networks as a source of competitive advantage is enabled by trust among network stakeholders.
- **Ownership:** Networks with members that take ownership of the network are far more likely to succeed.
- **Time:** The formation of a durable network takes time as network stakeholders have to build trust.
- **Critical mass:** The lack of critical mass can delay the outputs from a network.

- **Key player:** The presence of a major network member with the vision and resources can be influential in driving the network forward.

2.5.5.4 Management challenges and barriers of knowledge networks

The creation of successful knowledge networks poses a significant managerial challenge and requires skilled managers who understand the complexities and dynamics of knowledge generation and transfer. A number of these management challenges have been identified as (Schönström 2005; Schutte 2010):

- Identifying knowledge activists from within in the network to act as network coordinators.
- Making knowledge networks part of the corporate knowledge management strategy.
- Organisational restructuring may disrupt activities of knowledge networks.
- Creating an understanding of how formal networks relate to the line organisation.
- The difficulties involved in observing and measuring knowledge and related knowledge management activities.

The following barriers typically hamper the success of knowledge networks (Forfás 2004):

- A general lack of awareness as to the benefits of networks (as a concept distinct from “networking”) among the business community.
- A reluctance to commit time and resources to a process which is not well understood, or of which the results are unclear.
- A reluctance to share information and knowledge with other organisations, especially competitors.
- Enterprises are not always well placed to identify the opportunities for network relationships with other companies, since their knowledge and information base may be limited to their own contacts.
- Membership of a network may expose organisations to the danger of “lock-in”, in which excessive emphasis is placed on the affairs of the network, to the detriment of awareness and events in the outside environment.
- The “collective action problem”, in which a group of individuals or organisations may frequently fail to achieve successful cooperation, even though it would be beneficial to each individual in the group. This is largely attributed to a lack of skills or resources among managers to facilitate or coordinate the actual implementation of the network.

2.5.5.5 Generic knowledge network framework

Schutte (2010) proposed expansions to the framework for knowledge networks contributed by Seufert et al. (1999) (refer to section 2.5.5.1). This expanded generic knowledge network framework is illustrated in Figure 2.18.

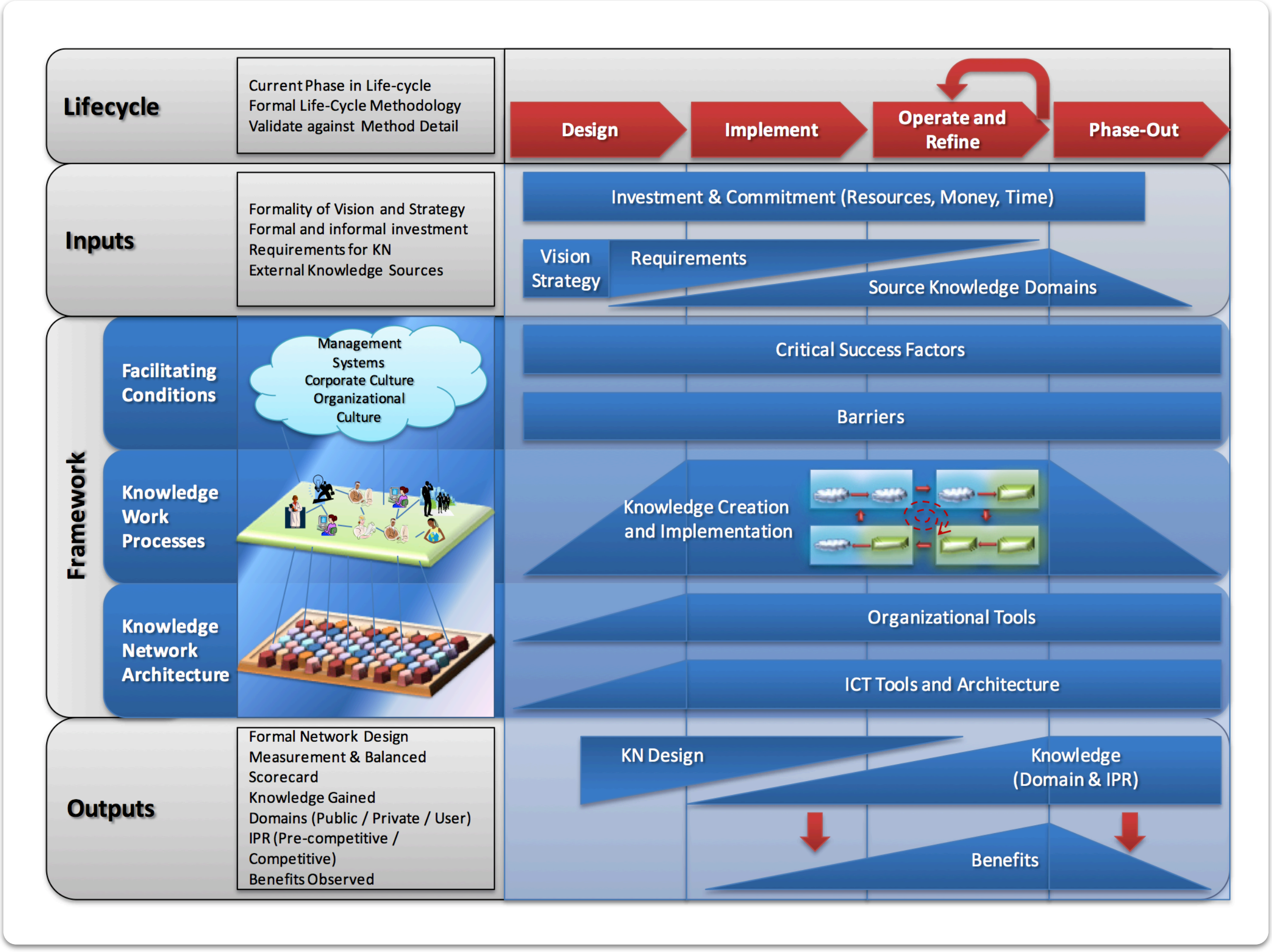


Figure 2.18: Generic knowledge network framework

(Schutte 2010)

The framework features two axes:

- A horizontal axis based on the life cycle view of the methodology for IKNs (refer to section 2.5.5.2), and
- A vertical axis based on a functional model of IKNs (refer to section 2.6.2.1), featuring inputs, the knowledge network process and outputs

The framework illustrates the following aspects of a generic knowledge network, along with an indication of their relevance or presence in each phase of the methodology for IKNs (Schutte 2010):

- **Knowledge network life cycle**, consisting of design, implementation, operation and refinement, and phase-out phases.
- **Inputs**
  - Vision and strategy

  - Requirements
  - Investment and commitment
  - Source knowledge domains
- **Framework**
  - Facilitating conditions
  - Knowledge work processes
  - Knowledge network architecture
- **Outputs**
  - Knowledge network design
  - Created knowledge
  - Benefits

The framework of Schutte (2010) represents a comprehensive functional view on IKNs, as it is concerned with the function and behaviour of this class of networks. Such a functional view is adequate for using and controlling systems, and has the “black-box” model as the corresponding kind of model (refer to section 2.6.2.1). However, in order to build and subsequently change IKNs, a view on the construction and operation of these networks is required (refer to section 2.6.2.2). Such a view can be achieved through the discipline of EE, which is discussed in the next section.

2.6 Enterprise engineering

This section discusses the engineering of enterprises in general. A more detailed discussion of functional perspectives on the engineering of IKNs in particular may be found in section 5.3.1, as part of the discussion of the scenarios in which the reference architecture for IKNs is required to function.

In order to enable the discussion of EE in this section, the definition of an enterprise adopted for the purpose of this study in section 1.2.2 is presented:

> *An enterprise is a complex, socio-technical system that comprises interdependent resources of people, information and technology that must interact with each other and their environment in support of a common mission.* (Giachetti 2010:4)

The term “enterprise” therefore encompasses all types of organisations, including private companies, government, non-profits, supply chains, virtual enterprises, as well as parts of a company such as a division or programme (Giachetti 2010). Given this understanding of the term enterprise, an early definition of EE reads as follows:

> *Enterprise engineering can be defined as the art of understanding, defining, specifying, analyzing, and implementing business processes for the entire enterprise life cycle, so that*

> *the enterprise can achieve its objectives, be cost-effective, and be more competitive in its market environment.* (Vernadat 1996:30)

More recently, Giachetti (2010) defines EE as:

> *…the body of knowledge, principles, and practices to design an enterprise.* (Giachetti 2010:3)

The key element of this definition is the phrase “to design”, which is considered the defining activity of engineering (Giachetti 2010; Hoogervorst 2009; Dietz et al. 2013). The first definition however implies that enterprises are not designed only once, but are, to varying degrees, redesigned on multiple occasions until their eventual disposal (Vernadat 1996; Giachetti 2010). EE is considered to be at the crossroads of various disciplines concerned with the design, re-engineering, and continuous improvement of business processes of enterprises in general. These disciplines include systems engineering, industrial engineering, logistics, manufacturing engineering, information systems engineering and software engineering (Vernadat 1996).

The primary motivation for EE is as an enabler for modern enterprises that are challenged by their own internal complexity, as well as a rapidly changing environment and increased competition (Liles et al. 1995; J. Martin 1995; Towill 1997; Kosanke et al. 2000; Rouse 2004; Saenz et al. 2009). EE is concerned with the design of the enterprise in order to enable a range of characteristics that are desirable in the modern economy, including flexibility and agility (Giachetti 2010). In order to achieve these characteristics, it is necessary to transition from a situation in which enterprises are evolved in an ad hoc fashion to a systematic, engineering approach to the design of enterprises (Giachetti 2010; Dietz et al. 2013).

EE initiatives are commonly conducted through projects that are executed within the enterprise by an EE team (Giachetti 2010). This team may include business systems analysts, enterprise architects, system architects, project managers, system designers, change managers, system engineers and even application developers (Giachetti 2010).

From the definition of EE provided by Vernadat (1996), it follows that the understanding of the enterprise life cycle is central to the activities of EE. This concept discussed in section 2.6.1. The definition provided by Giachetti (2010) highlights the design of enterprises as complex systems, and this is discussed in section 2.6.2.

2.6.1 Enterprise life cycle

This section discusses the life cycle of enterprises in general. A discussion of the life cycle of IKNs in particular may be found in section 5.3.2 as part of the discussion of the scenarios in which the reference architecture for IKNs is required to function.

Any entity that experiences a change in state, exhibits a life cycle that is comparable with others, and independent of both content and detail (Williams et al. 1998). This life cycle constitutes several phases, including a beginning and an end, that describe the evolutionary states of that entity (Giachetti 2010). The life cycle is not directly descriptive of the time-based order of events, as the phases depict the design process as opposed to time itself, and is known as a virtual life cycle (Department of Industrial Engineering, Stellenbosch University 2013). The progression of such a life cycle may be captured in a graphical or narrative description (Essmann 2009). All systems within an enterprise, including products, technologies, processes, strategies and the enterprise as a whole, can be modelled with an appropriate life cycle (Williams et al. 1998).

Since the description of a life cycle captures the progressive stages in the life history of an entity, it is able to describe the required steps in the development of a desired future version of the entity (Williams et al. 1998). Life cycles therefore provide the basis for the preparation of a methodology for carrying out the development of a new version of the enterprise, and is a foundational concept in EE (Williams et al. 1998).

The IFIP-IFAC Task Force (2003) identified elemental phases of the enterprise life cycle during their construction of the Generalised Enterprise Architecture and Methodology (GERAM, refer to section 2.7.4). The following phases (refer to Figure 2.19) represent the types of activities that are pertinent during the life of the enterprise (IFIP-IFAC Task Force 2003):

- *Identification* represents the activities that identify the content of the particular enterprise under consideration.
- *Concept* represents the activities required to develop the underlying conceptual requirements of the enterprise.
- *Requirements* represent the activities necessary for the development of descriptions of enterprise operational requirements, the relevant processes and a collection of all functional-, behavioural-, informational- and capability necessities.
- *Design* embodies all tasks that support the specification of the enterprise and all subcomponents necessary to satisfy the stipulated requirements.
- *Implementation* embodies the definition of the tasks necessary for the construction or reconstruction of the enterprise.
- *Operation* embodies all tasks necessary for the operation of the enterprise while producing the products and/or services along with all those tasks needed for monitoring, controlling, and evaluating the operation.
- *Decommission* embodies all tasks required for the recommissioning, retraining, redesign, recycling, preservation, transfer, disbanding, disassembling, or disposing of all or part of the enterprise, once the limits of their usefulness have been reached.

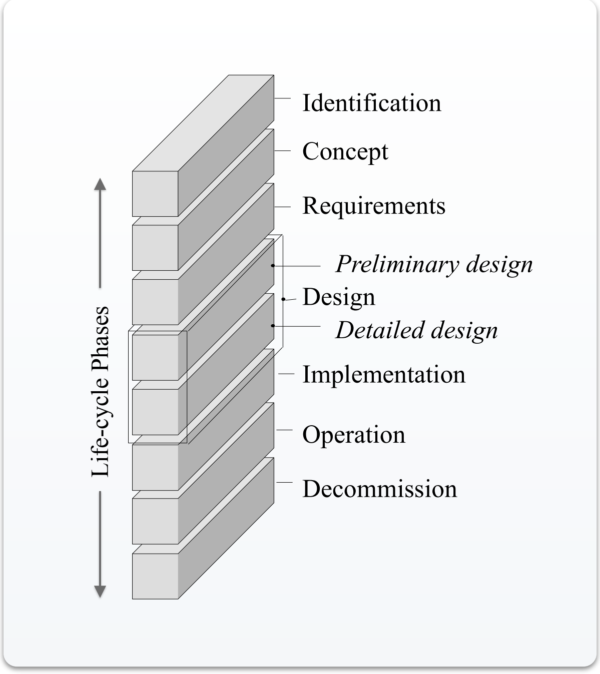


Figure 2.19: GERAM enterprise life cycle phases

(IFIP-IFAC Task Force 2003)

The IFIP-IFAC Task Force (2003), however, also identify a life history concept that complements the enterprise life cycle. They state that the life history of an enterprise is the “representation in time of tasks carried out on the particular entity during its entire life span” (IFIP-IFAC Task Force 2003). While the enterprise life cycle is a virtual life cycle, the life history of an enterprise is regarded as an actual life cycle (Department of Industrial Engineering, Stellenbosch University 2013). The life history of an enterprise allows for the identification of the tasks associated with life cycle phases as activity types (IFIP-IFAC Task Force 2003).

The concept of the life cycle of an enterprise versus its life history is illustrated in Figure 2.20. This figure shows that the enterprise may move to a previous phase in the life cycle as a result of an EE initiative, while remaining in the operation phase to ensure continued support for the current enterprise design. Enterprise life cycle phases are therefore not necessarily executed sequentially, which illustrates the iterative nature of the life cycle concept compared to the time-based approach of a life history. The iterations that comprise a life history serve to identify various change processes required on the operational processes, as well as the product of customer services (IFIP-IFAC Task Force 2003).

It is possible for multiple change processes to be executed at any given time, all of which occur simultaneously during the operation of the enterprise (Essmann 2009). Furthermore, these change processes may interact with one another on various levels of detail. For instance, the engineering design and implementation processes could be executed concurrently within an EE project and would occur simultaneously with the operation of the enterprise.

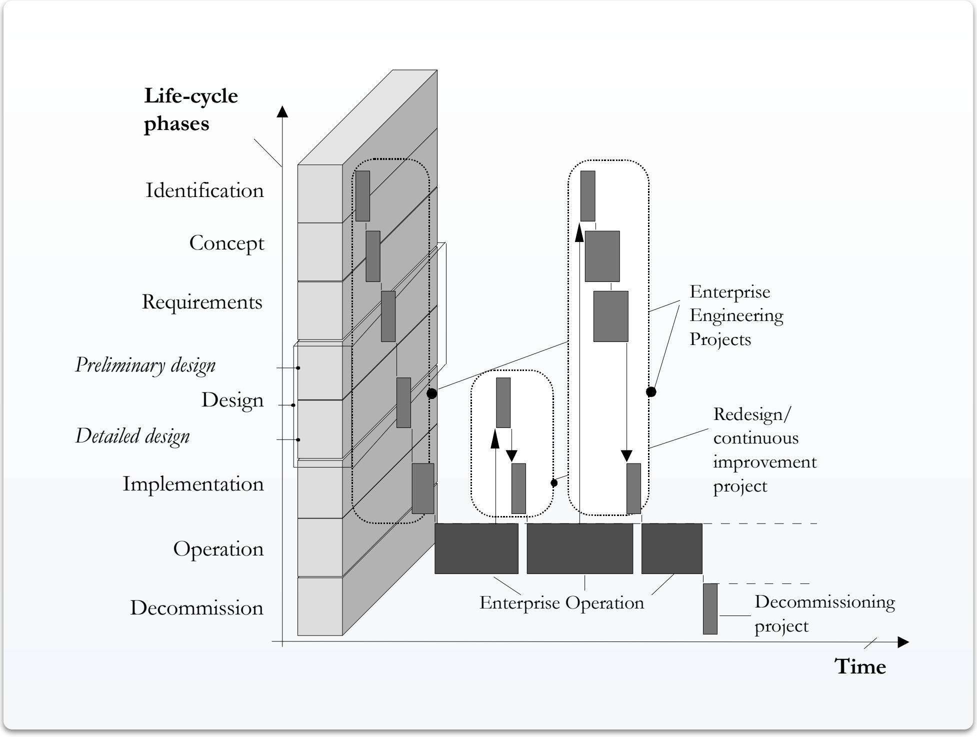


Figure 2.20: Parallel processes in the life history of an enterprise
(IFIP-IFAC Task Force 2003)

This section has presented the life cycle of an enterprise, representing the evolution of an enterprise as a system through time. The next section discusses system design perspectives that pertain to EE in more detail.

2.6.2 Enterprise systems

This section introduces the design of enterprises as systems in EE, and provides the context for the discussion of technical perspectives on the engineering of IKNs in section 6.3.1.

Two definitions of the term system read as follows:

> *A set of different elements so connected or related as to perform a unique function not performable by the elements alone.* (Rechtin & Maier 2000)

> *A set of elements standing in interrelation among themselves and with the environment.* (Bertalanffy 1969)

Given this understanding of systems, EE regards enterprises as complex socio-technical systems, and stresses that it is the interactions between the system components that are important to the enterprise behaviour (Giachetti 2010; Dietz & Hoogervorst 2008; Bernus & Nemes 1996). The enterprise is furthermore regarded as an open system in that it interacts with its environment, and is purposeful as it works toward accomplishing its goals (Giachetti 2010). The system approach offers a formal methodology to address the enterprise as a whole, while considering its constituent parts and their mutual relationships, in order to safeguard a unified and integrated system design (Dietz & Hoogervorst 2008).

Enterprises are intentionally created entities of human endeavour (Daft 2012; Dietz et al. 2013) and as such are examples of *organised complexities* (Weinberg 2001). This means that they are highly complex, while simultaneously exhibiting a high degree of organisation. These entities pose different challenges than “organised simplicities” that can be dealt with through analytical approaches, or “unorganised complexities” that can be addressed statistically (Weinberg 2001). Furthermore, modern enterprises are required to be in a continuous state of flux, as they are able to adapt to their surroundings. This requirement to accommodate change further compounds the complexity of the enterprise environment (Dietz et al. 2013). This makes the enterprise environment suitable to, and in need of, a systems approach (Bertalanffy 1969; Bunge 1979; Rechtin 1999; Gharajedaghi 2011). By applying a systems approach to enterprises, their study benefit from various typical system properties. These include system boundaries, subsystems, holism or complementation, open versus closed, purposefulness, feedback and control versus dynamic interactions, and complexity (Giachetti 2010).

Dietz (2006) identifies two system notions, each with its own value, its own purpose, and its own type of model: the teleological and the ontological system notions. These two system notions are discussed in the following two sections (sections 2.6.2.1 and 2.6.2.2 respectively), followed by a discussion of their interactions during the system design process in section 2.6.2.3.

2.6.2.1 Teleological system notion

The teleological system notion describes the function and external behaviour of a system, and evidence of teleology, i.e. purpose or goal-seeking behaviour (Rosenbleuth et al. 1943), is clearly observed in enterprises (Hitchins 2003). An understanding of the behaviour of a system allows managers to control the system and it is accordingly the dominant notion employed by managers (Dietz 2006). Management is usually concerned with the functions of an enterprise and how control of the input variables has an effect on output variables. A typical system property emphasised with the teleological system notion is that of system feedback and control (Rosenbleuth et al. 1943). Managers of enterprises typically use performance measurement to gain feedback and control over enterprise behaviour.

The teleological system notion corresponds to the “black-box” model type (Dietz 2006). A “black-box” model essentially consists of the relation between a set of input variables and a set of output variables, called a transfer function. These models conceptualise the functions and behaviours of the system without knowing the detail construction and operation of the system. “Black-box” models are therefore adequate for the purpose of using or controlling a system, and is also the dominant system concept in the social and organisational sciences (Dietz & Hoogervorst 2008). An example of a “black-box” model is the functional decomposition model of a computer (the computer being the system), e.g. a computer consists of a power system, processing system, display system and connectivity system. “Black-box” models are not useful to an engineer when maintaining a system (Dietz 2006).

2.6.2.2 Ontological system notion

An ontological or constructional system notion needs to be adopted for the purpose of building and changing systems (Dietz 2006). In order to understand the ontological system notion, various typical system properties are invoked. Bunge (1979) uses the system boundary property to distinguish between different constructs of an open system (refer to Figure 2.21).

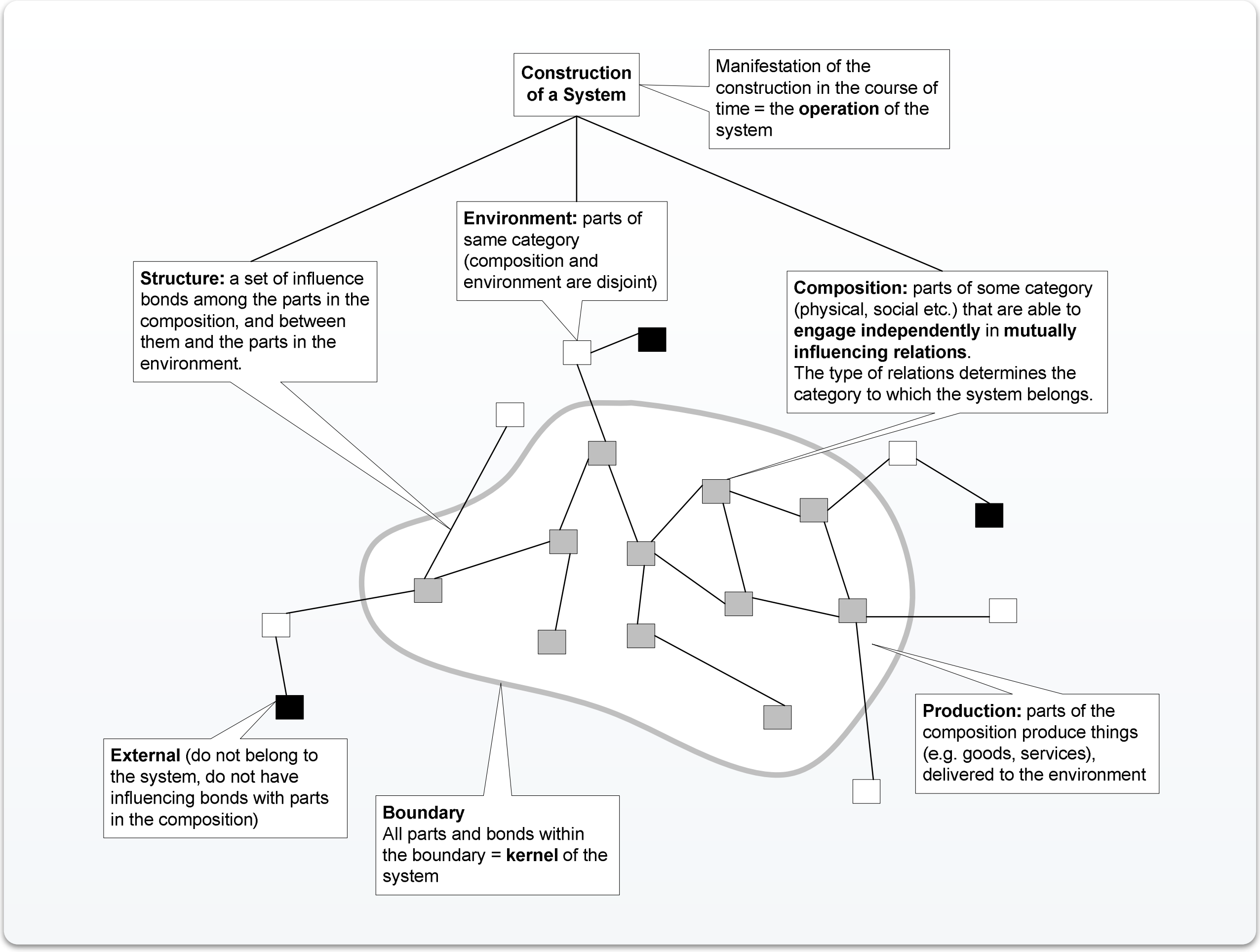


Figure 2.21: The structure of a system

(De Vries 2012; adapted from Dietz 2006)

Due to a logical and/or physical system boundary, a system consists of its (Dietz & Hoogervorst 2008):

- *composition* (parts of the same category, i.e., for example, physical, social and biological),
- *environment* (parts of the same category, but not within the boundary of the system),

- *structure* (a set of influencing bonds between the parts within the boundary, and between them and the parts in the environment), and
- *production* (the parts within the boundary produce things that are delivered to the parts in the environment).

Applying the constructs depicted in Figure 2.21 to an enterprise, the composition of the enterprise as a social system would consist of social individuals. The environment would consist of parts of the same category (social individuals) directly linked to the compositional parts, but outside the boundary, e.g. competitors and customers. The structure would be the mutual influencing relations among the system parts, i.e. individuals both within the boundary and certain individuals outside the boundary. The production of the enterprise would be goods and/or services that are delivered to the environment.

The ontological system notion, with its associated “white-box” model, describes the construction and operation of a system, and is therefore the dominant system concept in the engineering sciences (Dietz & Hoogervorst 2008). An example of a “white-box” model is the constructional decomposition model (i.e. bill-of-material) of a computer (the computer being the system), e.g. a computer consists of a case, power unit, processor, memory, display and input devices.

There exists a fundamental relationship between the teleological (refer to section 2.6.2.1) and ontological system notions in that the function and behaviour of a system is brought about and explained by its construction and operation. When designing a system, including an enterprise, both the teleological (functional) and ontological (constructional) system definitions are also relevant (Dietz & Hoogervorst 2008). This process is discussed in the following section.

2.6.2.3 System design

According to Dietz (2006), the system design process starts with a using system (US) that has a need for an object system (OS) (refer to Figure 2.22). This need stems from the construction of the US, and the process therefore starts with the development of a “white-box” model of the US (1). Thereafter the requirements for the OS are determined in terms of the construction and operation of the US, a process that is commonly referred to as function design. These requirements for the OS relate by nature to the function and behaviour of the OS, and are thus articulated in terms of a “black-box” model of the OS (2). This “black-box” model also includes various non-functional requirements, such as performance and quality aspects.

The next step in the system design process is to specify the construction and operation of the OS in terms of a “white-box” model of the OS, i.e. construction design (refer to Figure 2.22) (3). The steps of function design and construction design correspond to analysis and synthesis (Alexander 1964) (4). This system design process, however, is an iterative one, with the end result being a balanced compromise between reasonable requirements and feasible specification.

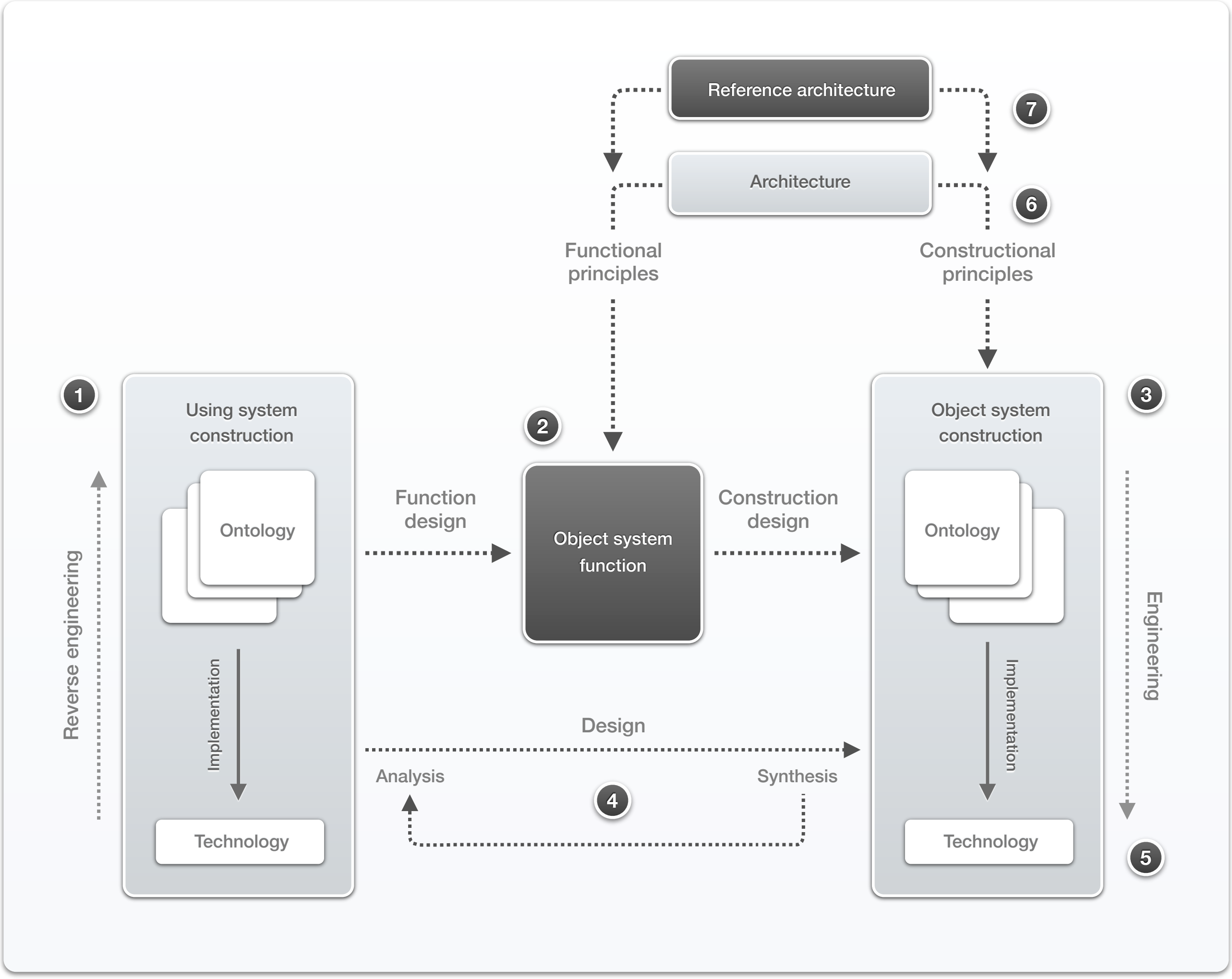


Figure 2.22: The generic system development process

(adapted from Dietz & Hoogervorst 2008)

Having designed a system at the ontological level in order to arrive at its constructional specification, it has to be designed in further detail in order for the system to be implementable. This detailed design process represents a narrow view on “engineering”, and consists of producing a coherent set of “white-box” models of the system (Dietz & Hoogervorst 2008). The “lowest” one of these construction models is commonly referred to as the implementation model and can be implemented on the available technological platform (refer to Figure 2.22) (5). The “highest” construction model, called the ontological model of the system, is fully independent of its implementation, and only describes the essential features of the system.

In the iterative system design process discussed above, architecture influences both function and construction design in order to produce an integrated system. The role of architecture in the design of systems may be understood as “the normative restriction of design freedom” (Dietz & Hoogervorst 2008) which is required to arrive at an integrated system (refer to Figure 2.22) (6). It may be argued that these restrictions are expressed in terms of a consistent and coherent set of design principles embodied in the artefacts that comprise the architecture. Among these design principles it is possible to distinguish between functional principles and constructional principles (Dietz & Hoogervorst 2008).

Furthermore, it is possible to define a reference architecture that holds for many systems, typically for a class of similar systems (Giachetti 2010) (refer to Figure 2.22) (7). Through this mechanism of reference architectures, it is possible to impose sound functional and constructional principles to ensure the integrated design of an entire class of systems. Enterprise architecture and the reference architecture mechanism are discussed in more detail in the following section.

2.7 Enterprise architecture

Diverse definitions exist for EA, indicating that it is a highly complex, dynamic construct that encapsulates both technical and social dimensions, the present and the future, as well as the logical and the physical aspects of the enterprise (Sidorova & Kappelman 2010). Giachetti (2010) defines EA as follows:

> *An Enterprise Architecture describes the structure of an enterprise, its decomposition into subsystems, the relationships between the subsystems, the relationships with the external environment, the terminology to use, and the guiding principles for the design and evolution of an enterprise.*

This study adopts the understanding that the *architecture* of an enterprise is a collection of artefacts (e.g. models and descriptions) that describe the enterprise (Kappelman 2011; Kaisler et al. 2005). In order to avoid confusion between this collection of artefacts and the manifestation thereof by the enterprise, there is however a distinction between an *architecture description* (the collection of artefacts) and an *architecture instantiation* (the manifestation). An architecture description serves to create a shared language to discuss and document important aspects of the enterprise, and also forms the basis for integrating and changing the enterprise (Kappelman & Zachman 2013; The Open Group 2011). Other scholars, however, also associate EA with the process of defining these enterprise standards and creating the required models (Bernard 2012; Kappelman & Zachman 2013).

Given the above understanding of EA, it relates to EE by providing a high-level design of the enterprise that ensures that EE projects deliver an integrated design (Giachetti 2010). In the previous section, the role of architecture in the system design process, however, was also described as “the normative restriction of design freedom” (Dietz & Hoogervorst 2008) that is required to arrive at an integrated system. It was argued that these restrictions are expressed in terms of a consistent and coherent set of design principles embodied in the artefacts that comprise the architecture.

Lapalme (2012) states that the debates on what EA entails may be traced back to different schools of thought that exist in the EA community. He (Lapalme 2012) suggests the use of three schools of thought to create common grounds in our understanding of the different value-propositions offered by EA authors. Lapalme hypothesises that three schools of thought on EA exist:

- enterprise IT architecting (EIT),
- enterprise integrating (E), and
- enterprise ecological adaptation (EiE).

This taxonomy of three schools of thought is not meant to be exhaustive and should be viewed as “ideal” types, i.e. author(s) typically do not fit perfectly into one school, but rather gravitate toward one (Lapalme 2012). Hoogervorst (2009) states that the understanding and designing of enterprises lies in avoiding the either-or scheme by combining the structural-functionalistic perspective (evident in EIT and E) with the interpretative perspective (evident in EiE).

Table 2.3: A subset of qualifiers for the three schools of thought on EA

(adapted from Lapalme 2012, De Vries 2012)

Enterprise IT architecting	Enterprise integrating	Enterprise ecological adaptation
Scope		
Enterprise-wide IT platform (EIT). All components (e.g. software and hardware) of the enterprise IT assets.	**Enterprise (E).** The enterprise as a socio-cultural-techno-economic system; hence ALL the facets of the enterprise are considered – the enterprise IT assets being one facet.	**Enterprise-in-environment (EiE).** Includes the previous scope but adds the environment of the enterprise as a key component as well as the bidirectional relationship and transactions between the latter and its environment.
Purposes		
Effective enterprise strategy execution and operation through IT-Business alignment. The purpose is to enhance business strategy execution and operations. The primary means to this end is the aligning of the business and IT strategies so that the proper IT capabilities are developed to support current and future business needs.	**Effective enterprise strategy implementation through execution coherency.** The purpose is effective enterprise strategy implementation. The primary means to this end is designing the various facets of the enterprise (e.g. governance structures, IT capabilities, remuneration policies and work design) to maximise coherency between them and minimise contradictions.	**Innovation and adaptation through organisational learning.** The purpose is organisational innovation and adaptation. The primary means is the fostering of organisational learning by designing the various facets of the enterprise (e.g. governance structures, IT capabilities, remuneration policies, and work design) as to maximise organisational learning throughout the enterprise.

Enterprise IT architecting	**Enterprise integrating**	**Enterprise ecological adaptation**
Motto		
"EA as the glue between business and IT."	"EA as the link between strategy and execution."	"EA as the means for organisational innovation and sustainability."
Principles and Assumptions		
• Reductionism. • Business strategies and objectives are provided by the business and are correct. • Independent design of organisational dimensions. • Disinterest in non-IT dimensions.	• Holism. • Business strategies and objectives are provided by the business and are correct. • Environment as something to manage. • Joint design of all organisational dimensions.	• Holism. • System-in-environment coevolution. • Environment can be changed. • Joint design of all organisational dimensions.
Principal authors		
• Spewak and Hill (1993) • Finkelstein (2006) • Van den Berg and Van Steenbergen (2007) • Hanschke (2009) • Perks and Beveridge (2011) • Ross et al. (2013)	• Bernus et al. (1996) • Op't Land et al. (2008) • Giachetti (2010) • Kappelman (2011, 2013) • John Zachman (1987, 2013)	• Martin (1995) • Graves (2008) • Hoogervorst (2009) • Gharajedaghi (2011) • Smith and Graves (2011) • Lapalme and De Guerre (2012)

Given the understanding of EA that was adopted in this section, this study may be primarily associated with the enterprise-integrating (E) school of thought on EA.

The development of an EA that describes an enterprise is a non-trivial exercise. Section 2.6.2.3 referred to the use of reference architectures to structure the design of a class of systems. These reference architectures are discussed in greater detail in the following section.

2.7.1 Reference architectures and architecture frameworks

This section introduces the concepts of reference architecture and architecture frameworks. These two concepts are discussed in greater detail in the following sections:

- *A more detailed discussion of the behaviour of reference architectures in EE may be found in section 5.3.3, as part of the discussion of the functional requirements for the reference architecture for IKNs.*

- *The use of various dimensions to describe and evaluate architecture frameworks is discussed in section 6.3.2 as part of the selection of an architecture framework that is suitable to the engineering of IKNs.*

In order to integrate and align its EE efforts, each enterprise could develop its own EA as it sees fit. Giachetti (2010:105) states that “the problem with this approach is that developing an EA is an enormous undertaking fraught with the risk of omitting crucial elements, creating inconsistent interfaces between the views, and not finishing the project in a reasonable time period”. The use of a reference architecture is proposed as an improved approach (Bernus et al. 1996; Williams & Li 1999; Schekkerman 2004; Greefhorst et al. 2006), with one definition of a reference architecture reading as follows:

> *… a generic architecture that can be used as the starting point to derive an enterprise’s architecture.* (Giachetti 2010)

A reference architecture is therefore a collection of the generic parts, functions, descriptions, or behaviours of a system and the associated structures or frameworks (Bernus et al. 1996). It serves as an intellectual paradigm, facilitating the accurate analysis, discussion, and specification of a given area of discourse, i.e. a manner of viewing, conceiving and discussing a matter of concern (Vernadat 1996).

Reference architectures generally include a model for architecture descriptions, as well as a method to produce them, with some reference architectures emphasising either the descriptions or the method (Greefhorst et al. 2006). Reference architectures may therefore be classified into roughly two types (Williams 1994):

1. **Type 1:** These reference architectures only deal with the structural arrangement (design) of a physical system such as the manufacturing component of the enterprise, or the structure of the complete enterprise.
2. **Type 2:** These reference architectures deal with the structural arrangement (organisation) of the development and implementation of a project or programme such as an enterprise engineering programme. They illustrate the life cycle of the project developing the enterprise and therefore have a specific reference model that addresses the flow of enterprise engineering efforts along the life cycle.

Furthermore, reference architectures can be roughly categorised as either enterprise-class or application-class (Greefhorst et al. 2006). Enterprise-class reference architectures are often referred to as *enterprise reference architectures* and focus on enterprise-level architecture descriptions that describe business units, entire organisations or even industry sectors (Schekkerman 2004; Greefhorst et al. 2006). Application-class reference architectures are also referred to as *software reference architectures* and relate to application-level architecture descriptions that describe the architecture of a specific software application or a group of similar applications (Schekkerman 2004; Greefhorst et al. 2006).

A common characteristic or dimension with which to differentiate between various reference architectures is genericity or meta-level (Kosanke et al. 1999; Williams & Li 1999; IFIP-IFAC Task Force 2003; Greefhorst et al. 2006). Varying levels of genericity refer to levels of aggregation in the reference architecture, and therefore also the varied scope of applicability thereof in EE efforts. Three levels of genericity are typically identified for reference architectures (IFIP-IFAC Task Force 2003):

- *Generic,* representing those enterprise integration factors generic to *all enterprises*.
- *Partial*, representing those enterprise integration factors generic to a *particular industry* or *class of enterprises.*
- *Particular,* representing only a *specific enterprise*, i.e. an architecture description.

The terminology regarding reference architectures, however, is not consistent in literature (Giachetti 2010), with some authors referring to the term “architecture frameworks”, rather than “reference architectures”. Indeed, a definition of an architecture framework from an IT point of view reads as follows:

> *An architecture framework is a tool. It should describe a method for designing an information system in terms of a set of building blocks, and for showing how the building blocks fit together. It should contain a set of tools and provide a common vocabulary. It should also include a list of recommended standards and compliant products that can be used to implement the building blocks.* (The Open Group 2011)

Since this definition shares various elements with the understanding of reference architectures as discussed previously, the genericity levels listed above are used to disambiguate these terms for the purpose of this study:

- The term “architecture framework” is used to refer to *generic reference architectures.*
- The term “reference architecture” is used to refer to *partial reference architectures.*

Given this terminology, architecture frameworks are regarded as sources of input and guidance for the development of reference architectures. Reference architectures, in turn, provide inputs for the development of architecture descriptions. These architecture descriptions are instantiated to construct the operational enterprise. This use of architectural terms is illustrated along a genericity dimension in Figure 2.23. The disambiguation in terminology allows for statements such as SO3 (refer to section 4.4.3): “to implement the selected architecture framework in a way that enables the artefact to function as a reference architecture”. Without the disambiguation discussed above, this objective would read as follows: “to

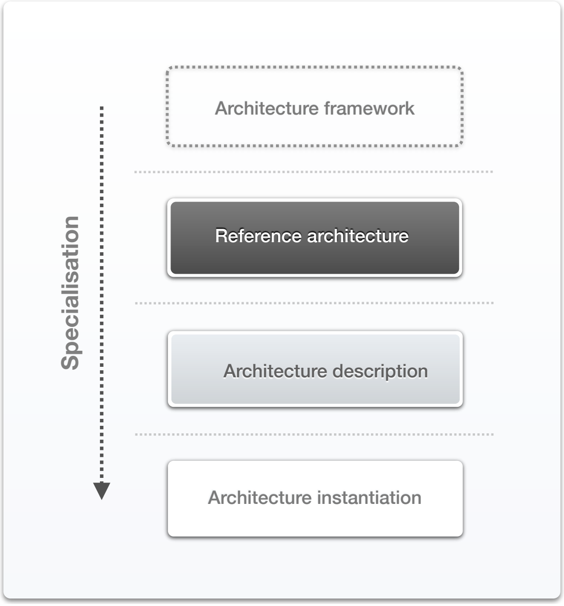


Figure 2.23: Architectural terms used in the study

implement the selected (generic) reference architecture in a way that enables the artefact to function as a (partial) reference architecture".

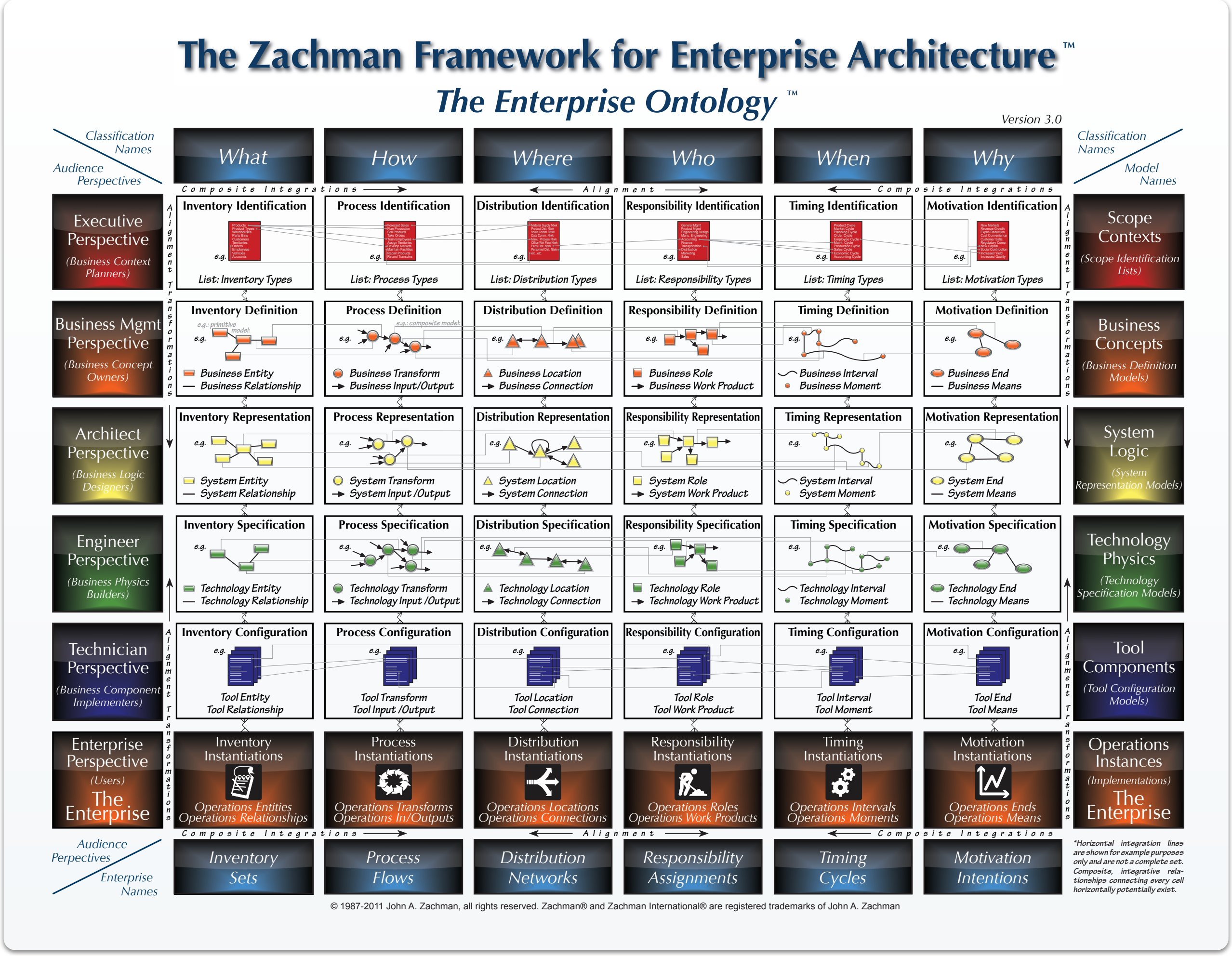
The following sections describe various architecture frameworks, namely the Zachman framework, PERA, TOGAF, GERAM and ARCON.

2.7.2 Zachman framework

This section provides a brief overview of the Zachman framework. Technical perspectives on the framework are discussed in greater detail in section 6.6, with functional perspectives on the framework discussed in section 7.3.

John Zachman, often referred to as the father of enterprise architecture, in 1987 first published a framework for information systems architecture while working at IBM (Zachman 1987). In 1992, the framework was extended to what is now known as the Zachman Framework for Enterprise Architecture (Sowa & Zachman 1992). This framework provides a logical structure for classifying and organising the descriptive representations that are significant to the management of the enterprise and the development of enterprise systems (Giachetti 2010). The Zachman Framework for Enterprise Architecture may furthermore be interpreted as an enterprise ontology (Kappelman & Zachman 2013). An ontology is "a theory of the existence of a structured set of essential components of an object for which explicit expression is necessary (or even mandatory) for designing, operating and changing the object" (Zachman 2008). The Zachman framework is regarded as a type 1 reference architecture (Greefhorst et al. 2006).

The framework consists of a six by six matrix (refer to Figure 2.24) that depicts six communication interrogatives (what, how, where, who, when and why) as columns and six reification transformations (scope contexts, business concepts, system logic, technology physics, tool components, and operations instances) as rows (Zachman 2011). It could be argued that this reification process is similar to the design process of systems engineering, which gradually transforms system requirements to implementations (De Vries 2012, refer to section 2.6.2.3).



Classification Names / Audience Perspectives	What	How	Where	Who	When	Why	Classification Names / Model Names
Executive Perspective (Business Context Planners)	Inventory Identification List: Inventory Types	Process Identification List: Process Types	Distribution Identification List: Distribution Types	Responsibility Identification List: Responsibility Types	Timing Identification List: Timing Types	Motivation Identification List: Motivation Types	Scope Contexts (Scope Identification Lists)
Business Mgmt Perspective (Business Concept Owners)	Inventory Definition Business Entity Business Relationship	Process Definition Business Transform Business Input/Output	Distribution Definition Business Location Business Connection	Responsibility Definition Business Role Business Work Product	Timing Definition Business Interval Business Moment	Motivation Definition Business End Business Means	Business Concepts (Business Definition Models)
Architect Perspective (Business Logic Designers)	Inventory Representation System Entity System Relationship	Process Representation System Transform System Input /Output	Distribution Representation System Location System Connection	Responsibility Representation System Role System Work Product	Timing Representation System Interval System Moment	Motivation Representation System End System Means	System Logic (System Representation Models)
Engineer Perspective (Business Physics Builders)	Inventory Specification Technology Entity Technology Relationship	Process Specification Technology Transform Technology Input /Output	Distribution Specification Technology Location Technology Connection	Responsibility Specification Technology Role Technology Work Product	Timing Specification Technology Interval Technology Moment	Motivation Specification Technology End Technology Means	Technology Physics (Technology Specification Models)
Technician Perspective (Business Component Implementers)	Inventory Configuration Tool Entity Tool Relationship	Process Configuration Tool Transform Tool Input /Output	Distribution Configuration Tool Location Tool Connection	Responsibility Configuration Tool Role Tool Work Product	Timing Configuration Tool Interval Tool Moment	Motivation Configuration Tool End Tool Means	Tool Components (Tool Configuration Models)
Enterprise Perspective (Users) The Enterprise	Inventory Instantiations Operations Entities Operations Relationships	Process Instantiations Operations Transforms Operations In/Outputs	Distribution Instantiations Operations Locations Operations Connections	Responsibility Instantiations Operations Roles Operations Work Products	Timing Instantiations Operations Intervals Operations Moments	Motivation Instantiations Operations Ends Operations Means	Operations Instances (Implementations) The Enterprise
Audience Perpectives / Enterprise Names	Inventory Sets	Process Flows	Distribution Networks	Responsibility Assignments	Timing Cycles	Motivation Intentions	*Horizontal integration lines are shown for example purposes only and are not a complete set. Composite, integrative relationships connecting every cell horizontally potentially exist.

© 1987-2011 John A. Zachman, all rights reserved. Zachman® and Zachman International® are registered trademarks of John A. Zachman

Figure 2.24: The Zachman Framework for Enterprise Architecture

(Zachman 2011)

2.7.3 PERA

PERA is referred to as a type 2 reference architecture, since it depicts the steps and structure necessary for the analysis, design and development of an enterprise integration initiative (Williams et al. 1996) (refer to Figure 2.25). It is suitable as a reference architecture for modelling any enterprise (Williams et al. 1994). The PERA structure is life-cycle based and incorporates the necessary life cycle concepts inherent in

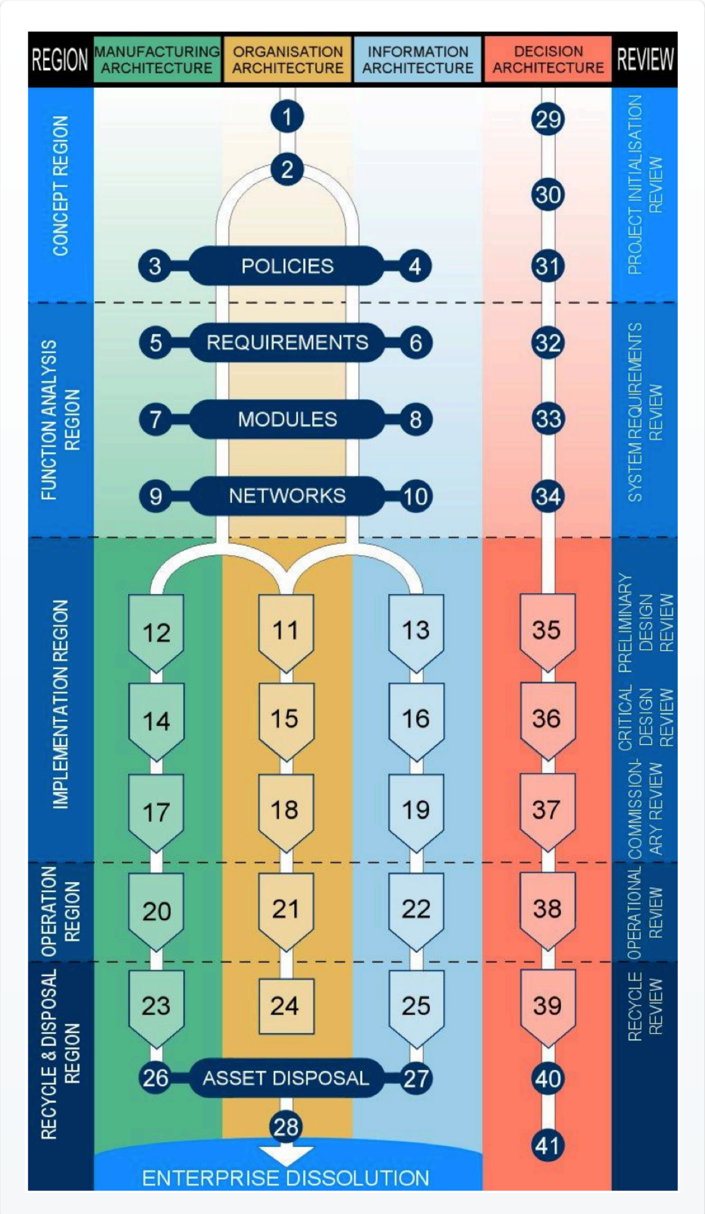


Figure 2.25: Extended PERA

(Katz 2005)

developing any enterprise-related entity. PERA describes the following three major components (sub-architectures) as the basis of any enterprise (Williams et al. 1998):

- Manufacturing Equipment Architecture – equipment performing physical manufacturing functions or tasks.
- Information Systems Architecture – equipment performing information functions or tasks.
- Organisation and Human Architecture – human execution of functions or tasks in either or both cases.

An extended version of PERA exists, which expands on these components by including a fourth component, namely the Decision Architecture, pertaining to decision-making protocol (Department of Industrial Engineering, Stellenbosch University 2013). The concepts of the Manufacturing Equipment Architecture are also applicable to the creation of intangible products, i.e. services, in the Extended PERA (Department of Industrial Engineering, Stellenbosch University 2013). The life cycle phases addressed by the reference architecture are those of concept, function analysis, implementation, operation, and recycle and disposal.

2.7.4 GERAM and VERA

GERAM (Generalised Enterprise Reference Architecture and Methodology) was developed in the late 1990s by the IFIP-IFAC Task Force by evaluating existing enterprise integration architectures (IFIP-IFAC Task Force 2003). Source reference architectures for the development of GERAM included CIMOSA, GRAI/GIM and PERA (Williams & Li 1999, refer to section 2.7.3). GERAM consists of various components, including a reference architecture (GERA) and an enterprise engineering methodology (EEM) (IFIP-IFAC Task Force 2003).

The methodology was designed with the purpose of being applied to all types of enterprises, and has a holistic approach to EA (Williams & Li 1999). GERAM acts as a toolkit for designing and maintaining enterprises through their entire life cycle (IFIP-IFAC Task Force 2003), making it an example of a type 2 architecture framework. GERAM intends to merge the methods of various disciplines in the change process that occurs in the enterprise, including industrial engineering, management science, control engineering, communication and information technology (Department of Industrial Engineering, Stellenbosch University 2013).

GERA provides an analysis and modelling framework (refer to Figure 2.26) that is based on the life cycle concept and identifies three dimensions for defining the scope and content of enterprise modelling (IFIP-IFAC Task Force 2003):

- *Life cycle dimension*: providing for the controlled modelling process of enterprise entities according to the various life cycle activities (refer to section 2.6.1).
- *Genericity dimension*: providing for the controlled particularisation (instantiation) process from generic and partial to particular.
- *View dimension*: providing for the controlled visualisation of specific views of the enterprise entity.

The reference part of the modelling framework illustrated in Figure 2.26 consists only of the generic and partial levels (IFIP-IFAC Task Force 2003). These two levels organise and structure the definitions of concepts, basic and macro level constructs, that are defined and utilised for the description of the given enterprise. The particular level represents the results of the modelling process, which is the architecture

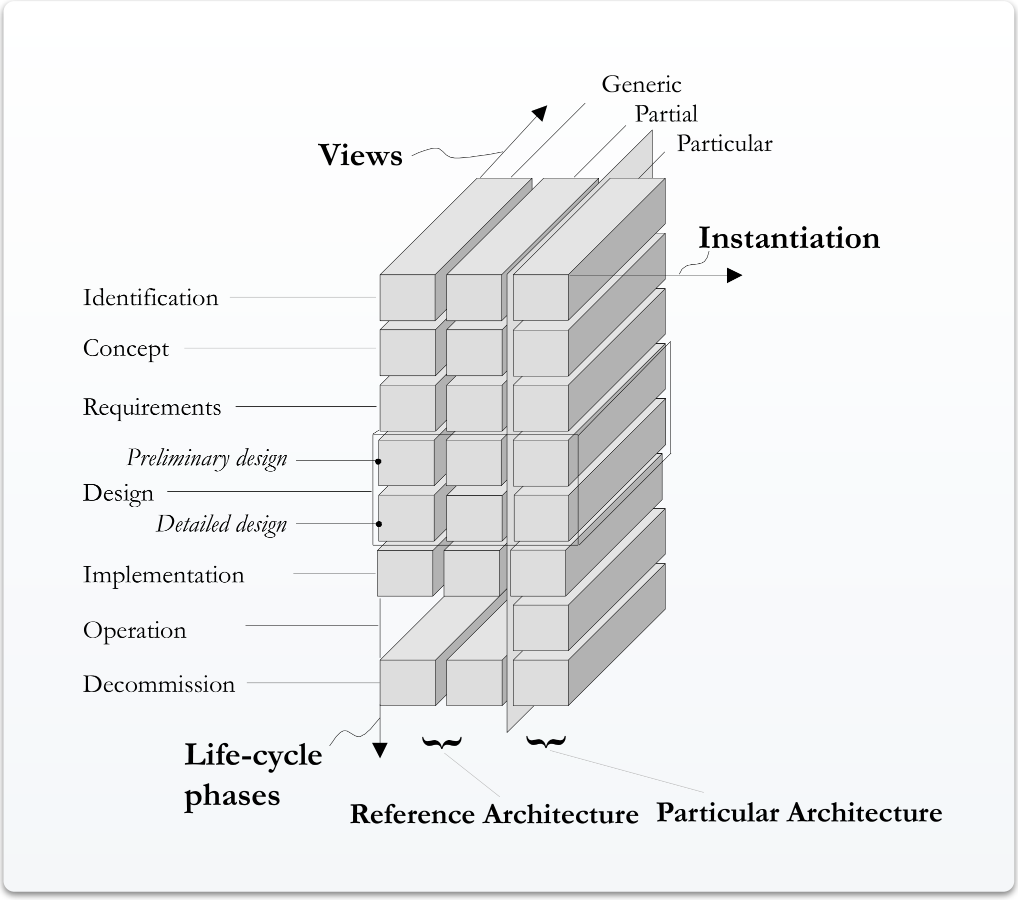


Figure 2.26: GERA modelling framework

(IFIP-IFAC Take Force 2003)

description of the enterprise at the state of the modelling process corresponding to the particular set of life cycle activities (IFIP-IFAC Task Force 2003).

GERA has since been adapted to suit the needs of virtual enterprises (refer to section 2.2.3), in the form of VERA (Virtual Enterprise Reference Architecture) (Tølle et al. 2002). VERA is aimed at groups of enterprises that form an inter-organisational network by assigning competencies to the network in order to be able to form VEs (Vesterager et al. 2003). The functionality of VERA is illustrated in Figure 2.27, which shows the nesting of life cycles on the left, and the practical result as a life history on the right. In the operational phase of the network, customer-focused VEs are formed, which produce the required products or services. These product or service deliverables correspond to some product life cycle phases, with the phases often varying from one VE to the other and correspondingly from customer to customer (Vesterager et al. 2003). Two double arrows are therefore included on the lines between the VE entity and product entity Figure 2.27.

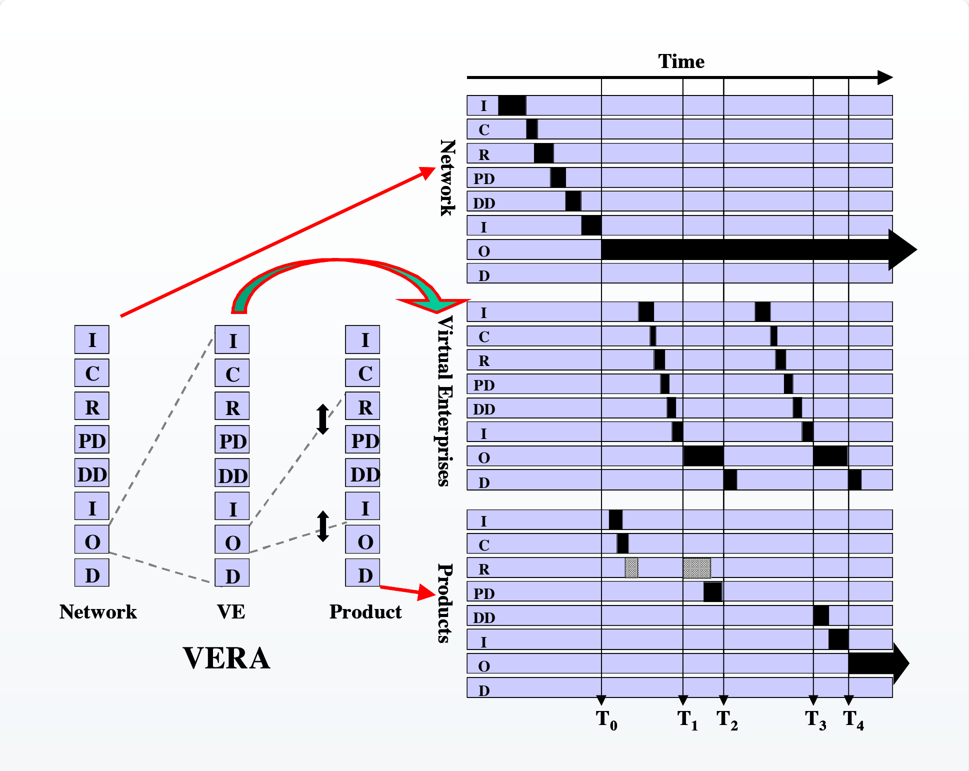


Figure 2.27: VERA and example of life history

(Vesterager et al. 2003)

2.7.5 TOGAF

TOGAF (The Open Group Architecture Framework), owned by the Open Group (The Open Group 2011), is primarily an IT architecture that looks to align the IT view with the business view. To this end, TOGAF describes four architectural views (Giachetti 2010):

- *Business architecture* describes the processes the business uses to meet its goals, and links strategy formulation to strategy implementation.
- *Application architecture* describes how specific applications are designed and how they interact with each other.
- *Data architecture* describes the enterprise's logical and physical data resources and how the data are managed.
- *Technical architecture* describes the hardware and software infrastructure that supports the business processes, applications and their interactions.

The TOGAF architecture framework consists of three main parts, namely the Architecture Development Methodology (ADM), the Enterprise Continuum, and the Resource Base (The Open Group 2011). TOGAF

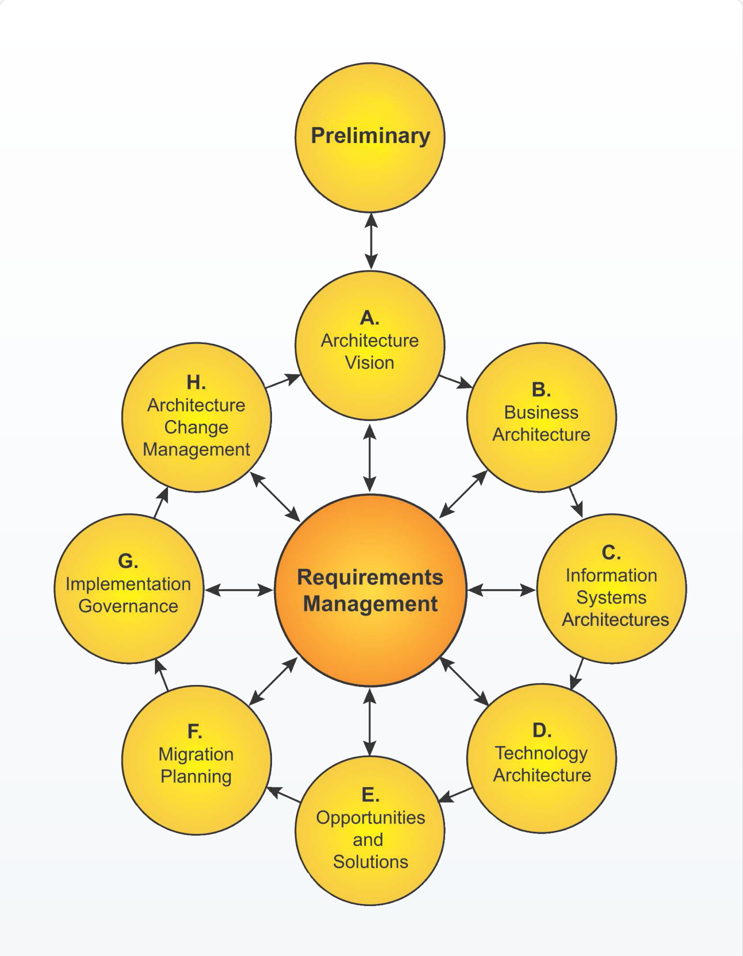


Figure 2.28: TOGAF ADM Cycle

(The Open Group 2011)

became best known for its ADM, which is an architectural process or methodology, rather than an architectural framework (Giachetti 2010). The ADM describes a detailed approach to generate architecture descriptions, and consists of ten phases (The Open Group 2011, refer to Figure 2.28):

- A *preliminary phase* defines the capabilities for doing architecture work, i.e. defining the "where, what, why, who and how we do architecture".
- *Phase A, Architecture vision*, defines the scope of the architecture effort and the constraints that must be dealt with.
- *Phase B, Business architecture*, defines the baseline and target business architectures, which is a prerequisite for architecture work in any other domain (data, application and technology).
- *Phase C, Information systems architecture*, defines the target data and/or application architectures that would support the target business architecture.
- *Phase D, Technology architecture*, maps the data and/or application components (defined in Phase C) to a set of technology components, representing required software and hardware components.

- *Phase E, Opportunities and solutions*, provides a logical grouping of IT activities into project work packages within the IT portfolio and other portfolios that are dependent upon IT.
- *Phase F, Migration planning*, creates a viable implementation/migration plan in co-operation with the portfolio and project managers.
- *Phase G, Implementation governance*, governs and manages the contract for implementing and deploying the solution(s).
- *Phase H, Architecture change management*, manages changes to the architecture in a consistent way.
- *Requirements management*, interacts with phases A to H and denotes the dynamic process of identifying, storing and managing the supply of enterprise architecture change requirements.

The Enterprise Continuum may be viewed as a “virtual repository” of all the architecture assets, i.e. models, patterns and descriptions, among others, available to the enterprise (Giachetti 2010). The Resource Base, in turn, is a set of tools (e.g. guidelines and templates) that are available to help the enterprise architect in the use of the ADM (Giachetti 2010).

2.7.6 ARCON

The ECOLEAD project sought to clarify basic concepts in CNs and their interrelationships, primarily through a taxonomy (with related definitions) of collaboration forms (refer to section 2.2) (Camarinha-Matos & Afsarmanesh 2007). To elaborate on the reference model contributed by the project, the ARCON reference modeling framework was developed (Camarinha-Matos & Afsarmanesh 2008a). ARCON offers a comprehensive modelling framework for CNs based on a three-dimensional approach, which includes (Camarinha-Matos & Afsarmanesh 2007) (refer to Figure 2.29):

1. The CN life-cycle dimension, including phases for network creation, operation, evolution/metamorphosis and dissolution.
2. The CN environmental perspectives dimension, including both endogenous and exogenous interactions.
3. The CN modelling intent dimension, with general concepts, specific modelling and implementation modelling.

Using this framework, the ECOLEAD initiative organised the most common general concepts under the endogenous elements and exogenous interactions perspectives (Mehandjiev & Grefen 2012). Endogenous elements are structured into four modeling dimensions as follows (Camarinha-Matos & Afsarmanesh 2007):

- *Structural*, relating to elements such as participants, relationships, roles and geographical location.
- *Componential*, e.g. hardware and software resources, human resources, information and knowledge resources and ontology resources.

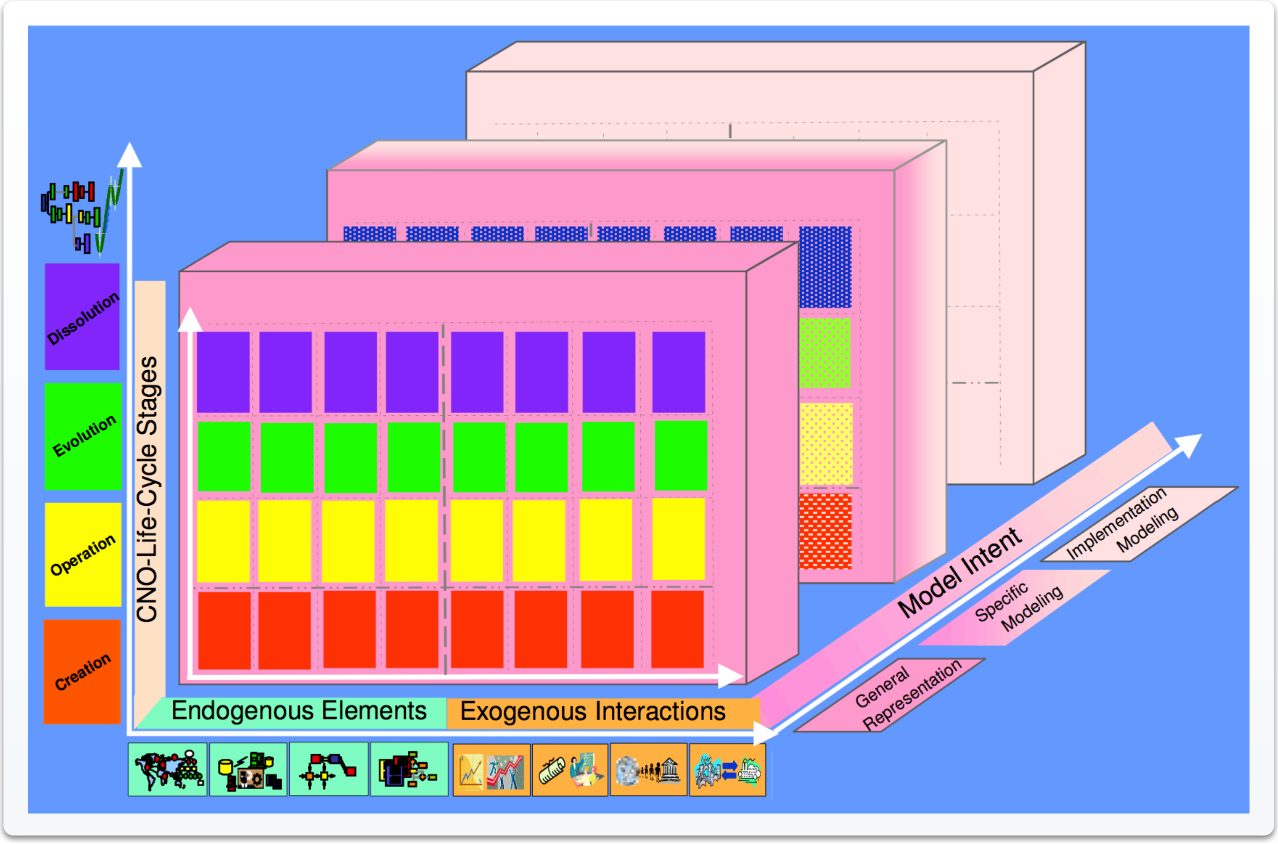


Figure 2.29: ARCON reference modeling framework for CNOs

(Camarinha-Matos & Afsarmanesh 2007)

- *Functional*, e.g. processes, auxiliary processes, procedures and methodologies.
- *Behavioural*, e.g. prescriptive behaviour, obligatory behaviour, constraints and conditions, contracts and agreements, and incentives.

Exogenous elements of the ARCON framework is organised as follows (Camarinha-Matos & Afsarmanesh 2007):

- *Market*, relating to elements such as customers, mission, transactions, marketing and branding, and market strategy.
- *Support*, e.g. certification, insurance, coaching and training.
- *Societal*, e.g. impacts, legal issues and public interactions.
- *Constituency*, e.g. attracting factors, rules of adhesion and sustainability factors.

The framework was first applied to the CN cases studied in ECOLEAD, namely VBEs, VOs and PVCs. An attempt to generalise from these cases was then made, trying to identify a common set of concepts and entities, which were discussed with a wide group of experts from different fields. This led to the first comprehensive reference model proposal (Camarinha-Matos & Afsarmanesh 2007). The ARCON reference modeling framework is seen as a starting point for further development of reference architectures in the CN domain (Camarinha-Matos & Afsarmanesh 2008a).

2.8 Conclusion

This chapter presented background information from literature that contextualises and reinforces the research problem identified in section 1.3.

The current landscape of collaborative networked organisations was discussed in section 2.2. Varying levels of interaction among network stakeholders, from simple networking to advanced collaboration, results in a host of base organisational forms within this domain. Among these organisational forms, an emergent collaborative network form was identified, namely inter-organisational innovation networks that are driven by knowledge sharing. These organisations may be described as IKNs, i.e. multi-domain knowledge-sharing networks aimed at fostering sustainable innovation that promote the competitiveness of their members. In order to characterise these networks, the concepts of knowledge and innovation were subsequently discussed.

Section 2.3 presented a brief introduction to epistemology, including a discussion of various types *knowledge*. The knowledge generation process and various knowledge work processes were introduced, highlighting the view that networks are required to effectively generate and exploit knowledge resources. Knowledge networks were therefore subsequently discussed.

Another key concept in understanding IKNs, namely *innovation*, was discussed in section 2.4. The phenomenon of innovation was defined and distinguished from other related terms, and types of innovation were identified. The innovation life cycle was discussed, followed by a discussion of various generations of innovation models. Particular attention was paid to recent models describing innovation networks and open innovation, leading to a discussion of IKNs.

IKNs were discussed in section 2.5, starting with an overview of the relationship between knowledge and innovation in these networks. The knowledge supply chain was discussed as a key concept in the definition of IKNs. The existing body of knowledge on IKNs, which amount to a functional view on these networks, was then presented. This effectively closed the loop of argumentation that started with the identification of CNOs that are driven by knowledge sharing in order to facilitate innovation at the end of section 2.2. The need for a constructional view on IKNs was identified, which led to a discussion of EE and EA in the following section.

EE was discussed in section 2.6, adopting a systems view on enterprises. This was followed by a discussion of the enterprise life cycle and the design of enterprises as systems. The role of EA to ensure the design of integrated enterprises through EE was discussed in section 2.7, along with a discussion of the reference architecture mechanism. Several architecture frameworks were reviewed, namely the Zachman framework, PERA, GERAM and VERA, TOGAF and ARCON.

This chapter made the extended case for the benefits of EE for IKNs, allowing them to deal with their own inter-organisational complexity, as well as their rapidly changing environment. The argument for EA to provide a high-level design that ensures an integrated system design, also holds within this context. It could therefore be argued that the development of architecture descriptions are vital to the adaptability of IKNs, since the availability of such descriptions enables organisational change (Kappelman & Zachman 2013). IKNs, however, are distinct from other CNOs discussed in this chapter, resulting in the limited applicability of existing reference architectures. A particular limiting factor is the predominantly functional focus of existing artefacts that address IKNs. The above argument serves as motivation for the research problem that was identified in section 1.3:

> **No reference architecture exists for use in the engineering of IKNs.**

A methodology and research design to develop a solution to this research problem is presented in chapter 3.

3. Methodology and research design

3.1 Introduction

Part 1 of this document identified and motivated the research problem addressed by the study, along with the research objective (RO) and primary research question (PRQ). This chapter presents the research methodology that is adopted in the study in order to answer the PRQ, and thereby achieve the RO. In order to support the discussion in this chapter, a number of terms are however firstly introduced, namely research method, research methodology, research paradigm and research design.

Research methods may be understood as the methods the researcher uses in performing research operations (Kothari 2004). The scope of a research methodology is wider than that of research methods, since it is the science of studying how research is done scientifically (Kothari 2004). The research methodology of a particular study considers the logic behind the methods that are employed in the study (Kothari 2004). This logic for the selection of certain methods over others is influenced by, amongst other things, various underlying assumptions regarding research in a field. The philosophical paradigm within which a study is conducted therefore also forms part of the research methodology of a study. Within this research methodology, a research design is constructed as a practical plan of action that links the paradigmatic assumptions to specific research methods, and indicates how the study will be conducted.

In order to present the research methodology for this study, various philosophical perspectives underlying research are presented in section 3.2. This is followed by an overview of research methods in section 3.3, with relevant approaches to data collection and analysis discussed in section 3.4. The methodology for the study is presented in section 3.5 and includes the research questions, research objectives, research philosophy, research design and methods. Limitations of the research design are discussed in section 3.6, before section 3.7 concludes this chapter.

3.2 Philosophical perspectives

This section presents an overview of various research paradigms that may form part of a research methodology.

Research is an investigation using a systematic process to discover reliable facts and knowledge (Olivier 2004). This systematic research process consists of collecting, analysing and interpreting information in order to increase our understanding of a phenomenon (Olivier 2004). All research, both qualitative and quantitative, is however based on underlying assumptions of the appropriate research methods and what constitutes valid research and knowledge (Gallupe 2007; Myers 1997). It is important to know what these assumptions are in order to conduct research in a responsible way (Leedy & Ormrod 2001). Based on the underlying epistemology, i.e. the assumptions about knowledge and how it can be obtained, at present the main trend indicates that three research paradigms can be identified. These are the positivist, the interpretive and the critical paradigms (Myers & Avison 2002; Orlikowski & Baroudi 1991).

1. *Positivism* asserts that reality is objective and testable and that it can be described by measurable properties (Myers 1997). The research focus is therefore on testable propositions and positivist studies are frequently hypothesis driven (Myers & Avison 2002). Positivist research is intended to produce an exact representation of reality and as such relies primarily on quantitative methods. The positivist paradigm originates from the natural sciences, but are also applied in the social sciences (De Villiers 2005).
2. *Interpretivism* attempts to understand phenomena through the meanings that people assign to them (Olivier 2004). Interpretive studies therefore assume that access to a given or socially constructed reality is only gained through social constructs such as language and shared meaning (Myers & Avison 2002). Accordingly, the way in which parties influence each other given the interaction between the research and the research participant, should be considered (Olivier 2004).
3. *Critical research*, similar to interpretive research, assumes that what is observed in society is socially constructed. The underlying epistemology of critical research is therefore that reality is historically constituted, produced and reproduced by people (Myers 1997). The research paradigm, however, focuses on critique of oppositions, conflicts and contradictions in contemporary society and to eliminate the causes of alienation and domination (Myers & Avison 2002).

The application of research approaches, their paradigmatic assumptions, research interest and activities may differ depending on the purpose of the study (Iivari & Venable 2009; Olivier 2004). Although assumptions about reality, knowledge and value underlie any intellectual endeavour, they are, most of the time, implicit. Researchers in the information systems community, however, are forced to consider the most fundamental bases of the socially constructed realities in which they operate as multiple paradigms may apply simultaneously (Vaishnavi & Kuechler 2004).

While qualitative research (refer to section 3.3) is often associated with interpretivism, alternatives in qualitative research do exist. Critical research and positivism are sometimes seen as options, but *pragmatism* has in recent years been touted as a suitable paradigm (Goldkuhl 2011). This philosophical paradigm is rooted in problem-solving and is associated with *action, intervention and constructive knowledge*, with the role of the researcher regarded as that of a change agent (Goldkuhl 2011). In recent years pragmatism has been promoted as a suitable philosophical underpinning for the implementation of the design science research method (refer to section 3.3.1) (Hevner & Chatterjee 2010b; Goldkuhl 2012) in qualitative research.

When research paradigms are compared, the philosophical assumptions that shape each paradigm may be analysed. These assumptions are broken down into four categories, namely ontology, epistemology, methodology and axiology (Vaishnavi & Kuechler 2007). Ontology is the study that describes reality, epistemology explores the nature of knowledge, methodology refers to the way in which knowledge is obtained and axiology is the study of values (Vaishnavi & Kuechler 2007). Table 3.1 provides a summary of the ontological, epistemological, methodological and axiological perspectives in the context of the four research paradigms discussed above, i.e. positivism, interpretivism, critical research and pragmatism (Adebesin et al. 2011; Terre Blanche & Durrheim 2006; Vaishnavi & Kuechler 2004; Goldkuhl 2011). Furthermore, in order to compare relevant research paradigms, Table 3.1 adds the assumptions of pragmatism to the well-documented ontological and epistemological assumptions of the natural and social science research approaches.

Table 3.1: Research paradigm and philosophical assumption summary

(Adebesin et al. 2011; Terre Blanche & Durrheim 2006; Vaishnavi & Kuechler 2004; Goldkuhl 2011)

		Philosophical assumption			
		Ontology	**Epistemology**	**Methodology**	**Axiology**
Research paradigms	**Positivism**	• Single stable reality • Law-like	• Objective • Detached observer	• Experimental • Quantitative • Hypothesis testing	• Truth (objective) • Prediction
	Interpretivism	• Multiple realities • Socially constructed	• Empathetic • Observer subjectivity	• Interactional • Interpretation • Qualitative	• Contextual understanding
	Critical	• Socially constructed reality • Discourse • Power	• Suspicious • Political • Observer constructs versions	• Deconstruction • Textual analysis • Discourse analysis	• Inquiry is value-bound. • Contextual understanding • Researcher's values affect the study.
	Pragmatism	• Symbolic realism • Meaning linked to practical consequences. • Actions and change	• Constructive knowledge • Useful for action and change.	• Inquiry • Data through assessment and intervention.	• Value claims tested and proven in practice. • Value determined in terms of usefulness.

Goldkuhl (2011) indicated that pragmatism has ontological label of “symbolic realism”. He further states that one of the foundational ideas within pragmatism is that the meaning of an idea, concept or artefact is linked to its practical consequences. Epistemologically, pragmatism emphasises constructive knowledge which is useful for action and change (Goldkuhl 2012). In terms of methodology, pragmatism is associated with inquiry as the main type of investigation, with data which is generated being used through both assessment and intervention (Mead & Morris 1938). With regard to axiology, pragmatism views things as value-neutral and suggests that the value of anything is determined solely in terms of its usefulness in achieving a certain end (Barger 2001).

In section 3.5.2, the summary provided in Table 3.1 is used to motivate the research paradigm adopted for the study. Continuing of the discussion of research methodology in this chapter, an overview of research methods is presented in the next section.

3.3 Research methods

This section provides an overview of various research methods that may be used to link a research paradigm to data collection and analysis within a research methodology.

A research method is an investigative approach which moves from the underlying philosophical assumptions to research design and data collection (Myers 1997). Consequently, the choice of research method impacts the way in which data will be collected in the study. Research methods may be classified into two broad categories, namely quantitative and qualitative research (Myers & Avison 2002; Olivier 2004).

Quantitative research attempts to answer questions about relationships among measured variables with the purpose of explaining, predicting and controlling phenomena (Smuts 2011). It is a positivist approach based on exact measurements and makes use of external standards against which all observations can be measured objectively (Olivier 2004). Quantitative information and methods are usually associated with the physical sciences, in which time, density, costs and other measures may be meaningfully expressed as numbers and manipulated mathematically (Mouton 2001).

The underlying assumption of *qualitative research,* in turn, is that multiple realities exist in any given situation and that the perspectives associated with these, including that of the researcher and the individuals that are studied, could be included in the study (Myers 2007). One of the goals is to uncover and discover patterns that will help to explain the phenomenon of interest (De Villiers 2005; Myers & Avison 2002). Qualitative research is also described as involving the use of qualitative data such as interviews, documents, participant observation, pictures and objects to understand and explain social phenomena (Myers 1997).

Paradigmatically, qualitative research could be regarded as an interpretive approach that investigates subjects in their natural surroundings. Qualitative information and methods are also associated with people-orientated research, emphasising words, feelings, the quality of an event or experience (Mouton 2001).

Qualitative researchers conduct their enquiry in their particular field of study and could, for instance, spend time within organisations in their endeavour to document circumstances and employees' interpretations of these circumstances (Smuts 2011). Qualitative research is also used to answer questions about the complex nature of phenomena, usually with the objective of describing and understanding the phenomena from the participant's point of view (Myers 1997; Orlikowski & Baroudi 1991). Table 3.2 (Smuts 2011; Hunter 2004; Myers 1997; Olivier 2004) provides a summary of a number of qualitative research methods.

Qualitative and quantitative approaches are not mutually exclusive and a specific study may require diverse methods of inquiry to cover the whole domain and provide triangulation (Leedy & Ormrod 2001). A qualitative approach can also be employed for investigative work, paving the way for quantitative research as findings from basic research in new areas can be used to formulate hypotheses and questions for quantitative analysis that provides empirical results (De Villiers 2005). A variety of benefits are derived from adopting mixed research methods, as each research method has different assumptions and procedures and different methods may therefore complement one another (Trauth & Jessup 2000; Orlikowski & Baroudi 1991).

Table 3.2: Qualitative research methods

Research methods	Description
Appreciative inquiry	• Process of collaborative enquiry that collects and celebrates the good news stories of a team, organisation or community that serve to enhance cultural identity, spirit and vision.
Design science research	• Involves the analysis of the use and performance of designed artefacts in order to comprehend, explain and improve the behaviour of aspects of information systems.
Action research	• Iterative method for determining current situation of interest and then designing an intervention (Baskerville & Wood-Harper 1996). • Researcher collaborates with practitioners and deliberately intervenes. • Contributes to both research and practice (De Villiers 2005).
Case study	• Explores a single entity or phenomenon bounded by time and activity to establish an understanding of an issue (Olivier 2004; Yin 2003). • Collects detailed information using a variety of data collection methods over a sustained period of time.
Focus group	• Stimulates thinking and creativity through the dynamics of interaction in the context of a small group – similar to a brainstorming session (Olivier 2004).
Ethnography and participant observation	• Researcher studies an intact group of individuals in a natural setting over a specific period of time. • Observes what people are doing as well as what they say they are doing – i.e. the

Research methods	Description
	participant as observer (Olivier 2004; Hunter 2004).
Hermeneutics	• Theory of interpretation of meaning, primarily concerned with the meaning of texts or other human artefacts from the point of view of its author (Olivier 2004).
Systematic review	• Formal and systematic review of literature, developed in order to gather and evaluate the available evidence pertaining to a focused topic (Biolchini et al. 2005).

This study employs design science research as a method and therefore a more detailed description is provided in section 3.3.1, with a discussion of guidelines for design research in 3.3.1.3. The research design for the study incorporates requirements specification as a complementary technique to design science research, and is discussed in section 4.2.1.

3.3.1 Design science research

The term *design research* refers to research into or about *design*, whereas *design science research* (DSR) is research using *design as a research method or technique* (Vaishnavi & Kuechler 2004). Design research is therefore a much broader domain than DSR and spans all design fields. DSR is an application of the constructivist research method and has the defining feature of creating knowledge through “learning through building” (Kuechler & Vaishnavi 2011).

The goal of DSR is to produce a purposeful artefact that addresses an organisational problem, such as improving a business process (Hevner et al. 2004). The DSR method has been widely adopted in the information systems domain, and has also been implicitly used in qualitative research in industrial engineering in recent years. These domains share a pragmatic philosophy rooted in their goal of solving real-world problems and generating knowledge that is valuable for action and change.

DSR is an appropriate technique in situations in which elements of the problem to which this artefact is a relevant solution will only arise during an attempted solution, or where the problem is not completely understood (Kuechler & Vaishnavi 2011). In such situations requirements for the designed artefact only become apparent as the design process progresses. The impact and handling of such requirements are discussed in section 4.2.1. The concept of “learning through building” is especially relevant in cases in which a study is testing previously untested interactions between existing artefact components, or in which new, untried principles are introduced (Kuechler & Vaishnavi 2011).

A further aspect of DSR is rooted in the notion of *design*. Design is inherently an iterative and incremental activity as the evaluation phase provides essential feedback to the construction phase regarding the quality of the design process and the artefact under development (Hevner et al. 2004; Walls et al. 1992). The word “design” implies both a verb and a noun; in other words, both a process and a product (Walls et al. 1992).

Design research must therefore deal with the process of design as well as the product of design, i.e. the artefact.

With regards to the execution of DSR as a research method, an activity framework for the interaction of DSR with research in other scientific paradigms is depicted in Figure 3.1 (Venable 2006b). This framework shows theory building as a central activity related to problem diagnosis, design or technology invention, and technology evaluation. While problem diagnosis and technology evaluation may be undertaken in the empirical domains of natural and particularly behavioural sciences, theory building is the central activity that unifies or binds the framework (Iivari & Venable 2009). Action research, case studies, survey studies and illustrative scenarios are shown as examples of methods to evaluate a new and innovative artefact.

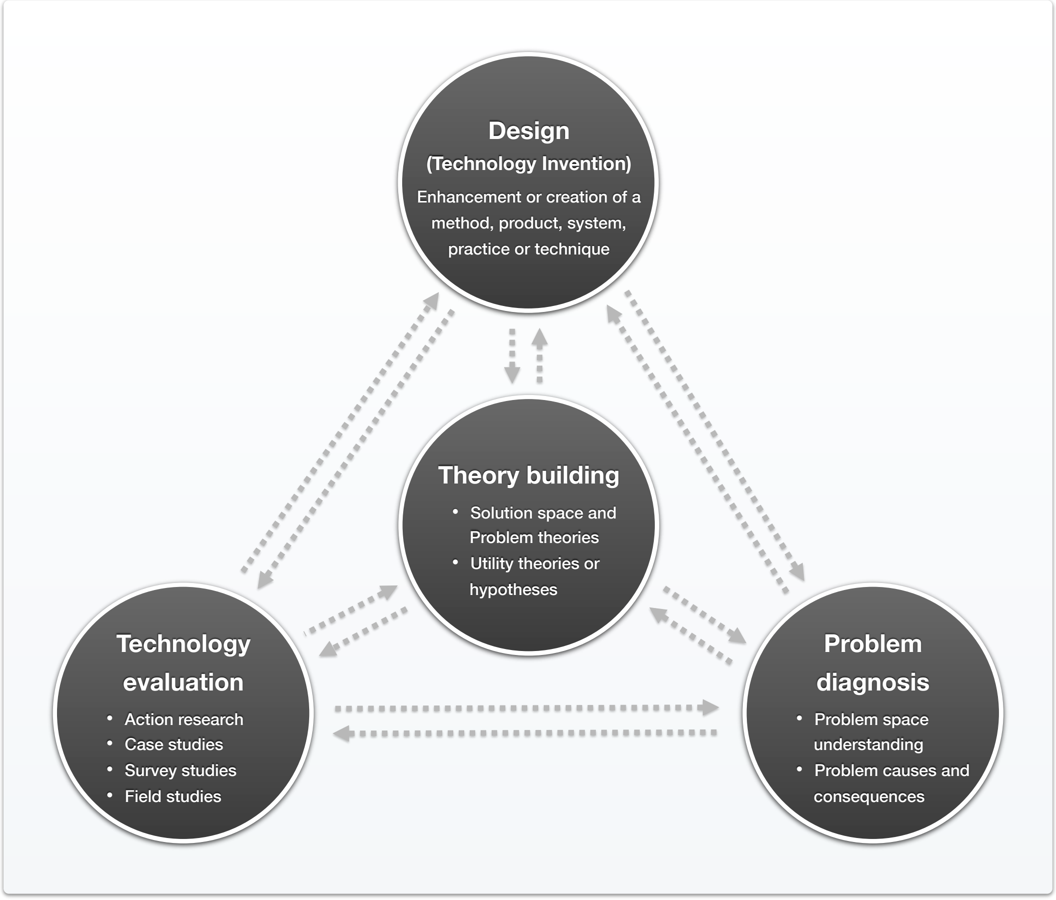


Figure 3.1: An activity framework for DSR

(adapted from Venable 2006b)

3.3.1.1 The three-cycle view of DSR

Hevner (Hevner 2007) presents a three-cycle view of the DSR methodology that spans the application environment, existing knowledge base and the design space as illustrated in Figure 3.2. These three cycles combine to support the notion of "learning through building".

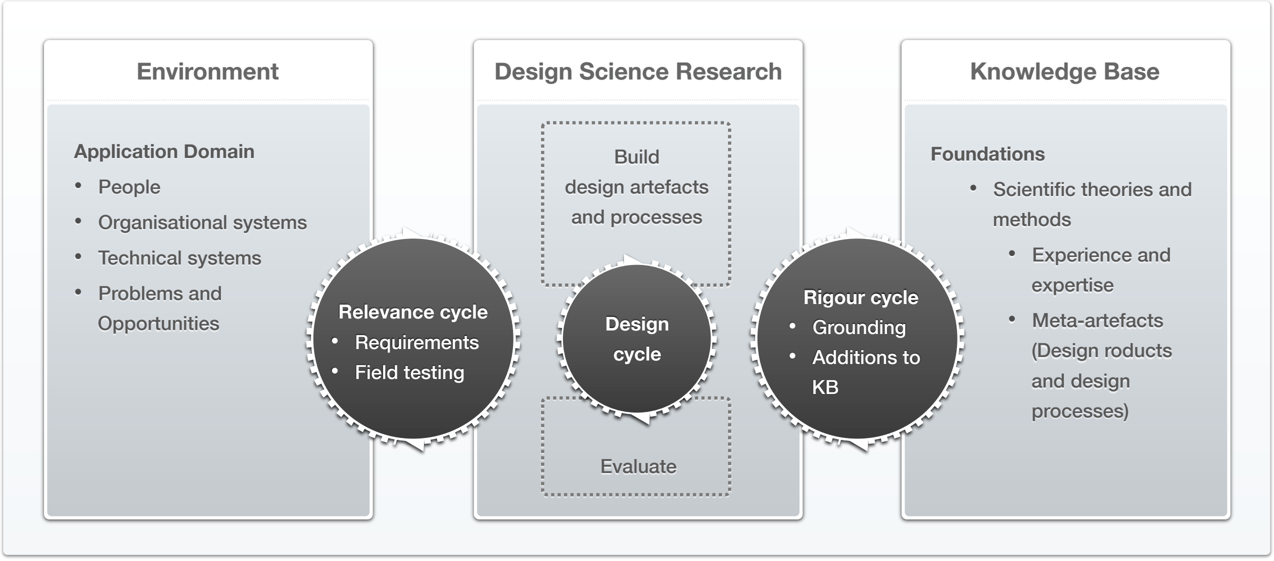


Figure 3.2: Three-cycle view of DSR

(Hevner 2007)

Within Figure 3.2, the first cycle, the *application domain,* represents the ”real world” in which design artefacts interact with people as well as organisational and technical systems. This cycle is also where the practical field problems and opportunities for design artefacts to improve business processes reside. The *knowledge base* (the third cycle), in turn, contains the existing foundations that design science artefacts build on in order to solve problems. The knowledge base includes existing scientific theories and methods, experience and expertise that define the state of the art in the application domain, while also explaining existing artefacts and processes. The middle cycle in Figure 3.2, the *design* space features iterations of building and evaluating to produce artefacts.

The first cycle is the *Relevance* cycle and bridges the contextual and design environments. The Relevance cycle initiates the DSR methodology by identifying the problem and opportunities in the application environment (Hevner 2007). As such the Relevance cycle also delivers the requirements for the artefact from the application domain (Hevner & Chatterjee 2010b). When a design artefact has been successfully evaluated in the Design cycle, the Relevance cycle also introduces the artefact to the application domain for field testing (Hevner 2007). This cycle is therefore imperative in ensuring that the artefact delivered from the design process is relevant in that it addresses an actual problem, but that it is also practically tested before being implemented in the field. The *Rigour* cycle provides the Design cycle with the theoretical grounding that artefacts require in their development (Hevner 2007). This grounding includes scientific theories and methods, experience and expertise, as well as existing design artefacts, processes and meta-artefacts (Hevner & Chatterjee 2010b; Iivari 2007). A successfully evaluated design artefact from the Design cycle is also contributed to the knowledge base via rigorous documentation and publication (Hevner & Chatterjee 2010b; Peffers et al. 2008).

The *Design* cycle is where the artefact is synthesised as a solution to the problem by generating design alternatives and evaluating them against requirements until a satisfactory design is achieved (Simon 1996). The Design cycle iterates between building the artefact and evaluating it, thereby incrementally improving the artefact until it is ready for field testing to ensure its relevance to the application domain (Hevner 2007). The Design cycle receives the requirements for the artefact as input from the Relevance cycle, and the relevant design and evaluation theories from the Rigour cycle (Hevner & Chatterjee 2010b). The construction and evaluation of the artefact is therefore based in both relevance and rigour and should be balanced to provide a pragmatic approach to learning through building (Iivari 2007; Hevner & Chatterjee 2010b).

It is important to note that this three-cycle view of DSR describes a body of work that will in most cases surpass the scope of a doctoral study (Hevner 2012). For such studies, a laboratory test as part of the evaluation phase of the design cycle is often regarded as sufficient to evaluate the research contribution. Upon successful evaluation, the research contribution is contributed to the knowledge base (Hevner 2012). Larger DSR projects in turn may consist of multiple iterations of the entire three-cycle view.

3.3.1.2 The logic of DSR

A general model illustrating design principles and the DSR method for generating and accumulating knowledge is presented in Figure 3.3 (Owen 1998; Vaishnavi & Kuechler 2004).

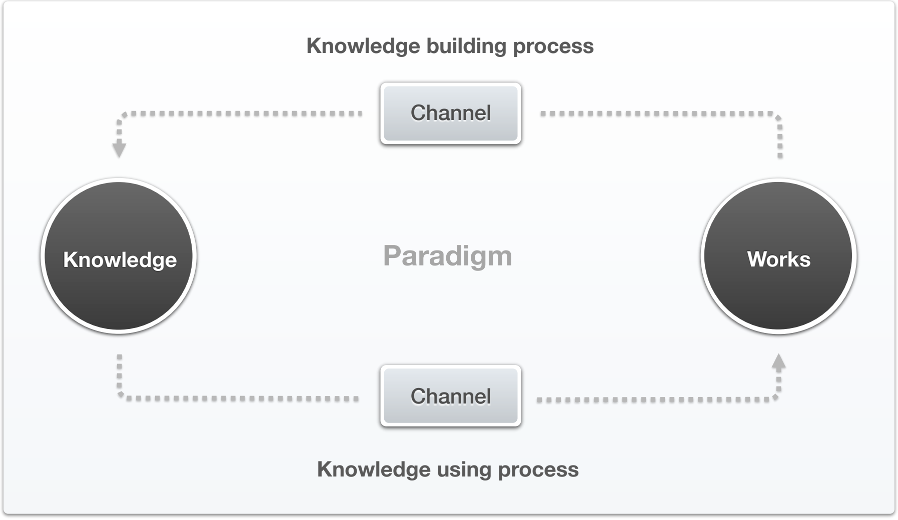


Figure 3.3: A general model for generating and accumulating knowledge
(Owen 1998)

Knowledge is generated and gathered through action and therefore the general model consists of "doing something" and judging the results (Owen 1998). The process is shown as a structured cycle in which knowledge is used to create works, and works are in turn evaluated to build knowledge (Owen 1998). The channels shown in Figure 3.3 are systems of principles and rules under which the discipline operates, and are represented by measures and values that have been developed empirically as ways of knowing as the discipline matured (Owen 1998).

Working toward a concrete process for DSR, Takeda et al. analysed the core reasoning during the general design cycle (Takeda et al. 1990). This was extended by Vaishnavi and Kuechler (Vaishnavi & Kuechler 2004; Vaishnavi & Kuechler 2007) to apply specifically to DSR.

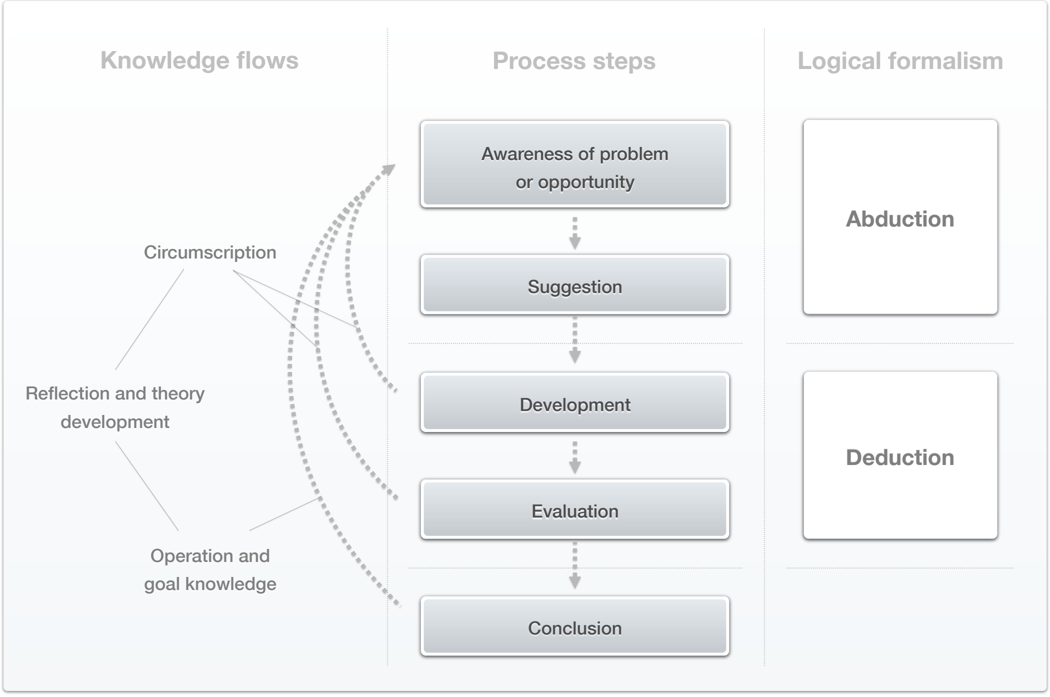


Figure 3.4: Reasoning in the Design Cycle

(Vaishnavi & Kuechler 2004)

One perspective on the execution of DSR studies comprises iterations through five phases (Vaishnavi & Kuechler 2007), as illustrated in Figure 3.4:

1. **Awareness of the problem:** The awareness of the problem emanates from multiple sources through the Relevance cycle and includes new developments in industry or academia. In this step the problem is identified and defined and the output is a formal or informal proposal for a new research effort.
2. **Suggestion:** This phase is closely linked to the awareness phase and is essentially a creative phase in which an artefact which may be a solution to the problem is suggested. This suggestion can be abductively drawn from existing knowledge or theory or by using an appropriate research methodology (Hevner & Chatterjee 2010a). The output is a tentative design of the artefact and could take many forms, including a prototype.

3. **Development:** The tentative design from the suggestion phase is implemented in this phase and the artefact is produced. During artefact production the tentative design may still be further refined and several iterations may be required. This is the only phase of the GDC that requires a constructivist methodology (Vaishnavi & Kuechler 2007).
4. **Evaluation:** Once an implementation of the artefact is ready, it is evaluated according to determine the degree to which it satisfies its requirements (Hevner et al. 2004) and to explain deviations from these. Researchers should select an appropriate evaluation method similar to theory testing (March & Smith 1995), which includes action research, controlled experiments, simulation and scenarios (Hevner & Chatterjee 2010a). The evaluation results, as well as lessons learnt in the development process, may lead to iteration of the Design Cycle starting with an improved awareness of the problem as indicated by the circumscription arrows in Figure 3.4. These cycles of suggestion, development and evaluation continue until the artefact is assessed as sufficient.
5. **Conclusion:** This is the final phase of the Design Cycle and signifies the production of an artefact, the behaviour of which was judged as adequate in the evaluation phase, although not necessarily optimal. In the conclusion phase researchers document the artefact, as well as lessons learnt that may lead to potential further research, for contribution to the body of knowledge and the Design Cycle concludes.

With regard to the epistemology and methodology of DSR, the arrows labelled “Circumscription” and “Operational and goal knowledge” in Figure 3.4 indicate the production of new knowledge. Circumscription is especially important in DSR as it builds the understanding that can only be achieved through the act of construction and is based on the assumption that every fragment of knowledge is valid only in certain conditions (Vaishnavi & Kuechler 2007). The researcher learns when the artefact deviates from theoretical predictions and, by iterating the Design Cycle, contributes valuable knowledge to the understanding of the incomplete knowledge that led to the initial design. This knowledge is packaged in the products of DSR, which may include (Vaishnavi & Kuechler 2004):

- Constructs (the conceptual vocabulary of a problem or solution domain),
- Models (set of propositions or statements articulating the relationships among constructs),
- Methods (set of steps to perform a task; how-to knowledge),
- Instantiations (operationalisation of constructs, models and methods), and
- Better theories (artefact construction as analogous to experimental natural science, coupled with reflection and abstraction).

3.3.1.3 Guidelines for DSR

Hevner et al. (2004) formulate some guidelines to assist researchers in understanding the requirements for effective DSR. The seven guidelines and a short description of each are presented in Table 3.3.

Table 3.3: DSR guidelines

(Hevner et al. 2004)

Guideline	Description
1. Design as an artefact	DSR must produce a viable artefact in the form of a construct, model, method or instantiation.
2. Problem relevance	The objective of DSR is to develop technology-based solutions to important and relevant business problems.
3. Design evaluation	The utility, quality and efficacy of a design artefact must be rigorously demonstrated via well-executed evaluation methods.
4. Research contributions	Effective DSR must provide clear and verifiable contributions in the areas of the design artefact, design foundations and/or design methodologies.
5. Research rigour	DSR relies on the application of rigorous methods in both the construction and evaluation of the design artefact.
6. Design as a search process	The search for an effective artefact requires using available means to reach desired ends while satisfying laws in the problem environment.
7. Communication of research	DSR must be presented effectively both to technology-oriented and management-oriented audiences.

The underlying methodological and epistemological principle of DSR, from which the seven guidelines are derived, is that knowledge and understanding of a design problem and its solution are acquired in the building and application of an artefact. DSR requires the creation of an innovative, purposeful artefact (guideline 1) for a specified problem domain (guideline 2). This artefact must be evaluated (guideline 3) thoroughly as it must be applied to the specified problem. The innovative artefact must provide an improved solution to a known problem (guideline 4). The artefact must be defined properly, be formally represented, coherent and internally consistent (guideline 5). The process by which the artefact is created enables a search process by means of which a problem is identified and a mechanism created to find an effective solution (guideline 6). Lastly, the results of DSR must be communicated effectively (guideline 7), to both technical and managerial audiences (Hevner et al. 2004).

These guidelines for DSR depicted in Table 3.3 will be used as a checklist to verify the integrity of the methodology for the study (refer to section 3.5.4) as well as to validate the contribution of the study in Part 5 of the document.

3.3.1.4 The DSR methodology

Peffers et al. (2008) presents a DSR methodology (DSRM) for the production and presentation of DSR. This methodology contributes a commonly accepted framework for successfully carrying out DSR, as well as a mental model for its presentation (Hevner & Chatterjee 2010a). Outcomes from DSR differ from those of theory testing or interpretive research. This process model provides guidance about what to expect from DSR outputs (Hevner & Chatterjee 2010a).

The DSRM consists of six steps, namely *problem identification and motivation*, *definition of solution objectives*, *design development*, *demonstration*, *evaluation* and *communication* (Peffers et al. 2008).

Activity 1. **Problem identification and motivation:** Define the specific research problem and justify the value of a solution. Since the problem definition will be used to develop an artefact that can effectively provide a solution, it may be useful to atomise the problem conceptually so that the solution can capture its complexity. Justifying the value of a solution accomplishes two things: it motivates the researcher and the audience of the research to pursue the solution and to accept the results and it helps to understand the reasoning associated with the researcher's understanding of the problem. Resources required for this activity include knowledge of the state of the problem and the importance of its solution.

Activity 2. **Definition of solution objectives:** This activity infers the objectives for a solution from the problem definition and knowledge of what is possible and feasible. The objectives can be quantitative, e.g. terms in which a desirable solution would be better than current ones, or qualitative, e.g. a description of how a new artefact is expected to support solutions to problems that have not until now been addressed. The objectives should be inferred rationally from the problem specification. Resources required for this include knowledge of the state of problems and current solutions, if any, and their efficacy.

On this second activity in the DSRM Hevner et al. (2010) notes that identified problems from the first activity do not necessarily directly translate into objectives for the artefact, as the process of design is one of partial and incremental solutions.

Activity 3. **Design and development:** This activity creates the artefact. Such artefacts are potentially constructs, models, methods, or instantiations (each defined broadly) (Hevner et al. 2004) or "new properties of technical, social, and/or informational resources" (Järvinen 2007). Conceptually, a design research artefact can be any designed object in which a research contribution is embedded in the design. This activity includes determining the artefact's desired functionality and its architecture and then creating the actual artefact. Resources required for moving from objectives to design and development include, amongst others, knowledge of theory that can bare a solution.

Activity 4. **Demonstration:** This activity demonstrates the use of the artefact to solve one or more instances of the problem. This could involve its use in experimentation, simulation, case study, proof or other appropriate activity. Resources required for the demonstration include, amongst others, effective knowledge of how to use the artefact to solve the problem. The demonstration of a DSR artefact is discussed in more detail in section 3.3.1.5.

Activity 5. **Evaluation:** This activity observes and measures how well the artefact supports a solution to the problem. This activity involves comparing the objectives of a solution to actual observed results from using the artefact in the demonstration. It requires knowledge of relevant metrics and analysis techniques. Depending on the nature of the problem venue and the artefact, evaluation could take many forms. It could include a comparison of the artefact's functionality with the solution objectives from activity two above, objective quantitative performance measures, such as budgets or production figures, the results of satisfaction surveys, client feedback or simulations. It could also include quantifiable measures of system performance, such as response time or availability. Conceptually, such evaluation could include any appropriate empirical evidence or logical proof. The evaluation of a DSR artefact is discussed in more detail in section 3.3.1.5.

At the end of this activity the researchers can decide whether to iterate back to step three to try to improve the effectiveness of the artefact or to continue on to communication and leave further improvement to subsequent projects. The nature of the research venue may dictate whether such iteration is feasible or not.

Activity 6. **Communication:** This activity communicates the problem and its importance, the artefact, its utility and novelty, the rigour of its design, and its effectiveness to researchers and other relevant audiences, such as practising professionals, when appropriate. In scholarly research publications, researchers might use the structure of this process to structure the paper, just as the nominal structure of an empirical research process (problem definition, literature review, hypothesis development, data collection, analysis, results, discussion and conclusion) is a common structure for empirical research papers. Communication requires knowledge of the disciplinary culture.

3.3.1.5 Evaluation of DSR

Evaluation is a central and essential activity in conducting rigorous DSR (Venable et al. 2012), with March and Smith (1995) identifying "build" and "evaluate" as two key DSR activities. In DSR, evaluation is concerned with examining DSR outputs, including design artefacts (March & Smith 1995) and design theories (Walls et al. 1992; Gregor et al. 2007). Hevner et al. (2004) describe evaluation as "crucial", and in their third guideline for DSR (refer to section 3.3.1.3) they state that "the utility, quality, and efficacy of a design artefact must be rigorously demonstrated via well-executed evaluation methods".

Evaluation provides evidence that a new design artefact or theory developed through DSR achieves the purpose for which it was designed, thereby putting the “science” in “design science” (Venable et al. 2012). Venable et al. (2012) identify the primary purpose of evaluation in DSR to be to “evaluate an instantiation of a designed artefact to establish its utility and efficacy (or lack thereof) for achieving its stated purpose”. They furthermore remark that, given this purpose, rigorous evaluation in DSR spans both the “demonstration” and “evaluation” activities of the DSRM (refer to section 3.3.1.4) contributed by Peffers et al. (2008).

Pries-Heje et al. (2008) present a simple 2-by-2 framework for deriving the design of the evaluation of a DSR project (refer to Table 3.4). The first dimension of this framework contrasts artificial versus naturalistic evaluation. *Artificial* evaluation includes laboratory experiments, field experiments, simulations, criteria-based analysis, theoretical arguments, and mathematical proofs (Venable 2006a). *Naturalistic* evaluation explores the performance of a solution technology in its real environment i.e., within the organisation (Venable 2006a). By performing evaluation in a real environment, i.e. real people, real systems, and real settings (Sun & Kantor 2006), naturalistic evaluation embraces all of the complexities of human practice in real organisations. The second dimension of the framework presented by Pries-Heje et al. (2008) contrasts *ex ante* with *ex post* evaluation. Ex post evaluation refers to the evaluation of an instantiated artefact, while ex ante evaluation is regarded as the evaluation of an uninstantiated artefact, such as a design or model (Pries-Heje et al. 2008).

Table 3.4: Strategic DSR Evaluation Framework

(adapted from Pries-Heje et al. 2008)

	Ex ante	**Ex post**
Naturalistic		
Artificial		

This simple framework is extended by Venable et al. (2012) in order to form a DSR Evaluation *Strategy* Selection Framework (refer to Table 3.5). This framework helps to identify a particular DSR evaluation strategy (or combination of strategies) that is appropriate and supports decision making about which particular evaluation method(s) are appropriate to achieve those strategies (Venable et al. 2012).

As depicted in Table 3.5, the framework maps various aspects of the context of the evaluation of a DSR project to the framework shown in Table 3.4. Relevant aspects of the context of the DSR evaluation that serve as the starting point of the design of the DSR evaluation, include (Venable et al. 2012):

Table 3.5: DSR Evaluation Strategy Selection Framework

(adapted from Venable et al. 2012)

		Ex ante	**Ex post**
		• Formative • Lower build cost • Faster • Evaluate design, partial prototype of full prototype • Less risk to participants (during evaluation) • Higher risk of false positive	• Summative • Higher build cost • Slower • Evaluate instantiation • Higher risk to participants (during evaluation) • Lower risk of false positive
Naturalistic	• Many diverse stakeholders • Substantial conflict • Socio-technical artefacts • Higher cost • Longer time (slower) • Organisational access needed • Artefact effectiveness evaluation • Desired rigor: “proof of the pudding” • Higher risk to participants • Lower risk of false positive (safety critical systems)	• Real users, real problem, and somewhat unreal system • Low to medium cost • Medium speed • Low risk to participants • Higher risk of false positive	• Real users, real problem, and real system • Highest cost • Highest risk to participants • Best evaluation of effectiveness • Identification of side effects • Lowest risk of false positive (safety critical systems)
Artificial	• Few similar stakeholders • Little or no conflict • Purely technical artefacts • Lower cost • Less time (faster) • Desired rigor: control of variables • Artefact efficacy evaluation • Less risk during evaluation • Higher risk of false positive	• Unreal users, problem and/or system • Lowest cost • Fastest • Highest risk of false positive regarding effectiveness	• Real system, unreal problem and possibly unreal users • Medium to high cost • Medium speed • Low to medium risk to participants

- the different purposes of evaluation in DSR,
- the characteristics of the output to be evaluated,
- the type of output to be evaluated, and
- the specific goals that must be balanced in the design of the evaluation of a DSR project.

In using the framework depicted in Table 3.5 to formulate a DSR evaluation strategy or strategies, it is important to prioritise the different criteria, as they are likely to conflict (Venable et al. 2012). Identifying relevant, higher priority criteria supports identifying an appropriate quadrant or quadrants, i.e. the relevant shaded cell(s) in Table 3.5. Venable et al. (2012) note that selecting a single quadrant may not be the best strategy, and that a hybrid strategy consisting of more than one quadrant may be used to resolve conflicting goals.

The DSR Evaluation Strategy Selection Framework is complimented by a DSR Evaluation *Method* Selection Framework (refer to Table 3.6). This Method Selection Framework is expressed as a mapping of DSR evaluation strategies to relevant evaluation methods. Once a high level strategy is selected for the evaluation design, i.e. which of the quadrants in Table 3.5 will be used for the evaluation, the particular evaluation research method(s) are selected and the evaluation is designed in detail. Venable et al. (2012) note that the mapping depicted in Table 3.6 may omit some potential evaluation methods and other evaluation methods may be developed or adopted for a DSR evaluation.

Table 3.6: DSR Evaluation Method Selection Framework

(adapted from Venable et al. 2012)

	Ex ante	**Ex post**
Naturalistic	• Action research • Focus group	• Action research • Case study • Illustrative scenario • Focus group • Participant observation • Ethnography • Phenomenology • Survey (qualitative or quantitative)
Artificial	• Mathematical or logical proof • Criteria-based evaluation • Laboratory experiment • Computer simulation	• Mathematical or logical proof • Laboratory experiment • Role playing simulation • Computer simulation • Field experiment

The combination of these two frameworks bridges the gap between the contextual factors relevant to the DSR evaluation and appropriate means (methods) to evaluate the DSR artefacts (Venable et al. 2012). The discussion in this section is summarised in a simple method for designing the evaluation component(s) of a DSR project, which accompanies the frameworks (Venable et al. 2012):

1. Analyse the requirements for the evaluation to be designed.

2. Map the requirements to one or more of the dimensions and quadrants in the Strategy Selection Framework (refer to Table 3.5).
3. Select an appropriate evaluation method or methods that align with the chosen strategy quadrant(s), using the Method Selection Framework (refer to Table 3.6).
4. Design the evaluation in more detail.

3.4 Data collection and analysis

Each of the research methods discussed in section 3.3 employs techniques for collecting empirical data (Myers 1997; Leedy & Ormrod 2001). It is possible to distinguish between primary and secondary sources of data (Hofstee 2006). Primary data is data that is unpublished and has been collected directly from people, research participants or the organisation under investigation. Secondary data refers to all materials that have previously been published, e.g. books, journals and articles (Myers 1997). Both primary and secondary data can be of either a quantitative or qualitative nature (Olivier 2004), and it is the role of the researcher to evaluate, organise and synthesise research results in order to report them as a cohesive whole (Leedy & Ormrod 2001).

An important consideration when collecting data is the degree to which the researcher participates or influences the process of data collection. For instance, when interviewers clarify the questions they ask they may influence the answers or where productivity measurement is conducted the presence of the researcher may influence the results (Olivier 2004).

Two primary techniques for collecting and evaluating qualitative data were used in this study, namely a qualitative systematic review and interviews.

3.4.1 Qualitative systematic review

A literature review creates a firm foundation for advancing knowledge by facilitating theory development (Webster & Watson 2002). A literature review can be defined as a method for "identifying, evaluating and synthesising the existing body of completed and recorded work produced by researchers, scholars and practitioners" and offers numerous opportunities to engage and interact with theory (Booth et al. 2012).

Booth et al. (2012) identify eleven different types of review, one of which is the *qualitative systematic review*. A qualitative systematic review integrates and compares findings from qualitative studies, with the objective of finding themes or constructs in or across individual studies. Such analyses may include conceptual models and can combine constructs across published theories with different labels, removing redundancy and overlap (Booth et al. 2012; Damschroder et al. 2009).

Contrary to the usual process of unsystematic literature review that is conducted at the beginning of any research study, a systematic review is developed in a formal and systematic way (Biolchini et al. 2005). The research process of a systematic review follows a well defined sequence of methodological steps, according to an aprioristically developed protocol. These methodological steps are outlined by Khan et al. (2003) as follows:

Step 1. **Framing questions for a review:** The problems to be addressed by the review should be specified in the form of clear, unambiguous and structured questions before beginning the review work.

Step 2. **Identifying relevant work:** The search for studies should be extensive. Multiple resources (both computerised and printed) should be searched without language restrictions. The study selection criteria should flow directly from the review questions and be specified a priori.

Step 3. **Assessing the quality of studies:** Study quality assessment is relevant to every step of a review. Question formulation (Step 1) and study selection criteria (Step 2) should describe the minimum acceptable level of design. Selected studies should be subjected to a more refined quality assessment by use of general critical appraisal.

Step 4. **Summarising the evidence:** Data synthesis consists of tabulation of study characteristics, quality and effects. Exploration of heterogeneity and its sources should be planned in advance (Step 3).

Step 5. **Interpreting the findings:** Any issues highlighted during the four previous steps should be considered when interpreting the findings. The risk of publication bias and related biases should be explored. The evidence data should be normalised in such a way as to make results from different studies comparable, in terms of their magnitude of effect, even when they are presented in diverse ways but related to compatible concepts (Biolchini et al. 2005). Any recommendations should be graded by reference to the strengths and weaknesses of the evidence.

This research instrument is constructed around a central issue, which represents the core of the investigation, and which is expressed by using specific concepts and terms (Khan et al. 2003). These concepts and terms must be addressed towards information related to a specific, pre-defined, focused, and structured question. The methodological steps above, the strategies to retrieve the evidence, and the focus of the question are explicitly defined, so that other researchers can reproduce the same method and also be able to judge the adequacy of the themes or constructs that were found.

3.4.2 Illustrative scenarios

Illustrative scenarios are related to case studies and field studies in being a naturalistic evaluation method that is suited to DSR. Peffers et al. (2012) state that “illustrative scenarios apply the artefact in a synthetic or real-world situation to demonstrate its utility”. They further state that “case studies implement the artefact in a real-world situation to evaluate not only its utility, but also its effect on its environment”. Both of these evaluation methods therefore relate to the Relevance cycle of the three-cycle view on DSR (Hevner et al. 2004). There is however a difference in focus between an illustrative scenario and a case study, with an illustrative scenario concentrating on the demonstration and evaluation of the utility of the artefact. Case studies typically require longer timeframes than illustrative scenarios to execute, since more time is required for the effect of a DSR artefact to be visible on the participating organisations.

Illustrative scenarios involve multiple, diverse stakeholders and are suited to the evaluation of socio-technical artefacts, i.e. artefacts for which interaction with humans is required to bring about their utility (Venable et al. 2012). When the illustrative scenario method is conducted in a real-world situation, i.e. with real users, real systems and real problems (Sun & Kantor 2006), organisational access is required. An illustrative scenario limits the risk to participating organisations, since they are not obliged to implement the particular artefact (Venable et al. 2012).

3.4.3 Interviews

A research interview is an “interview where knowledge is constructed in the interaction between the interviewer and the interviewee” (Kvale 2008). Various types of interviews can be identified, e.g. in-depth interviews, semi-structured interviews and structured interviews (Hesse-Biber & Leavy 2010).

An in-depth interview is employed when the interviewer seeks knowledge from the interviewee’s point of view. In such an interview the questions are open-ended and the interview’s degree of structure depends on the extent to which the interviewer has a specific agenda. A semi-structured interview consists of specific research questions, selected by the interviewer to guide the discussion, but used on discretion. The structured interview starts with a pre-defined set of questions posed to every interviewee, and if the participant strays away from the topic at hand the interviewer will guide the conversation back to the interview questions (Kvale 2008; Hesse-Biber & Leavy 2010).

There are several advantages and disadvantages to the use of an interview for data-gathering (Bentley et al. 2007). The main disadvantage being that interviews are time-consuming, but with the prime advantage of allowing for communicative interaction between the interviewer and the interviewee in obtaining a richer data set than with a questionnaire (De Vries 2012).

3.5 Methodology and research design

This section details the specific methodology and research design adopted for this study. As an overview, the research objective and research question are stated in section 3.5.1, followed by the research paradigm in section 3.5.2. Section 3.5.3 details the research design and section 3.5.4 shows how this research design adheres to the guidelines for DSR. Section 3.5.5 focuses on data collection and the approach to data analysis.

3.5.1 Research objective and question

As described in section 1.3 and based on the context of this study, the research objective and primary research question addressed in this study are presented in Table 3.7.

Table 3.7: Research objective and primary research question

Code	Research objective	Code	Research question
RO	**To develop a reference architecture for use in the engineering of IKNs.**	**PRQ**	**How can a reference architecture for use in the engineering of IKNs be developed?**

3.5.2 Research paradigm

The research philosophy adopted for this study is *pragmatism*. Pragmatism supports the DSR method by calling for the creation of innovative artefacts to solve real-world problems (Simon 1996; Hevner & Chatterjee 2010b; Goldkuhl 2012). To confirm that pragmatism is a suitable philosophy to adopt for this study in industrial engineering, the philosophical assumption and research paradigm summary table presented in section 3.2 (refer to Table 3.1) was revisited. The scope of this study as described in chapter 1 was considered against every cell of the table to determine suitability of the particular philosophical assumption of the particular research paradigm. The results of this verification are depicted in Table 3.8, with cells in which the content applied to this study shaded in green. As indicated in Table 3.8 with shading, it is possible to motivate that pragmatism is the appropriate philosophical paradigm for this research study.

Ontologically, pragmatism assumes a symbolic approach to realism and that actions can change reality. This speaks to the principle that pragmatic research starts with a problem, which suits this study as it inherently addresses the problem of IKNs having to adapt in an ever-changing environment without the help of a reference architecture to provide input in their engineering. From an *epistemological* perspective, pragmatism generates constructive knowledge that is useful for action and change, as is seen in the way in which knowledge generated in a DSR cycle influences ensuing cycles. The pragmatic *methodology* of knowledge creation is one of inquiry and data generation through assessment and intervention. This is true for this study, since it is driven by research objectives and questions of which the answers are synthesised

Table 3.8: Research philosophy suitability verification

	Philosophical assumption			
	Ontology	**Epistemology**	**Methodology**	**Axiology**
Positivist	• Single stable reality. • Law-like.	• Objective. • Detached observer.	• Experimental. • Quantitative. • Hypothesis testing.	• Truth (objective). • Prediction.
Interpretive	• Multiple realities • Socially constructed	• Empathetic • Observer subjectivity	• Interactional • Interpretation • Qualitative	• Contextual understanding.
Critical	• Socially constructed reality. • Discourse. • Power.	• Suspicious. • Political. • Observer constructs versions.	• Deconstruction. • Textual analysis. • Discourse analysis.	• Inquiry is value bound. • Contextual understanding. • Researcher's values affect the study.
Pragmatism	• Symbolic realism. • Actions and change.	• Constructive knowledge. • Useful for action and change.	• Inquiry. • Data through assessment and intervention.	• Value claims tested and proven in practice. • Value determined in terms of usefulness.

to solve the research problem. From an *axiological* perspective, the value of a reference architecture for IKNs is verified through rigorous evaluation of the artefact.

Goldkuhl (2011) states that pragmatism is concerned with *action and change and the interplay between knowledge and action*. Pragmatism is therefore appropriate as a basis for research approaches intervening in the world as DSR does through the construction of artefacts. The primary objective of this study is to develop a reference architecture for IKNs that will serve as a reference for the engineering of implementation-specific architectures. This would arguably increase the ability of IKNs to adapt to changes in their environment, which constitutes the real-world intervention that Goldkuhl (2011) refers to. The IKN reference architecture artefact was furthermore developed and tested through multiple cycles, since the true nature of the requirements for such an artefact could only be known by engaging in its incremental development. This intention fits closely with the constructive and inquiry-based principles of pragmatism, as shown by the assessment in Table 3.8.

3.5.3 Research design

This section presents the practical plan of action followed in this study in order to solve the research problem.

The design, development and evaluation of a new artefact in a given organisational context affords the opportunity to apply both empirical and qualitative methods (Hevner et al. 2004). The rich phenomena that emerge from the interaction of people, organisation and technology, however, need to be qualitatively assessed to yield an adequate understanding of these phenomena for theory development or problem-solving (Myers 2007). This study is therefore firmly rooted in the qualitative research discipline, with DSR as research method.

The DSR method involves a rigorous process to design artefacts in order to solve observed problems, make research contributions, evaluate designs and communicate the results to appropriate audiences. As such artefacts may include constructs, models, methods and instantiations (Peffers et al. 2008; Hevner et al. 2004), the DSR method is deemed a suitable method for the development of a reference architecture for IKNs. DSR has also recently been identified as an appropriate candidate for being the main research methodology in the discipline of EE, since the discipline is by nature about designing (Dietz et al. 2013).

True to the pragmatic philosophy of this study, the research objective for this study is set up in such a way that achieving the objective will result in solving the research problem. Achieving the objective of developing a reference architecture that can be used in the engineering of IKNs is therefore the expected result of this study. In order to achieve this objective, a suitable research design that facilitates the construction of a reference architecture for IKNs is required.

Given the pragmatic nature of the stated research objective, the research design should allow for the constructivist gathering and application of requirements specifications and design information throughout the development cycle. The research design should therefore allow for iterative improvement of the artefact when such information regarding requirements and subsequent design elements enter the equation, thereby progressing towards an artefact that can be evaluated as a candidate solution. Where relevant inputs from the existing knowledge base from both a theoretical and a practical perspective are available, the research design should also take advantage of these.

The research design adopted for the study is based on the DSRM presented by Peffers et al. (2008), as this mental model for the conduct and presentation of DSR helps to enable effective research of this kind (Hevner & Chatterjee 2010a). The research design features the five basic steps of the DSRM and includes four embedded design cycles that contribute to incrementally developing the artefact (refer to Figure 3.5). The discussion of the research design in this section, however, will be limited to a presentation of the design's approach to research, without detailed discussion of the content of each design cycle. These

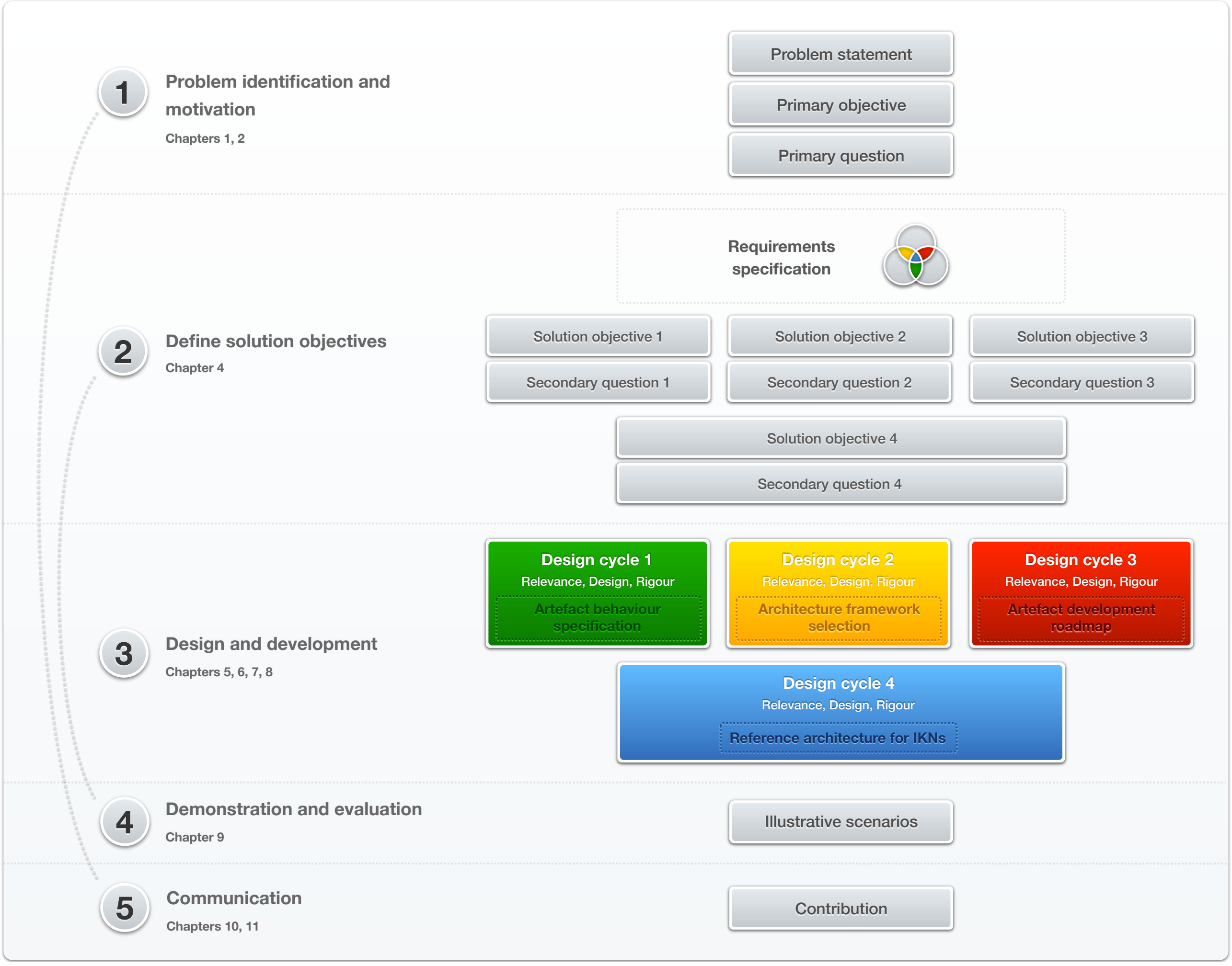


Figure 3.5: Research design

design cycles are each discussed in detail in chapter 4 as part of the discussion on the definition of objectives for the artefact.

Activity 1. *Problem identification and motivation* – In this first activity the following *research problem* for the study is identified and motivated:

No reference architecture exists for use in the engineering of IKNs.

The value of a solution to this problem is confirmed through a series of personally conducted exploratory interviews, which motivates the study to pursue a solution and contributes to understanding the complexity of the problem.

Given the pragmatic stance taken in the study, this problem leads to the following *research objective* for solving it:

RO: To develop a reference architecture for use in the engineering of IKNs.

This research objective is accompanied by the following *primary research question*:

PRQ: How can a reference architecture for use in the engineering of IKNs be developed?

Answering this question leads to achieving the research objective stated above and thereby solving the research problem. Resources required for this activity include knowledge of the state of the problem and the importance of its solution, and is therefore informed by Part 1 of this document which features chapters 1 and 2.

This first activity corresponds to the fifth activity in the research design in which the success, utility and value of the solution to the problem are communicated (refer to Figure 3.5).

Activity 2. *Define the objectives for a solution* – The objectives for a solution to the problem identified in activity 1 are defined in this activity and should be inferred rationally from the problem statement. In order to achieve this, a process of qualitative requirements specification for the solution artefact is performed, resulting in the following *solution objectives*:

SO1: To exhibit the desired behaviour of a reference architecture in the engineering of IKNs.

SO2: To employ an architecture framework that is suitable to the engineering of IKNs.

SO3: To implement the selected architecture framework in a way that enables the artefact to function as a reference architecture.

SO4: To function as a reference architecture in the engineering of IKNs through the implementation of a suitable architecture framework.

These solution objectives are achieved by answering the following accompanying *secondary research questions*:

SRQ1: What is the desired behaviour of a reference architecture in the engineering of IKNs?

SRQ2: Is an architecture framework that is suitable to the engineering of IKNs available?

SRQ3: How can the selected architecture framework be implemented in a way that enables the artefact to function as a reference architecture?

SRQ4: How can an artefact that functions as a reference architecture in the engineering of IKNs be developed by implementing a suitable architecture framework?

These secondary research questions reflect the pragmatic approach of the study in that they are seen as worthwhile to answer, given their ability to contribute to solving the research problem.

This activity of requirements specification and the subsequent identification of secondary research questions and objectives are presented in Part 2 which features chapter 4 in which the planning of the third activity is also done.

This second activity corresponds to the fourth activity in the research design in which the solution's ability to achieve its objectives is demonstrated and evaluated (refer to Figure 3.5).

Activity 3. *Design and development* – In this activity the solution artefact for the problem is created, which for this study is a reference architecture for IKNs. The activity includes designing the desired functionality and architecture of the artefact to ensure that the solution objectives are achieved, after which the artefact is constructed. This process of design and development is done through a series of four DSR design cycles that incrementally move the artefact from objectives to design and development and is discussed in Part 3.

The *first cycle* is driven by SO1 and SRQ1 (refer to activity two) and delivers an artefact behaviour specification that describes the desired behaviour of the solution in certain scenarios. This cycle is discussed in chapter 5.

The *second cycle* is driven by SO2 and SRQ2 (refer to activity two) and delivers an architecture framework selection which is suitable in the engineering of IKNs. This cycle is discussed in chapter 6.

The *third cycle* is driven by SO3 and SRQ3 (refer to activity two) and delivers an artefact development roadmap detailing how the selected architecture framework can be implemented to enable the artefact to function as a reference architecture. This cycle is discussed in chapter 7.

The *fourth cycle* is driven by SO4 and SRQ4 (refer to activity two) and constructs the solution artefact itself. This is done according to the artefact behaviour specification by implementing the selected architecture framework as set out by the artefact development roadmap in the context of the engineering of IKNs. This cycle is discussed in chapter 8, and includes the development of reference models for IKNs based on the results of a qualitative systematic review.

Activity 4. *Demonstration and Evaluation* – This fourth activity in the research design for the study combines activities four and five in the DSRM (refer to section 3.3.1.4). The reason for this is the scope and implementation of the demonstration and evaluation technique chosen for this study. An approach that employs illustrative scenarios is used to demonstrate and evaluate the artefact, and specific care is taken to demonstrate and evaluate the artefact in terms of each of its objectives. The detailed design of this evaluation is discussed in section 3.5.5.2.

This fourth activity therefore corresponds to, and is checked against, the second activity in the research design in which these solution objectives were defined (refer to Figure 3.5). The subsequent evaluation phase of this activity then compares these observed results with the solution objectives from activity two to ascertain how well the artefact supports a solution to the problem.

At the end of this activity a decision is made about whether to iterate back to activity 3 to try and improve the effectiveness of the artefact, or to continue the communication of the current artefact as a satisfactory solution and leave further improvement to subsequent projects. This activity is presented in Part 4, which features chapter 9.

Activity 5. *Communication* – This activity reflects on the artefact, its novelty and utility, the rigour of its design and its effectiveness. The focus is therefore on illuminating and validating the overall research contribution and conclusion of the study. Part 5 of this document, which features chapters 10 and 11, captures this activity.

This fifth activity corresponds to, and is checked against, the first activity in the research design in which the problem as well as the opportunity for making a research contribution were communicated (refer to Figure 3.5).

3.5.4 Adherence to DSR guidelines

Table 3.9 demonstrates how the research design for this study adheres to the guidelines for DSR as presented in section 3.3.1.3. As depicted in Table 3.9, the research design adheres to the guidelines for DSR proposed by Hevner (2004). Furthermore, these guidelines assisted with the identification of an appropriate approach and evaluation methods.

Table 3.9: Adherence to DSR guidelines

(Hevner et al. 2004)

Guideline	Description	Adherence
1. Design as an artefact	DSR must produce a viable artefact in the form of a construct, model, method or instantiation.	• The reference architecture for IKNs provides a purposeful contribution within the domain of innovation networks. • As a reference architecture, the artefact is a construct containing several reference models and is therefore a valid DSR artefact.
2. Problem relevance	The objective of DSR is to develop technology-based solutions to important and relevant business problems.	The reference architecture for IKNs is used as a reference when engineering case-specific architecture descriptions for IKNs, thereby improving their ability to adapt to change.
3. Design evaluation	The utility, quality and efficacy of a design artefact must be rigorously demonstrated via well-executed evaluation methods.	The methodology includes a rigorous evaluation of the reference architecture for IKNs through a series of illustrative scenarios.
4. Research contributions	Effective DSR must provide clear and verifiable contributions in the areas of the design artefact, design foundations and/or design methodologies.	The reference architecture for IKNs enhances the current state of IKN engineering and development. The artefact extends the knowledge base, but also applies existing knowledge through extensive use of existing constructs and models in the development of the reference models. Refer to Part 5 for an in-depth discussion of research contributions.
5. Research rigour	DSR relies on the application of rigorous methods in both the construction and evaluation of the design artefact.	• A rigorous approach to considering research questions through requirements specification leads to the identification of clear objectives for the reference architecture for IKNs. • These requirements and objectives are also central to the approach for evaluating the artefact.
6. Design as a search process	The search for an effective artefact requires using available means to reach desired ends while satisfying laws in the problem environment.	• The methodology features four iterations of the design cycle as knowledge gained through the construction of the artefact is utilised to make design choices.

Guideline	Description	Adherence
		• The study demonstrates the identification of available means (existing constructs and models) that may address desired ends, posed by specifying requirements for a reference architecture for IKNs.
7. Communication of research	DSR must be presented effectively both to technology-oriented and management-oriented audiences.	• The artefact evaluation shows that the reference architecture for IKNs provides sufficient detail to architecture practitioners that utilise it as a reference when engineering case-specific architecture descriptions for IKNs. • The management-level benefits of using the reference architecture for IKNs as a reference to develop case-specific architecture descriptions for IKNs are articulated amongst the research contributions of this study.

3.5.5 Data collection and analysis

This section describes that particular approaches to data collection and analysis that forms part of the research design and overall methodology adopted in the study.

3.5.5.1 Qualitative systematic review

A research instrument based on the qualitative systematic review method (refer to section 3.4.1) is employed to collect and analyse data for the development of the reference architecture for IKNs in the fourth design cycle of the research design. This review integrates and compares findings from various qualitative studies, with the objective of finding themes or constructs in or across individual studies. The resulting data is then interpreted in order to generate the reference models that populate the reference architecture for IKNs. The steps followed in the qualitative systematic review in the fourth design cycle of the research design employed in this study is as follows:

Step 1. **Framing questions for a review:** The questions for the review were based on the abstractions (columns) of the Zachman framework (refer to section 2.7.2). The following questions are therefore asked when a data source is reviewed:

1. *"What?"*, i.e. which generalisable inventory sets are significant to the construction of IKNs?
2. *"How?"*, i.e. which generalisable process flows are significant to the construction of IKNs?

3. *"Where?"*, i.e. which generalisable distribution networks are significant to the construction of IKNs?
4. *"Who?"*, i.e. which generalisable responsibility assignments are significant to the construction of IKNs?
5. *"When?"*, i.e. which generalisable timing cycles are significant to the construction of IKNs?
6. *"Why?"*, i.e. which generalisable motivation intentions are significant to the construction of IKNs?

Step 2. **Identifying relevant work:** An extensive search for relevant to studies is conducted, with multiple sources (both computerised and printed) consulted. Since the concept of IKNs is a relatively recent one, it was to be expected that limited resources that explicitly address this subject were found. This is compounded by the specific term of "integrated knowledge networks" currently primarily being adopted within the Department of Industrial Engineering, Stellenbosch University. The search for relevant studies is therefore expanded to include "collaborative networks", "innovation networks", "knowledge networks", "knowledge" and "innovation". Titles and abstracts of found studies are analysed on a case by case basis to determine their applicability. Heterogeneity amongst data sources were also considered by ensuring that studies by a range of authors were included.

Step 3. **Assessing the quality of studies:** The quality of potential studies to review is assessed, including the clarity of their research design and articulation of findings. The level of acceptance of studies within the academic community as expressed in number of citations according to scholarly search engines is also taken into account. Upon assessing the quality of studies and discarding studies that were deemed as unsuitable to the review, a total of 100 studies were selected for review.

Step 4. **Summarising the evidence:** Data synthesis consists of tabulation of the various answers provided by each study to the review questions framed in step 1 above. Data is furthermore sorted into evidence of a construction component of IKNs being identified, i.e. *identification data*, versus being defined more detail, *definition data*.

Step 5. **Interpreting the findings:** The evidence data is normalised according to the review questions, i.e. the abstractions of the Zachman framework, to make the results from different studies comparable. This enables the unification of concepts that are expressed in diverse ways in different studies. The evidence data are graded by reference to the strengths and weaknesses of the evidence from the studies reviewed. The resulting evidence data represents generalisable elements that are significant to IKNs, and serve as inputs for the development of reference models that populate the reference architecture for IKNs.

This research instrument is constructed around the central issue of the construction of IKNs, which represents the core of the investigation, and which is expressed by using specific concepts and terms (Khan et al. 2003). The methodological steps above, the strategies to retrieve the evidence, and the focus of the questions are explicitly defined, so that other researchers can reproduce the same method and also be able to judge the adequacy of the themes or constructs that were found. An example of the review of a study and the synthesis of the collected evidence is provided in Appendix B.

3.5.5.2 Evaluation

The evaluation of the reference architecture for IKNs is designed according to the strategy and method selection frameworks presented by Venable et al. (2012) (refer to section 3.3.1.5). This evaluation design is executed in the fourth activity of the research design employed in this study, and is depicted in Table 3.10.

Table 3.10: Evaluation design for reference architecture for IKNs

		Ex ante	**Ex post**
			• Summative • Slower • **Evaluate instantiation**
Naturalistic	• Many diverse stakeholders • **Socio-technical artefacts** • Longer time (slower) • Organisational access needed • Artefact effectiveness evaluation • Desired rigor: “proof of the pudding” • Higher risk to participants		• **Real users, real problem, and real system** • Best evaluation of effectiveness • **Illustrative scenarios**
Artificial			

This evaluation strategy emphasises that the reference architecture for IKNs is a socio-technical artefact, i.e. an artefact with which humans must interact to provide its utility. The evaluation therefore takes a naturalistic approach that includes many stakeholders and evaluates the effectiveness of the artefact with a high level of rigor. This approach does however require a longer timeframe and requires organisational access. The evaluation strategy furthermore emphasises that an instantiation of the reference architecture for IKNs will be evaluated, and not merely its design. The evaluation may therefore be classified as “ex post” and provides a summative view on the performance of the artefact. This approach is slower, since the evaluation can only be performed once the reference architecture has been constructed. The evaluation strategy therefore maps to the top right quadrant of the framework depicted in Table 3.10. From this strategy, it is possible to motivate the selection of either case studies or illustrative scenarios as the appropriate evaluation methods. Both of these methods employ real users, real problems and real systems to evaluate an artefact (Sun & Kantor 2006).

Peffers et al. (2012) describe a case study as the "application of an artefact to a real-world situation, evaluating its effect on the real-world situation". The execution of a case study requires extensive cooperation from the organisations that are involved. In the case of an artefact such as the reference architecture for IKNs, an extended timeframe is also required in order for the effect of the artefact to be observed on case study networks. For rigorous case studies to be conducted to evaluate the reference architecture for IKNs, it would be required that the participating organisations instantiate the particular architecture descriptions that are developed during the case studies. This represents a timeframe that is outside the scope of this study, and poses significant risk to the participating organisations.

Furthermore, Peffers et al. (2012) describe an illustrative scenario as an "application of an artefact to a synthetic or real-world situation aimed at illustrating suitability or utility of the artefact". While an illustrative scenario also evaluates an artefact in a naturalistic setting, the method may be executed in a shorter timeframe as the purpose is to illustrate the utility of the artefact. An illustrative scenario also limits the risk to participating organisations, since they are not obliged to instantiate the particular architectures. Given these considerations, *illustrative scenarios* are selected as the preferred method for evaluating the reference architecture for IKNs.

In order for the utility of the reference architecture for IKNs to be demonstrated and evaluated, the illustrative scenario method is designed to closely resemble the scenarios that were considered as part of the design of the artefact. The cooperation of a number of operational IKNs was therefore enlisted. The eligibility of these networks as potential illustrative scenarios was determined through various definitive criteria, including:

- A network of organisations that collaborate within the knowledge supply chain,
- A focus on fostering sustainable innovation that promotes the competitiveness of network stakeholders,
- Willingness to provide access to architectural information.
- Willingness to commit time from employees with a strategic and business focus to the research.

For each of the selected scenario networks, the reference architecture for IKNs was implemented in as close an approximation of its expected usage environment as possible. Workshops aimed at describing the particular architectures of the scenario networks were therefore conducted with representatives from the organisations that participate in each scenario network. The artefact was utilised as a reference architecture in these workshops. The researcher acted as a participant observer in the workshops, playing the role of an architect with knowledge of the reference architecture that is to be employed. The workshops took the form of structured interviews that were conducted with the entire group of representatives of the scenario network. The workshops were therefore structured according to a predefined set of interview questions, helping to maintain the objectivity of the process. Due to constraints on time and knowledge of the reference architecture for IKNs from the participants, the researcher further played the role of architect by interpreting and modelling the data collected from the network representatives. The accuracy and objectivity of these interpretations and models were confirmed through a round of member checking.

3.6 Limitations

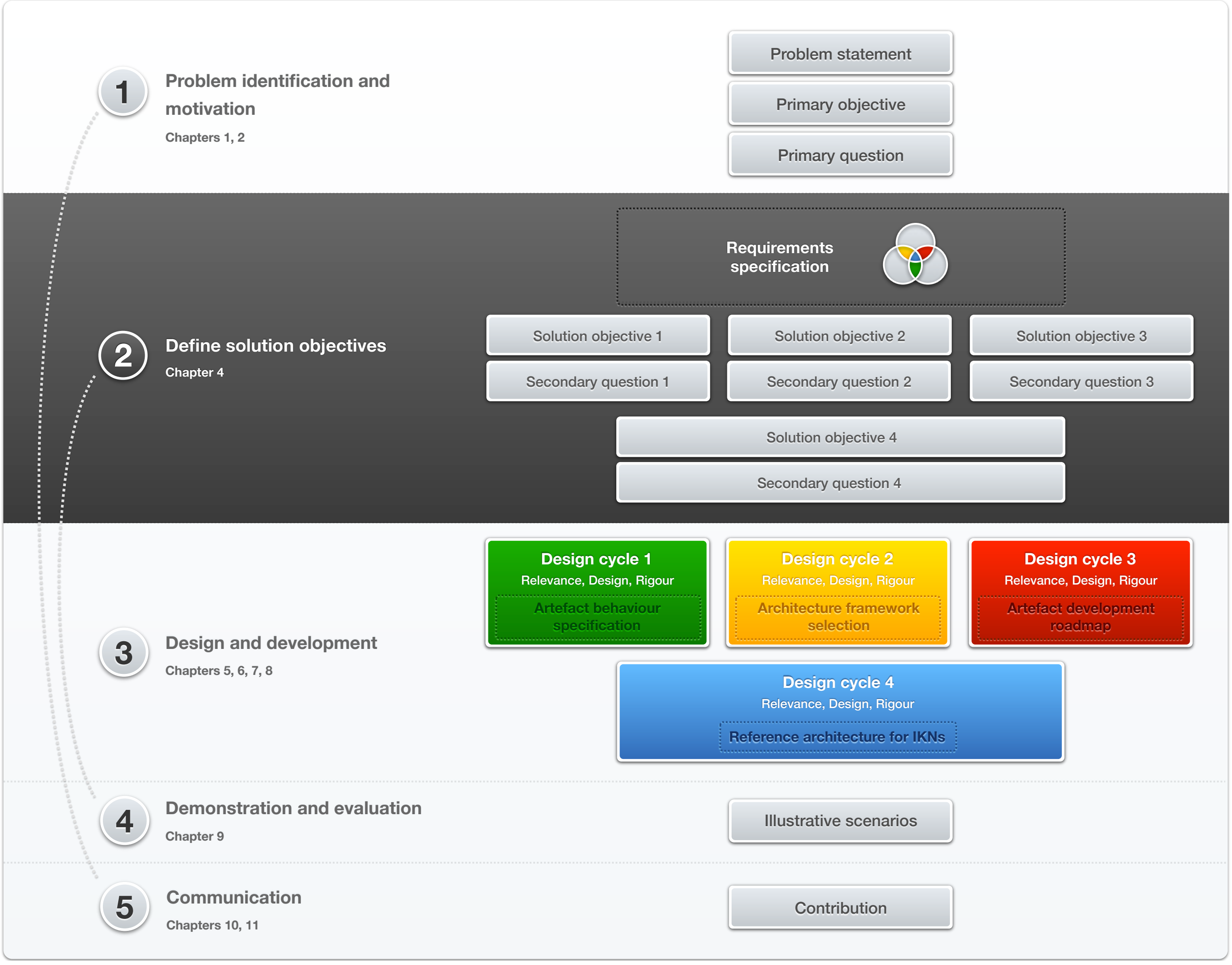
The limitations of the research methodology include the limited evaluation of the artefact because of the timeframe and scope that is required to do a full evaluation. As stated in section 3.3.1.1, it is often the case that DSR describes a body of work that will in most cases surpass the scope of a doctoral study (Hevner 2012). The evaluation which will be presented in chapter 9 is equivalent to a laboratory test through illustrative scenarios at the end of the Design cycle in the three-cycle view of DSR (refer to Figure 3.2). While this is regarded as sufficient for the scope of a doctoral study (Hevner 2012), it is not as comprehensive as feedback resulting from extensive case studies or field tests, as would be the case in the last phase of the Relevance cycle. The kind of field test in which the reference architecture for IKNs is fully implemented in multiple case studies and its effect on the participating IKNs measured, however, will take an extensive amount of time to conduct and does not fit in the timeframe afforded to this doctoral study. Furthermore, it was not possible to conduct the illustrative scenarios without extensive involvement of the researcher as a practitioner. While the results of the evaluation show that the artefact satisfies all of its requirements and is therefore regarded as a solution to the problem, the artefact's performance is linked to the researcher's ability to implement and interpret it.

3.7 Conclusion

A constructivist research methodology adopted for the production of a reference architecture for IKNs is presented in this chapter. The research design is based on the DSRM presented by Peffers et al. (2004). The aim of the research design is to produce a reference architecture for IKNs, as that would solve the identified problem of a lack of a suitable reference architecture for the engineering of IKNs. As mentioned earlier, complete motivation for the content of the design cycles in the research design is provided as the plan is progressively executed, as that corresponds to the decision-based development process of the artefact based on learning through building. Chapter 4 continues on to activity 2 of the research design by rationally defining objectives for a solution to the problem and planning the design and development of the artefact (activity 3) based on these objectives.

Part 2 - Define solution objectives

This second part of the document contains a single chapter (chapter 4) and presents the second activity of the research design employed in this study, namely the definition of objectives for a solution to the problem articulated in Part 1. These objectives are defined through a structured approach to requirement specification and are accompanied by secondary research questions that direct the design and development of an artefact that aims to achieve these objectives in Part 3. These solution objectives furthermore form the basis of the criteria for the demonstration and evaluation of the artefact in Part 4.



4. Solution objectives

4.1 Introduction

This chapter presents the second activity in the research design employed in this study and aims to define objectives for a solution to the problem articulated in Part 1. In order to infer these solution objectives rationally, a structured approach to requirements specification is employed that considers functional requirements, technical requirements and implementation scenarios for a solution, and combines these to form compound requirement perspectives. These compound requirement perspectives are then translated into solution objectives with accompanying secondary research questions that direct the search process toward achieving the objectives.

In keeping with the pragmatic paradigm in which the study is conducted, solution objectives from this chapter also serve as secondary research objectives for the study. This is underlined by the DSR method that embeds elements of its research contribution in the design of the artefact.

Background information on solution objectives and requirement specification is provided in section 4.2, and the requirement specification frame of reference for this study is introduced in section 4.3. An integrated approach to compound requirements specification develops solution objectives and secondary research questions in section 4.4. These solution objectives and secondary research questions are aligned with design cycles in section 4.5, before section 4.6 concludes this chapter.

4.2 Background

This section provides an overview of the concepts of requirements and solution objectives. A structured and comprehensive approach to requirements specification is also introduced.

4.2.1 Philosophical and semantic perspectives

Section 3.5.2 identified pragmatism as the research paradigm for this study. From an ontological point of view pragmatism's stance toward reality relies on action and change (Goldkuhl 2012), i.e. that reality is constituted by actions and the changes that occur as a result of these actions. Within this paradigm, DSR changes the state-of-the-world through the introduction of novel artefacts and hence design researchers are comfortable with alternative world-states (Vaishnavi & Kuechler 2004).

Given these alternative world-states, a *problem* can be formally defined as the difference between a goal state and the current state of a system (Hevner et al. 2004) and a *solution* is therefore typically an artefact or approach that moves the system from the current state to the goal state. A *requirement* is defined as something "that is needed or wanted" (Oxford Dictionaries n.d.) and therefore, within this scheme, requirements are conditions that are imposed on the solution artefact and serve as a frame of reference for judging design alternatives (Miedema et al. 2007). The *specification* of these requirements is the registration of constraints, demands and wishes that are established to state the design problem and its envisaged solution (Miedema et al. 2007). Requirements therefore express the solution's minimum required utility in order to solve the problem and move the system from the current state to the goal state. An *objective* is defined as something that is "aimed at or sought" (Oxford Dictionaries n.d.), and solution objectives are therefore defined as goals that the solution artefact is designed to achieve.

An acceptable artefact's objectives can exceed the requirements for a solution. The design researcher as pragmatist, however, holds the axiological view that value is solely determined by usefulness (Goldkuhl 2011). This means that an accurate alignment of requirements and solution objectives is a desired situation and would represent an accurate solution to the problem by migrating the system from the current state to the goal state. The DSRM shares this view by emphasising that solution objectives have to be inferred rationally from the problem (Peffers et al. 2008). Any solution objectives that are included in the design of the artefact, but do not directly align with the requirement specifications, have to be justified with another pragmatic argument that shows how they contribute to the required or desired utility of the artefact.

In this study significant emphasis is therefore placed on the comprehensive articulation of requirements for a solution to the research problem, and the alignment of these requirement specifications with solution objectives to be designed for, as discussed in this chapter.

4.2.2 Requirements specification

In Hevner's three-cycle view of DSR (Hevner (2007), refer to section 3.3.1.1) the Relevance cycle bridges the contextual and design environments and initiates the DSR method by identifying the problem and opportunities in the application environment. The Relevance cycle also delivers the requirements for the artefact from the application domain to the Design cycle and these factor into the design process, while both constructing and evaluating the artefact (Hevner & Chatterjee 2010b).

Requirement specification is therefore a critical element of DSR, and of particular importance is the way in which requirement specifications can be used to qualitatively consider an artefact's desired design characteristics. Modern artefact development methods consider requirement specifications as important sources of information for design decisions, both prior to as well as during the artefact development cycle, and are therefore no longer confined to merely being a point of departure for the development cycle (Miedema et al. 2007).

Technical requirements are the traditional way of specifying artefact parameters prior to development and are unequivocal expressions of requirements (Miedema et al. 2007). These technical requirement specifications relate to how an artefact achieves its objectives. However, they may fix constraints on the artefact too early and are not a good way to address the unquantifiable requirements of the artefact. The technical requirements of an artefact, for example a pair of earphones, could include their weight, size, impedance and decibel levels.

Miedema et al. (2007) discuss functional requirements and scenarios as ways to balance the strict nature of technical requirements in order to provide balanced information for design decisions to be based on. Functional requirement specifications are a representation of the desired behaviour of the artefact (Miedema et al. 2007), i.e. what the artefact does. Extending the example of earphones, their functional requirements, on the one hand, may be to provide an immersive listening experience with robust delivery of the bass tones in modern dance music. Scenarios, on the other hand, emphasise an artefact’s environment and its interaction with it and can be seen as a way of describing what will happen to an artefact during its life cycle (Miedema et al. 2007). For the earphones, usage scenarios may include their use by joggers or DJs in clubs. It is therefore clear that decisions based on these scenarios will greatly impact the design. However, it is imperative to ensure synergy and coherence between these different types of requirement specifications, as they combine to form a requirement frame of reference.

There is a growing trend in artefact development to no longer pre-specify the entire development cycle, as design decisions will inevitably cause changes in the development process (Miedema et al. 2007). These design decisions are based on the most current information on the artefact and its requirement specifications. As these requirement specifications evolve along with the making of design decisions, they are a living constituent of the development cycle. Thus, while the requirement frame of reference provides the most up-to-date information on the artefact, it also evolves. The purpose of the development cycle, however, remains static and ensures that the artefact which is delivered does indeed still address the initial problem statement. As the technical requirements, functional requirements and scenarios for an artefact evolve throughout the development cycle, so will the relationships between these specifications (Miedema et al. 2007). These relationships can be expressed in the form of compound perspectives or viewpoints that combine the separate requirement specification types, as illustrated in Figure 4.1.

There are both divergent and convergent phases in the combination of requirement specifications to form compound perspectives in the development cycle (Miedema et al. 2007). The divergent phase comprises the perspectives indicated in green, yellow and red in Figure 4.1 and considers a single, compound requirement perspective at a time. This allows for the development of divergent, and not necessarily coherent, ideas about the artefact that is to be developed.

The green perspective is an intersection of the scenarios and the functional requirement specification. It considers the artefact’s desired behaviour in the context of its environment throughout its life cycle. Reflecting on the earlier example, for a jogger an immersive listening experience with rich bass tones must be balanced with being able to hear traffic and handle sweat. A DJ, in turn, might be more concerned with

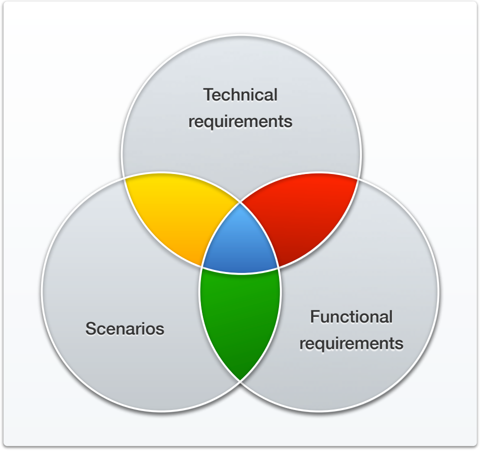


Figure 4.1: Perspectives on requirements specification

the way the earphones allow him to block out sounds from the club and only wants to hear his source material.

The yellow perspective combines scenarios and the technical requirement specification, i.e. how the artefact works in the context of its environment. The jogger will require a decibel-level specification which is quite different to that of the DJ.

The red perspective is an intersection of the technical and functional requirement specification and considers how the artefact’s quantifiable parameters allow it to exhibit the desired behaviour. The decibel-level specification should be of such a nature that it is loud enough without being so loud that the mid-range frequencies drown out the bass tones.

The convergent phase (indicated in blue in Figure 4.1) combines all three these perspectives to form a coherent model of the artefact’s requirements considered from a technical, functional and scenario point of view (Miedema et al. 2007). The earphone design should therefore ensure an immersive listening experience with rich bass tones for both joggers and DJs through its use of materials, weight, size, decibel level and impedance. As this might result in a substandard design for both scenarios, it could be that the requirements specification process leads to the development of two different designs.

By sequentially considering each of these perspectives (green, yellow, red and blue) it is possible to gradually reduce design uncertainty throughout the development life cycle (Lutters et al. 2004). Each consideration of a perspective updates its constituent requirements specifications and therefore updates the requirements frame of reference for making design decisions (Miedema et al. 2007). This is akin to the DSR mantra of ‘learning through building’ as discussed in section 3.3.1.

Once an implementation of the artefact is produced through DSR, it is evaluated to determine the degree to which it has achieved its objectives and hence satisfied its requirements (Hevner et al. 2004). Requirements specifications should therefore not only be combined with DSR at the start of the Design cycle, but also at the end when the artefact is tested prior to field implementation.

The development of requirement specifications throughout the development cycle can therefore be used in conjunction with DSR to form a powerful combination for the development of artefacts where a pragmatic and constructivist approach is required.

4.3 Requirement specification frame of reference

This section develops the requirement specification frame of reference for a solution to the research problem addressed in this study (refer to section 1.3):

No reference architecture exists for use in the engineering of IKNs.

The requirement specification frame of reference is the basis for considering compound requirement perspectives on a possible solution to this problem and consists of three types of specifications, namely functional requirements, technical requirements and scenarios (refer to Figure 4.2). This frame of reference provides the most up-to-date information on requirements for the solution artefact to inform design decisions, and is also updated as the design and development process progresses.

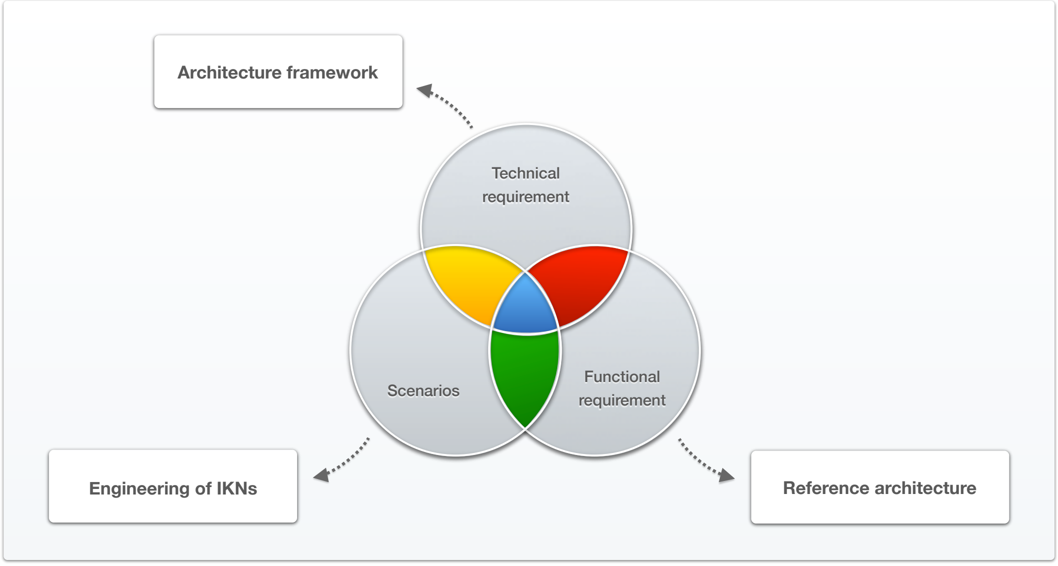


Figure 4.2: Requirement specification frame of reference

The requirement specifications articulated in sections 4.3.1, 4.3.2 and 4.3.3 below are the constituents of the requirement specification frame of reference for this study and is illustrated in Figure 4.2.

4.3.1 Functional requirements

Functional requirements articulate the desired behaviour of the artefact as a solution to the problem. For this study the functional requirement for the solution artefact is as follows:

- The artefact must function as a reference architecture.

This requirement statement implies that the artefact must direct the design of architectures for a given domain by defining a unified terminology, describing the functionality and roles of components, providing template components and giving example architectures (Camarinha-Matos & Afsarmanesh 2008a). The artefact must therefore be the basis for designing the implementation specific architectures for instances of enterprises in the class of systems for which the artefact is valid.

The description of the problem statement in section 1.3 clarifies why this requirement statement describes the desired behaviour of the artefact as a solution to the problem. The need for an artefact that behaves in a specific way, i.e. as a reference architecture, has been illustrated from both an academic and a practical perspective. Adhering to this requirement therefore also ensures the minimum required functional utility to solve the problem from an academic and a practical perspective.

4.3.2 Technical requirements

Technical requirements relate to unequivocal characteristics that enable the solution artefact to achieve its objectives, i.e. how it technically goes about solving the problem. For this study the technical requirement for the solution artefact is as follows:

- The artefact must employ an architecture framework.

This requirement implies that the solution artefact’s content must be structured, classified and organised via some architecture framework to ensure comprehensiveness and coherence. This architecture framework must be free of domain context and must describe the structure of architectural content at a generic level, thereby ensuring the integrity of the content.

A core idea behind the development of architectures is that the descriptions consist of models (refer to section 2.7). This is true for both implementation and reference architecture descriptions. When creating architecture descriptions, a major challenge is to ensure that the models that are created to comprise the description represents a comprehensive view on the system architecture (Schekkerman 2004; Greefhorst et al. 2006; Muller & Hole 2006). An associated challenge is to ensure that the level of detail of these artefacts is appropriate to make them useful, while ensuring coherence amongst them (Muller & Hole 2006).

Given these challenges, the required characteristic of the solution artefact is that it must employ an architecture framework that ensures that the models it contains are comprehensive and coherent. This comprehensiveness and coherence must also be characteristics of the implementation-specific architecture descriptions that it produces. An architecture framework furthermore provides a generic solution space and a common vocabulary within which a specific problem can be solved (Schekkerman 2004). Adhering to this requirement, i.e. employing an architecture framework, ensures that the artefact is technically equipped to achieve its objectives.

4.3.3 Scenarios

In requirements specification, scenarios emphasise an artefact’s environment and its interaction with it. For this study the scenario specification for the solution artefact is as follows:

- The artefact must address the engineering of IKNs.

This requirement implies that the domain of IKNs forms the artefact’s implementation environment and that its interaction with this domain should be designed for. The artefact should therefore not relate to CNOs in general, but to any organisation that can be deemed to be an instantiation of an IKN (as defined in section 2.2) as a specific limited subset of this larger range of organisations.

The scenario further implies that it is not simply the existence of IKNs that form the artefact’s environment, but that it is to be used specifically in the engineering of these organisations. In this context, the term ‘engineering’ refers to its usage in the context of the field of enterprise engineering, as discussed in section 2.6. The reference architecture therefore needs to help transform IKNs from organically grown entities with a high probability of failure, low adaptability to change and limited opportunities for optimisation, to purposefully designed and engineered social systems. This, however, entails making the high complexity of these networks intellectually manageable, and to achieve this a separation of the intention (construction) and content (function) of IKNs is required.

Engineering of an IKN is performed in various contexts through the life cycle of these networks and the artefact therefore needs to contribute to multiple actions, including the design, redesign, deployment and subsequent transformation of IKNs. In all of these cases, engineering the network entails moving the enterprise with a roadmap from a current state (as-is) to a future state (to-be).

Due to the collaborative nature of an IKN, the engineering of such an organisation is performed by an inter-organisational team. This team is tasked with the governance of the network and may include external consultants. It represents the users of the reference architecture for IKNs and forms part of the scenario in which the artefact is required to operate. Adhering to this requirement ensures that the artefact’s environment, and its interaction with its environment, is also considered during its design. This greatly increases the likelihood of it being a satisfactory solution to the identified problem.

4.4 Compound requirement perspectives, solution objectives and secondary research questions

This section exploits the requirements specification frame of reference that was constituted in the previous section to rationally infer various solution objectives for the artefact.

The requirement specifications articulated in section 4.3 can be combined to create viewpoints for considering compound requirement perspectives. This section introduces the compound requirement perspectives that will be considered in this study. Solution objectives for the artefact and accompanying secondary research questions are consequently developed based on each of these perspectives.

The translation of the compound requirement statements to solution objectives reveals a simple alignment between the two sets – solution objectives are expressed in such a way that achieving them will ensure that the requirements are met. The associated secondary research questions are in turn designed to guide the search and design process to a point where the artefact has achieved the solution objectives.

4.4.1 Functional requirement vs. Scenarios

This first compound requirement perspective considers the intersection of functional requirements and scenarios. For the artefact designed in this study, these requirements are as follows (refer to Figure 4.3):

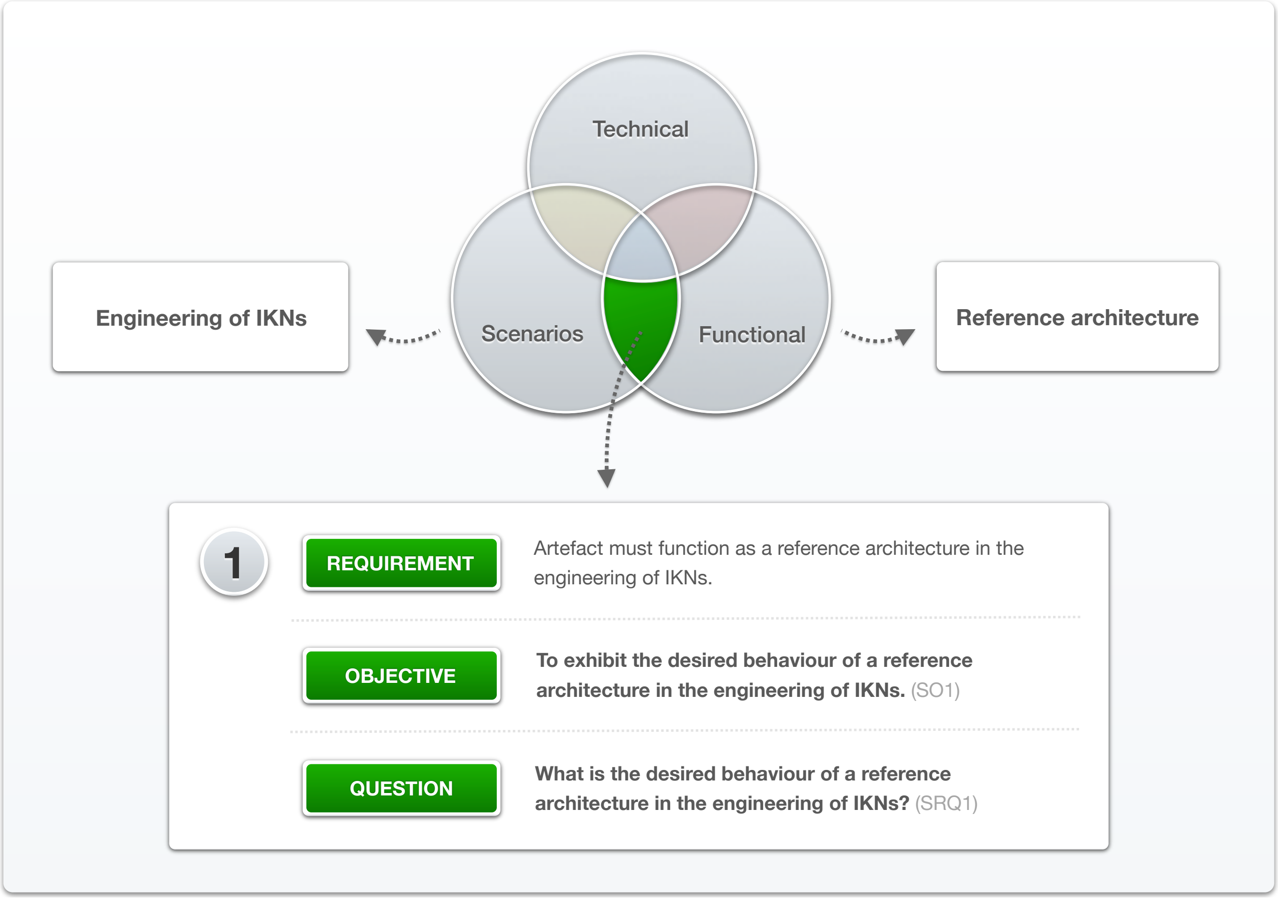
Functional requirement: The artefact must function as a reference architecture.

Scenarios: The artefact must address the engineering of IKNs.

This perspective states that the artefact will be required to exhibit its desired behaviour (as described by the functional requirement) when used in a specified environment (as described by the scenario). The scenario therefore influences the understanding of the artefact's desired behaviour. Combining the functional requirement and scenario yields the following compound requirement perspective:

Compound requirement: The artefact must function as a reference architecture in the engineering of IKNs.

The implications of this compound statement can be expounded from both sides of the perspective. By stating that the artefact must function as a *reference architecture* to be used in the engineering of IKNs, it is clear that the artefact is not designed to function as a *methodology* for engineering IKNs, as that would be exhibiting an incorrect behaviour in the correct scenario. In the same sense the artefact must function as a reference architecture to be used in the *engineering of IKNs* and not in the *functioning of IKNs* or even the *engineering of CNOs*, as that would be the correct behaviour in an incorrect scenario. The statement



Artefact must function as a reference architecture in the engineering of IKNs.

To exhibit the desired behaviour of a reference architecture in the engineering of IKNs. (SO1)

What is the desired behaviour of a reference architecture in the engineering of IKNs? (SRQ1)

Figure 4.3: Compound requirement perspective - Functional requirement vs. scenarios

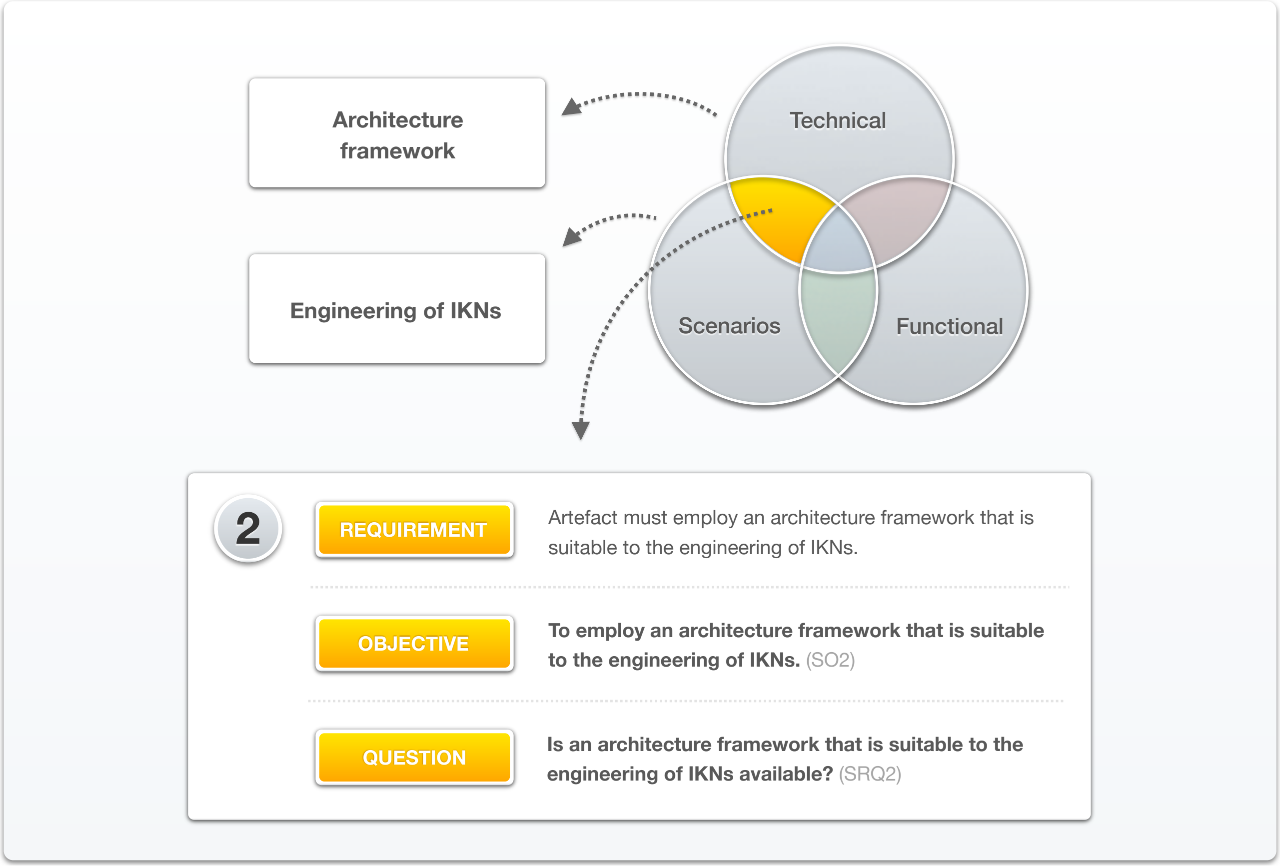
therefore expresses a need for a specific behaviour in a specific environment, and does not focus on the technical requirement.

This compound requirement can be translated into the following solution objective with an accompanying secondary research question:

Solution objective:	To exhibit the desired behaviour of a reference architecture in the engineering of IKNs. (SO1)
Secondary research question:	What is the desired behaviour of a reference architecture in the engineering of IKNs? (SRQ1)

4.4.2 Technical requirement vs. Scenarios

The second compound requirement perspective considers the intersection of technical requirements and scenarios. For the artefact designed in this study, these requirements are as follows (refer to Figure 4.4):



Artefact must employ an architecture framework that is suitable to the engineering of IKNs.

To employ an architecture framework that is suitable to the engineering of IKNs. (SO2)

Is an architecture framework that is suitable to the engineering of IKNs available? (SRQ2)

Figure 4.4: Compound requirement perspective - Technical requirement vs. Scenarios

Technical requirement: The artefact must employ an architecture framework.

Scenario: The artefact must address the engineering of IKNs.

This perspective considers that the artefact will be required to exhibit certain characteristics (as described by the technical requirement) in a specified environment, as described by the scenario. This scenario, however, has now been augmented through the development of the artefact behaviour specification in the green perspective. The updated scenario therefore influences the understanding of the behaviour that the artefact is expected to exhibit. Combining the technical requirement and updated scenario yields the following compound requirement perspective:

Compound requirement: The artefact must employ an architecture framework that is suitable to the engineering of IKNs.

The implications of this compound statement can be viewed from both sides of the perspective. By stating that the artefact must employ an *architecture framework* that is suitable to the engineering of IKNs, the design is constrained to include a specific form of inherent organisation and structure. Likewise, by stating the artefact must employ an architecture framework that is *suitable to the engineering of IKNs*, the artefact is constrained to implement an architecture framework that suits the focus on the construction of IKNs. The statement therefore expresses the need for a certain characteristic in a certain environment without focusing on the functional requirement, although the scenario's content has been augmented to include knowledge of the artefact behaviour specification.

This compound requirement can be translated into a solution objective with an accompanying secondary research question:

Solution objective: To employ an architecture framework that is suitable to the engineering of IKNs. (SO2)

Secondary research question: Is an architecture framework that is suitable to the engineering of IKNs available? (SRQ2)

4.4.3 Functional requirement vs. Technical requirement

The third compound requirement perspective considers the intersection of functional requirements and technical requirements. For the artefact designed in this study, these requirements are as follows (refer to Figure 4.5):

Functional requirement: The artefact must function as a reference architecture.

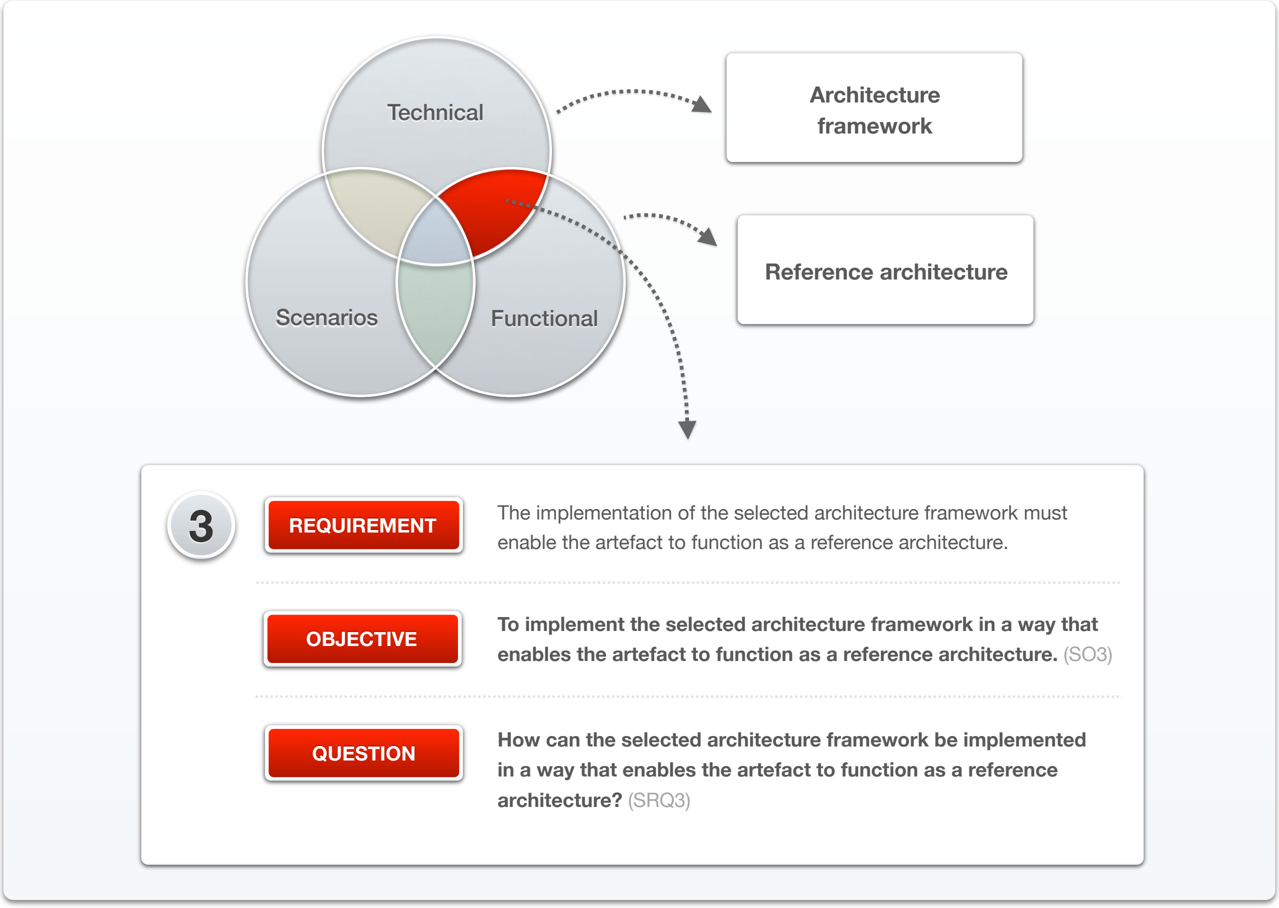
Technical requirement: The artefact must implement the selected architecture framework.

This perspective considers that the artefact will be required to embody certain characteristics (as described by the augmented technical requirement which now features an architecture framework selection) in order to exhibit certain behaviour (as described by the augmented functional requirement that includes an artefact behaviour specification). The technical requirement therefore influences the understanding of how the design achieves the desired behaviour.

Combining these two requirements, the following compound requirement perspective can be identified:

Compound requirement: The implementation of the selected architecture framework must enable the artefact to comply with the artefact behaviour specification.

The implications of this compound statement can be explained from both sides of the perspective. By stating that the *implementation of the selected architecture framework* must enable the artefact to comply with the artefact behaviour specification, the artefact is constrained to produce its behaviour in a certain way. Likewise, by stating that the implementation of the selected architecture framework must enable the artefact to *comply with the artefact behaviour specification*, the artefact's behaviour influences the



REQUIREMENT The implementation of the selected architecture framework must enable the artefact to function as a reference architecture.

OBJECTIVE **To implement the selected architecture framework in a way that enables the artefact to function as a reference architecture.** (SO3)

QUESTION **How can the selected architecture framework be implemented in a way that enables the artefact to function as a reference architecture?** (SRQ3)

Figure 4.5: Compound requirement perspective - Functional requirement vs. Technical requirement

approach to the implementation of the selected architecture framework. The statement therefore constitutes a causal relationship between the artefact characteristics and the artefact behaviour. The perspective's requirement statement does not explicitly include the scenario although the functional and technical requirements have been updated by knowledge of the artefact behaviour specification and architecture framework selection, respectively.

This compound requirement can be translated into a solution objective with an accompanying secondary research question:

Solution objective:	To implement the selected architecture framework in order to enable the artefact to comply with the artefact behaviour specification. (SO3)
Secondary research question:	How can the selected architecture framework be implemented to enable the artefact to comply with the artefact behaviour specification? (SRQ3)

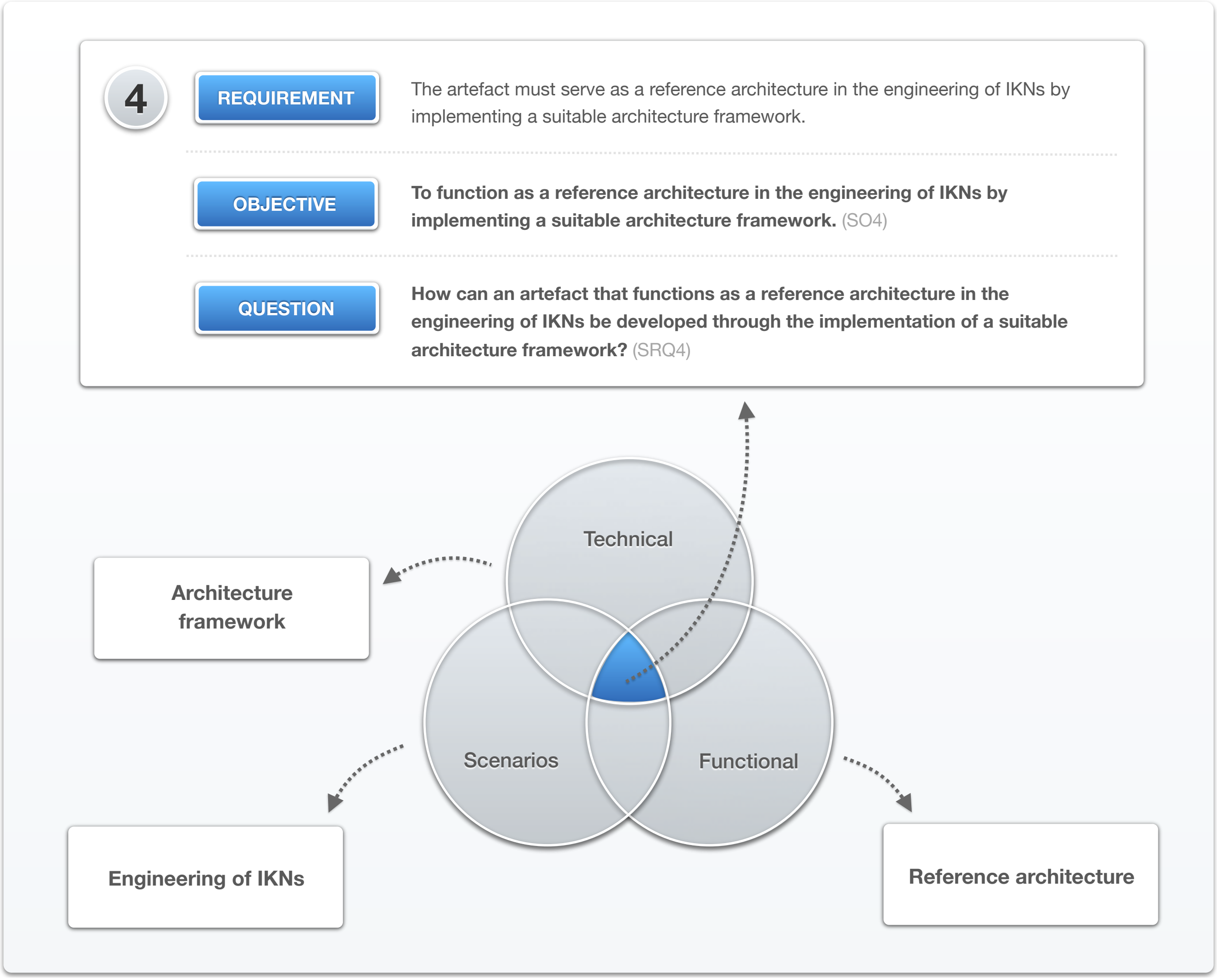
4.4.4 Functional requirement vs. Technical requirement vs. Scenarios

The fourth and final compound requirement perspective considers the intersection of all three of the requirement specification types, namely functional requirements, technical requirements and scenarios. For the artefact designed in this study, these requirements are as follows (refer to Figure 4.6):

Functional requirement: The artefact must function as a reference architecture.

Technical requirement: The artefact must employ an architecture framework.

Scenarios: Engineering of IKNs.



4 **REQUIREMENT** The artefact must serve as a reference architecture in the engineering of IKNs by implementing a suitable architecture framework.

OBJECTIVE **To function as a reference architecture in the engineering of IKNs by implementing a suitable architecture framework.** (SO4)

QUESTION **How can an artefact that functions as a reference architecture in the engineering of IKNs be developed through the implementation of a suitable architecture framework?** (SRQ4)

Figure 4.6: Compound requirement perspective - Functional requirement vs. Technical requirement vs. Scenarios

By combining these three requirements, the following compound requirement perspective can be identified:

Compound requirement: The artefact must function as a reference architecture in the engineering of IKNs by implementing a suitable architecture framework.

This compound requirement provides a holistic view on the requirement frame of reference by simultaneously considering all three types of requirement specifications, and can be translated into the following solution objective with an accompanying secondary research question:

Solution objective: To function as a reference architecture in the engineering of IKNs by implementing a suitable architecture framework. (SO4)

Secondary research question: How can an artefact that functions as a reference architecture in the engineering of IKNs be developed through the implementation of a suitable architecture framework? (SRQ4)

This fourth solution objective and its accompanying secondary research question relates to the primary research objective (RO) and accompanying primary research question (PRQ) by offering a holistic view on the proposed solution to the research problem.

4.4.5 Overview of solution objectives and research questions

Table 4.1 provides an overview of solution objectives with accompanying research questions that were defined through the consideration of compound requirement perspectives.

Table 4.1: Solution objectives and secondary research questions

Code	Solution objective	Code	Secondary research question
SO1	To exhibit the desired behaviour of a reference architecture in the engineering of IKNs.	SRQ1	What is the desired behaviour of a reference in the engineering of IKNs?
SO2	To employ an architecture framework that is suitable to the engineering of IKNs.	SRQ2	Is an architecture framework that is suitable to the engineering of IKNs available?
SO3	To implement the selected architecture framework in a way that enables the artefact to function as a reference architecture.	SRQ3	How can the selected architecture framework be implemented in a way that enables the artefact to function as a reference architecture?
SO4	To function as a reference architecture in the engineering of IKNs through the implementation of a suitable architecture framework.	SRQ4	How can an artefact that functions as a reference architecture in the engineering of IKNs be developed through the implementation of a suitable architecture framework?

4.5 Alignment of solution objectives with design cycles

The research design employed in this study includes four DSR design cycles, which incrementally develop a reference architecture for IKNs (refer to section 3.5.3). This study has a pragmatic approach to DSR as a research method and emphasis is placed on the drivers of each design cycle, thereby ensuring their relevance and the value of their contribution. Each of these design cycles therefore features a specific solution objective with an accompanying secondary research question as drivers.

The first, second and third cycles represent the divergent phase of the requirement specification process and each considers a specific perspective on the requirement frame of reference of the artefact. The first cycle (green) is driven by the intersection of the functional requirement specification and scenarios and considers the artefact’s desired behaviour in the context of its environment throughout its life cycle. The second cycle (yellow) combines the technical requirement specification and scenarios, i.e. how the artefact’s environment influences its construction. The third cycle (red) deals with the intersection of the functional and technical requirements specifications and considers how the artefact’s construction enables it to exhibit the desired behaviour. The fourth (blue) cycle represents the convergent phase of the requirement specification process. It combines the perspectives and builds on the previous design decisions that were made in the green, yellow and red cycles to form a coherent model of the final artefact as it is to be constructed.

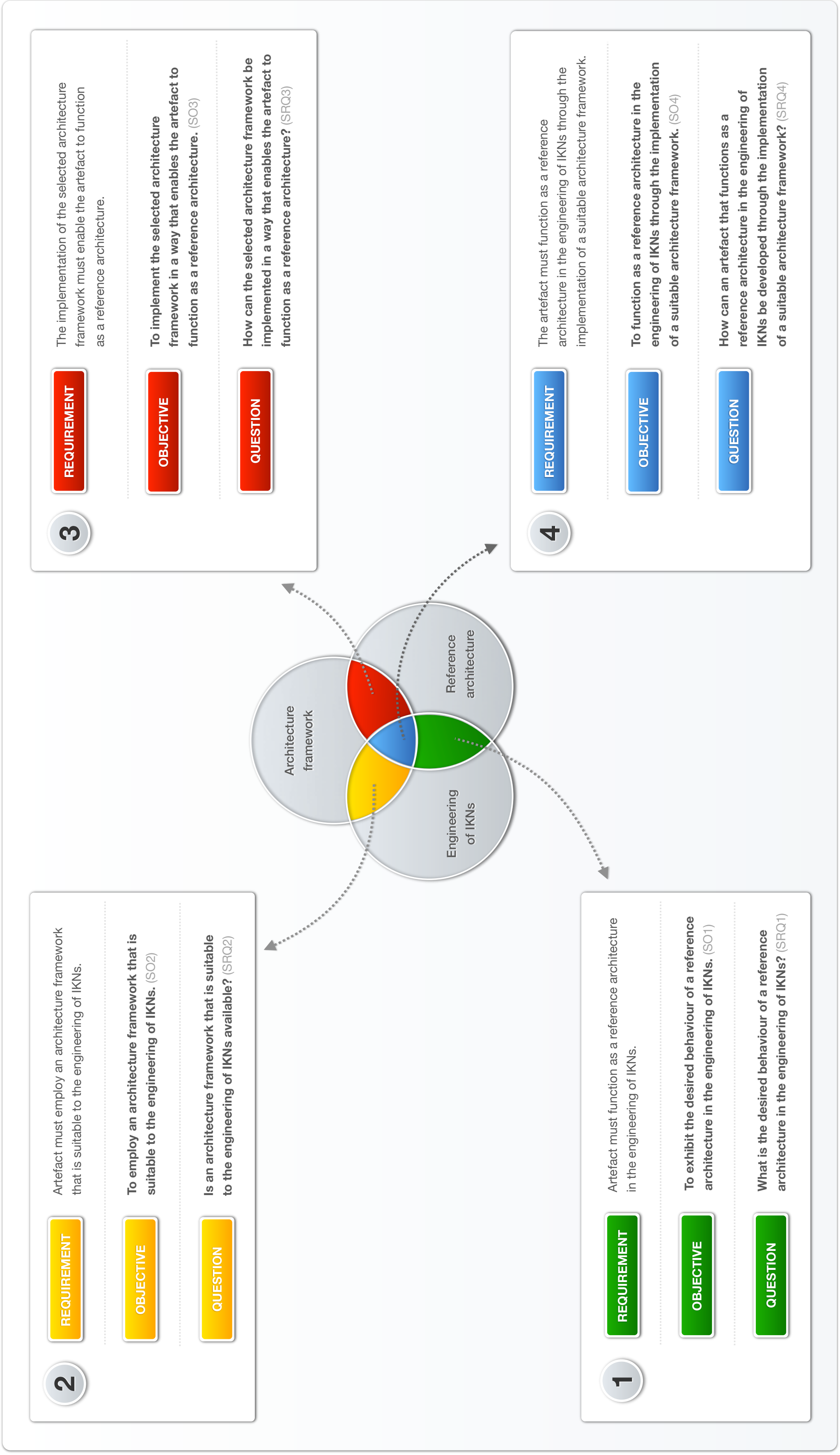
Each of the four design cycles includes elements from the three-cycle view of DSR, namely Relevance, Design and Rigour. They all contribute to incrementally develop a reference architecture for IKNs, with the first, second and third cycles delivering versions of the artefact that show improvements in the understanding of the final version’s requirements and design. Throughout the four cycles of the research design, design decisions are based on the most up-to-date information on the artefact and its requirement specifications. These decisions, however, also update the requirement specification frame of reference as new knowledge about the artefact is acquired, e.g. detailed determination of the desired behaviour of the artefact, the architecture framework and the approach to implementing it. This progression is indicative of the pragmatic philosophical underpinning of this study, as knowledge generated is immediately valuable to induce action and change.

All four cycles follow the structure of Awareness, Suggestion and Development. The inner mechanics of the four design cycles will be discussed in greater detail in their relevant chapters (refer to chapters 5, 6, 7 and 8), with a presentation of the overall logic and reasoning presented in this section.

4.5.1 First design cycle

The cycle is driven by SO1 and SRQ1 as derived by considering the requirements intersection of scenarios and functional requirements (refer to the green elements in Figure 4.7). This cycle is discussed in chapter 5, and follows the structure of Awareness, Suggestion and Development, before circumscription leads to the initiation of a second design cycle.

Requirement:	The artefact must function as a reference architecture in the engineering of IKNs.
Solution objective:	To exhibit the desired behaviour of a reference architecture in the engineering of IKNs. (SO1)
Secondary research question:	What is the desired behaviour of a reference architecture in the engineering of IKNs? (SRQ1)
Awareness:	The behaviour of reference architecture for IKNs should address the functional requirements of the network engineering process for such an artefact.
Suggestion:	Develop an artefact behaviour specification that registers the functional requirements for a reference architecture in the engineering of IKNs.
Development:	1. Constitute the network engineering process as context, the modelling target and the modelling framework. 2. Investigate and specify the desired behaviour of the artefact in terms of the i. meta-relationships, ii. IKN life cycle and iii. artefact users.
Artefact version:	Artefact behaviour specification



1

REQUIREMENT: Artefact must function as a reference architecture in the engineering of IKNs.

OBJECTIVE: **To exhibit the desired behaviour of a reference architecture in the engineering of IKNs.** (SO1)

QUESTION: **What is the desired behaviour of a reference architecture in the engineering of IKNs?** (SRQ1)

2

REQUIREMENT: Artefact must employ an architecture framework that is suitable to the engineering of IKNs.

OBJECTIVE: **To employ an architecture framework that is suitable to the engineering of IKNs.** (SO2)

QUESTION: **Is an architecture framework that is suitable to the engineering of IKNs available?** (SRQ2)

3

REQUIREMENT: The implementation of the selected architecture framework must enable the artefact to function as a reference architecture.

OBJECTIVE: **To implement the selected architecture framework in a way that enables the artefact to function as a reference architecture.** (SO3)

QUESTION: **How can the selected architecture framework be implemented in a way that enables the artefact to function as a reference architecture?** (SRQ3)

4

REQUIREMENT: The artefact must function as a reference architecture in the engineering of IKNs through the implementation of a suitable architecture framework.

OBJECTIVE: **To function as a reference architecture in the engineering of IKNs through the implementation of a suitable architecture framework.** (SO4)

QUESTION: **How can an artefact that functions as a reference architecture in the engineering of IKNs be developed through the implementation of a suitable architecture framework?** (SRQ4)

Figure 4.7: Requirements specification, secondary research questions and objectives

4.5.2 Second design cycle

The cycle incrementally improves the artefact produced in the first cycle and is driven by SO2 and SRQ2 as derived by considering the requirements intersection of scenarios and technical requirements (refer to yellow elements in Figure 4.7). This cycle is discussed in chapter 6 and follows the structure of Awareness, Suggestion and Development, before circumscription leads to initiation of a third design cycle.

Requirement:	The artefact must employ an architecture framework that is suitable to the engineering of IKNs
Solution objective:	To employ an architecture framework that is suitable to engineering of IKNs. (SO2)
Secondary research question:	Is an architecture framework that is suitable to the engineering of IKNs available? (SRQ2)
Awareness:	Not all architecture frameworks are suitable to the engineering of IKNs. The requirements for suitability, however, can be expressed in terms of architecture framework dimensions.
Suggestion:	Investigate and evaluate the Zachman framework for suitability to the engineering of IKNs, as previously described in terms of architecture framework dimensions.
Development:	1. Investigate the Zachman framework in terms of the architecture framework dimensions used to describe suitability to engineering of IKNs. 2. Compare the Zachman framework's fit with the requirements for each of the dimensions. 3. Select the Zachman framework for implementation, or iterate the investigation with a different candidate framework.
Artefact version:	Architecture framework selection

4.5.3 Third design cycle

The cycle incrementally improves the artefact produced in the second cycle and is initiated by SO3 and SRQ3 as derived by considering the requirements intersection of technical requirements and functional requirements (refer to red elements in Figure 4.7). This cycle is discussed in chapter 7 and follows the structure of Awareness, Suggestion and Development, before circumscription leads to the initiation of a fourth and final design cycle.

Requirement:	The implementation of the selected architecture framework must enable the artefact to function as a reference architecture.
Solution objective:	To implement the selected architecture framework in a way that enables the artefact to function as a reference architecture. (SO3)
Secondary research question:	How can the selected architecture framework be implemented in a way that enables the artefact to function as a reference architecture? (SRQ3)
Awareness:	The implementation of the selected architecture framework must enable the artefact to function as a reference architecture.
Suggestion:	Compile an artefact development roadmap indicating how the selected architecture framework can be implemented to enable the artefact to function as a reference architecture.
Development:	1. Investigate the implications in terms of architecture framework dimensions of enabling the artefact to function as a reference architecture. 2. Determine how the selected architecture framework can be implemented within the artefact through the specification of architecture framework dimensions. 3. Compile an artefact development roadmap.
Artefact version:	Artefact development roadmap

4.5.4 Fourth design cycle

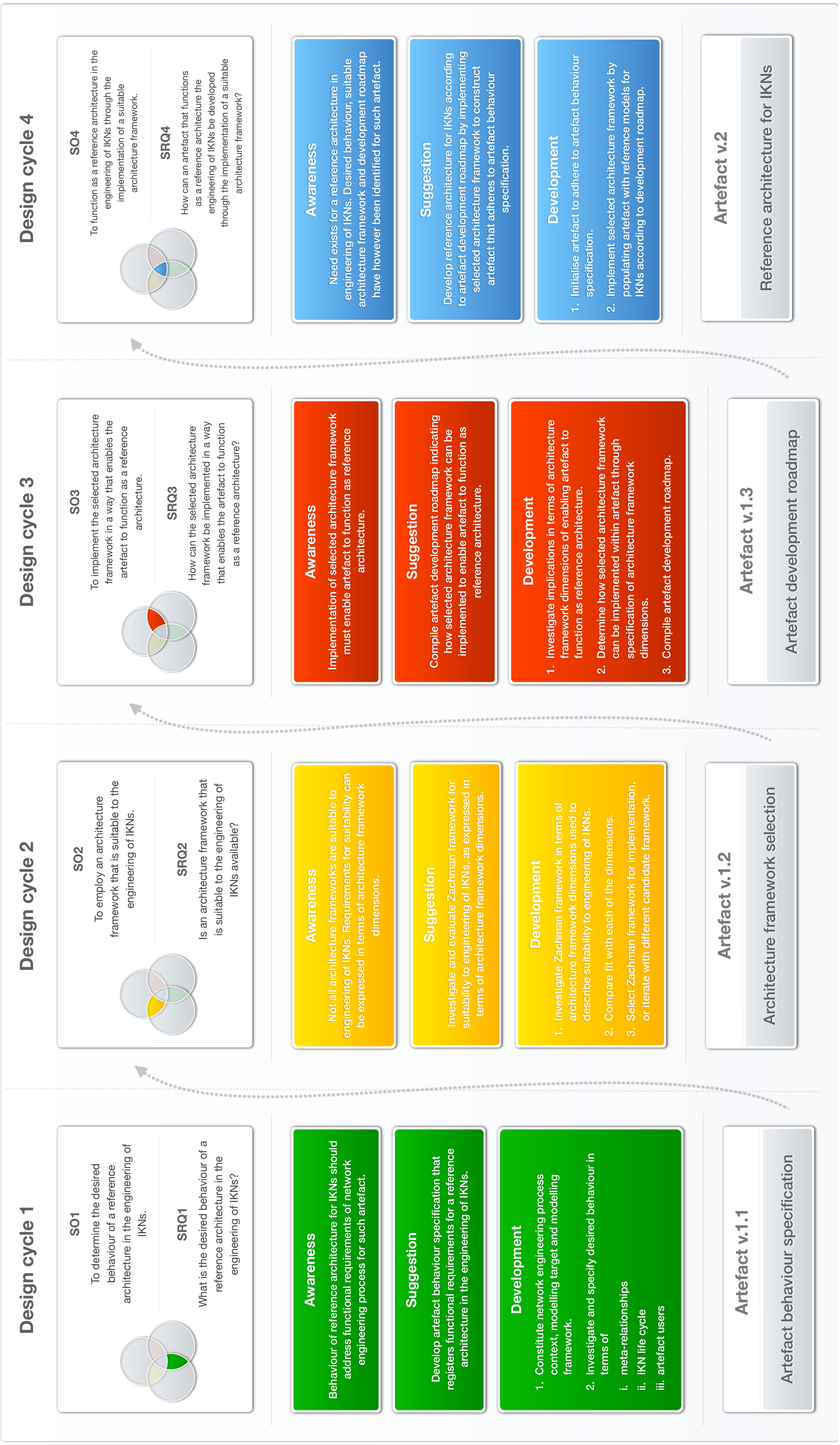
The cycle incrementally improves the artefact produced in the third cycle and is initiated by SO4 and SRQ4 and considers the requirements intersection of scenarios, technical requirements and functional requirements (refer to the blue element in Figure 4.7). This cycle is discussed in chapter 8, and follows the structure of Awareness, Suggestion and Development of the final artefact.

Requirement:	The artefact must function as a reference architecture in the engineering of IKNs through the implementation of a suitable architecture framework.
Solution objective:	To function as a reference architecture in the engineering of IKNs through the implementation of a suitable architecture framework. (SO4)

Secondary research question:	How can an artefact that functions as a reference architecture in the engineering of IKNs be developed through the implementation of a suitable architecture framework? (SRQ4)
Awareness:	There is a need for a reference architecture in the engineering of IKNs. The desired behaviour of such an artefact, a suitable architecture framework and an artefact development roadmap have, however, been identified previously.
Suggestion:	Develop a reference architecture to be used in the engineering of IKNs by adhering to the artefact behaviour specification and implementing the selected architecture framework according to the artefact development roadmap.
Development:	1. Initialise artefact to adhere to the artefact behaviour specification. 2. Implement the selected architecture framework by populating the artefact with reference models for IKNs according to the artefact development roadmap.
Artefact version:	Reference architecture for IKNs.

4.5.5 Overview of reasoning in design cycles

The alignment of the four design cycles with the compound requirement perspectives and resulting solution objectives is illustrated in Figure 4.8. The figure also shows the internal reasoning in each of the design cycles and the incremental development of the solution artefact. It is an application of the generic DSR reasoning presented in Figure 3.4.



SO1
To determine the desired behaviour of a reference architecture in the engineering of IKNs.

SRQ1
What is the desired behaviour of a reference architecture in the engineering of IKNs?

Awareness
Behaviour of reference architecture for IKNs should address functional requirements of network engineering process for such artefact.

Suggestion
Develop artefact behaviour specification that registers functional requirements for a reference architecture in the engineering of IKNs.

Development
1. Constitute network engineering process context, modelling target and modelling framework.
2. Investigate and specify desired behaviour in terms of
   i. meta-relationships
   ii. IKN life cycle
   iii. artefact users

SO2
To employ an architecture framework that is suitable to the engineering of IKNs.

SRQ2
Is an architecture framework that is suitable to the engineering of IKNs available?

Awareness
Not all architecture frameworks are suitable to engineering of IKNs. Requirements for suitability can be expressed in terms of architecture framework dimensions.

Suggestion
Investigate and evaluate Zachman framework for suitability to engineering of IKNs, as expressed in terms of architecture framework dimensions.

Development
1. Investigate Zachman framework in terms of architecture framework dimensions used to describe suitability to engineering of IKNs.
2. Compare fit with each of the dimensions.
3. Select Zachman framework for implementation, or iterate with different candidate framework.

SO3
To implement the selected architecture framework in a way that enables the artefact to function as a reference architecture.

SRQ3
How can the selected architecture framework be implemented in a way that enables the artefact to function as a reference architecture?

Awareness
Implementation of selected architecture framework must enable artefact to function as reference architecture.

Suggestion
Compile artefact development roadmap indicating how selected architecture framework can be implemented to enable artefact to function as reference architecture.

Development
1. Investigate implications in terms of architecture framework dimensions of enabling artefact to function as reference architecture.
2. Determine how selected architecture framework can be implemented within artefact through specification of architecture framework dimensions.
3. Compile artefact development roadmap.

SO4
To function as a reference architecture in the engineering of IKNs through the implementation of a suitable architecture framework.

SRQ4
How can an artefact that functions as a reference architecture the engineering of IKNs be developed through the implementation of a suitable architecture framework?

Awareness
Need exists for a reference architecture in engineering of IKNs. Desired behaviour, suitable architecture framework and development roadmap have however been identified for such artefact.

Suggestion
Develop reference architecture for IKNs according to artefact development roadmap by implementing selected architecture framework to construct artefact that adheres to artefact behaviour specification.

Development
1. Initialise artefact to adhere to artefact behaviour specification.
2. Implement selected architecture framework by populating artefact with reference models for IKNs according to development roadmap.

Figure 4.8: Design cycle driver alignment and reasoning

4.6 Conclusion

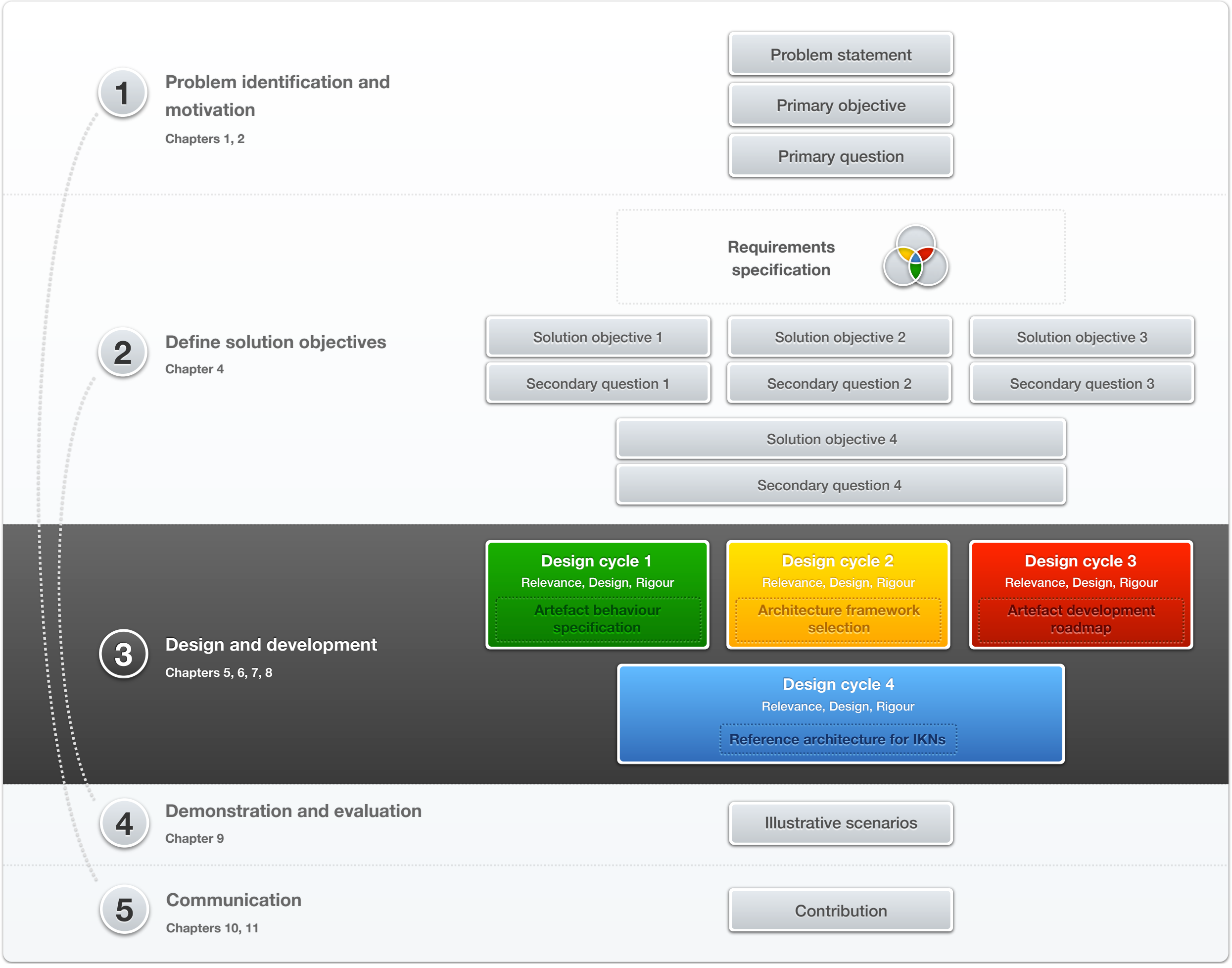
This chapter discussed the second activity in the research design employed in this study that defined objectives for a solution to the problem articulated in Part 1. After providing some background on solution objectives and requirement specification in section 4.2, these solutions objectives were inferred rationally through a structured approach to requirements specification. This approach considers functional requirements, technical requirements and implementation scenarios for a solution (refer to section 4.3). These are combined to form compound requirement perspectives that are then translated into solution objectives with accompanying secondary research questions that direct the search process toward achieving these objectives (refer to section 4.4).

The next part of this document, Part 3, presents the third activity in the research design employed in the study and discusses the design and development of the solution artefact. This design and development process follows the four design cycles presented in section 4.5 and incrementally develops the artefact by focusing on each of the solution objectives with its associated secondary research question, as identified in this chapter.

Part 3 – Design and development

The third part of the document contains four chapters (chapters 5, 6, 7 and 8) that represents the third activity of the research design employed in this study. This activity designs and develops an artefact that aims to achieve the solution objectives identified in Part 2. To this end, the artefact is designed and developed incrementally by sequentially considering each of the solution objectives and their associated secondary research questions in four consecutive design cycles.

Each of the chapters in this part of the document discusses a single design cycle. Chapter 5 presents the first (green) cycle, chapter 6 the second (yellow) cycle and chapter 7 the third (red) cycle. These cycles represent the divergent phases of the development process and progressively update the requirement frame of reference as design decisions are made and the uncertainty about the final solution artefact is reduced. Chapter 8 discusses the fourth (blue) design cycle that represents the convergent phase of the development process by combining the design decisions and development from the previous cycles to instantiate the reference architecture for IKNs. This final version of the artefact is demonstrated and evaluated in Part 4 of this document.



5. Artefact behaviour specification

5.1 Introduction

This chapter presents the first cycle in the incremental design and development process of the reference architecture for IKNs. The cycle considers the compound requirement perspective that combines scenarios and functional requirements (refer to green intersection in Figure 5.1).

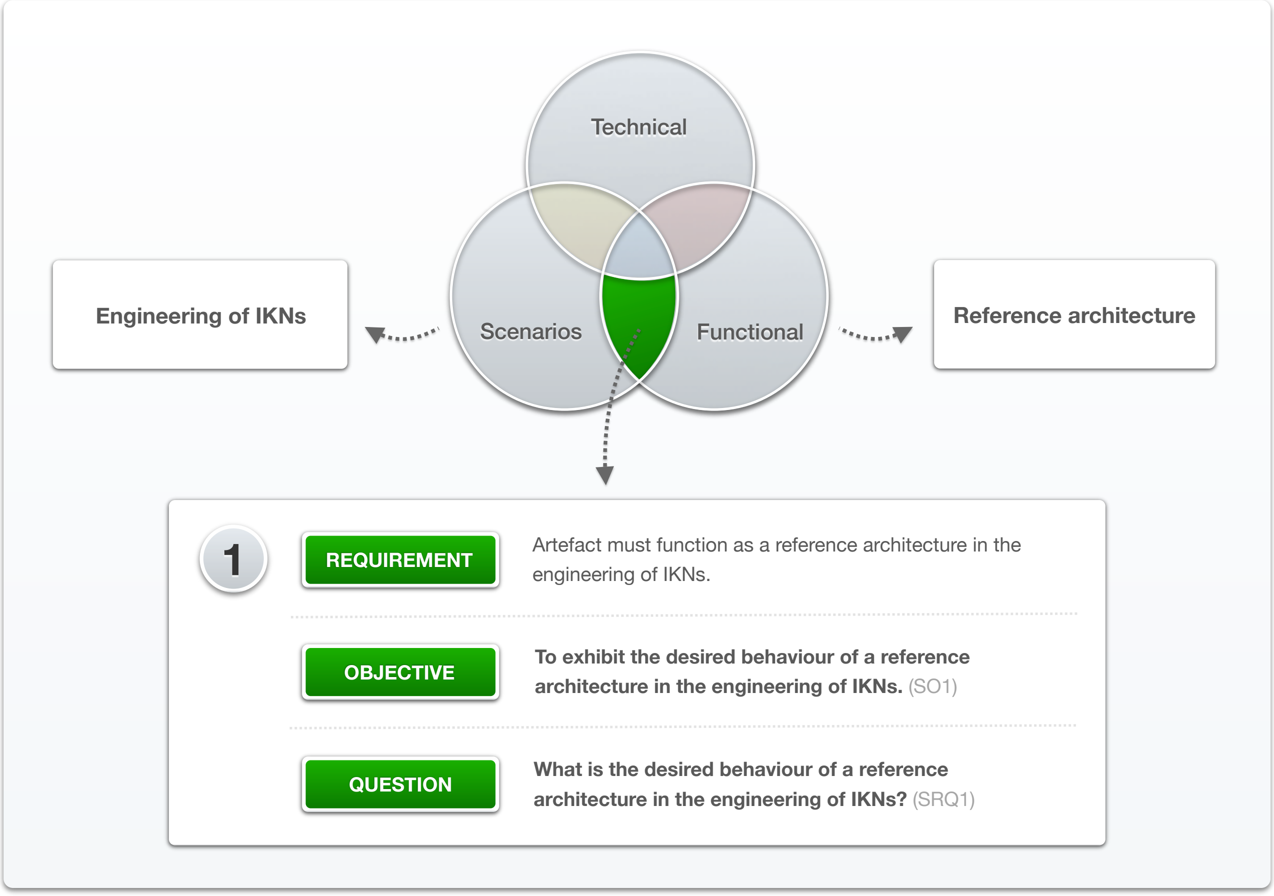
The design cycle addresses the first solution objective (SO1) by identifying the desired behaviour of a reference architecture in the engineering of IKNs. The search process in this chapter is directed by the first secondary research question (SRQ1), which asks what such a desired behaviour would entail.

This chapter initiates the design and development contributed in this study, with its contribution delivered in the form of an *artefact behaviour specification* that *indicates the desired behaviour of a reference architecture in the engineering of IKNs*. This behaviour specification achieves the solution objective (SO1) and therefore circumscribes, and restricts, the activities of the design cycle, which leads to the initiation of subsequent design cycles that further develop the artefact in chapters 6, 7 and 8.

The solution objective and associated secondary research question that drives the design cycle are presented in section 5.2. Section 5.3 presents background selected from existing literature that forms the basis for the ensuing discussion in this chapter. Design reasoning follows with section 5.4 discussing the awareness that the behaviour of a reference architecture for IKNs should address the functional requirements of the network engineering process for such an artefact. Section 5.5 discusses the suggestion to develop an artefact behaviour specification that registers the functional requirements for a reference architecture in the engineering of IKNs. Section 5.6 documents the development of the artefact behaviour specification, including the constitution of a network engineering process context, modelling target and modelling framework. The desired behaviour of the artefact is then investigated in terms of the relationships between relevant axes of the modelling framework. Section 5.7 concludes this chapter.

5.2 Design cycle drivers

This section presents the solution objective and secondary research question that drives this design cycle. This is done in order to position the design cycle (and chapter) in the context of the overall design and development work performed in Part 3 of the research design.



Artefact must function as a reference architecture in the engineering of IKNs.

To exhibit the desired behaviour of a reference architecture in the engineering of IKNs. (SO1)

What is the desired behaviour of a reference architecture in the engineering of IKNs? (SRQ1)

Figure 5.1: Drivers of first design cycle

The design cycle is driven by SO1 and SRQ1 as derived by considering the requirements intersection of scenarios and the functional requirement (refer to Figure 5.1).

Compound requirement:	The artefact must function as a reference architecture in the engineering of IKNs.
Solution objective:	To exhibit the desired behaviour of a reference architecture in the engineering of IKNs. (SO1)
Research question:	What is the desired behaviour of a reference architecture in the engineering of IKNs? (SRQ1)

5.3 Background

This section presents literature that directly informs the awareness for the first (green) design cycle.

5.3.1 Functional perspectives on the engineering of IKNs

This section discusses elements of the scenarios that specifically pertain to the functional requirement of the artefact.

The investigation into the engineering of IKNs which is presented in this section stems from the discussion of enterprise engineering in section 2.6. The field is based on the following definition of an enterprise:

> *An enterprise is a complex, socio-technical system that comprises interdependent resources of people, information and technology that must interact with each other and their environment in support of a common mission.* (Giachetti 2010:4)

In the context of this definition, an enterprise may also be described as:

> *…an intentionally created cooperative of human beings with a certain societal purpose* (Dietz et al. 2013; Daft 2012)

This study takes the view that an IKN is an example of such an entity. It can be shown from the CNO taxonomy (Camarinha-Matos & Afsarmanesh 2008a) and the methodology for IKNs (Schutte 2010) that these networks allow for planning and are not merely ad hoc network phenomena that do not fit the above definition of an enterprise.

The intentional nature of IKN creation requires design activities and enterprise engineering is seen as a suitable discipline to contribute in this regard (Dietz et al. 2013). In this context, design is not seen as a mechanistic, inhibiting factor to the creation of networks, but should rather be interpreted broadly and seen as “devising courses of action aimed at changing existing (network) situations into preferred ones” (Simon 1996). Design concerns the interaction between understanding and creation (Winograd & Flores 1986). The discipline of enterprise (and by extension also IKN) engineering should be viewed from this perspective (Dietz et al. 2013).

There is recognition for the fact that networks pose a fundamentally different challenge than traditional enterprises to design and engineering methods due to their extremely high level of complexity (Camarinha-Matos & Afsarmanesh 2008a). Not only do IKNs inherit all the complexity of their constituent enterprises, but they add the concepts of networking, collaboration and autonomous behaviour of these socio-technical subsystems. This is, however, by no means an argument against the extension of enterprise engineering to the field of IKNs. To the contrary, enterprise engineering is viewed as a critical enabler in making this high level of complexity intellectually manageable (Dietz & Hoogervorst 2008). Enterprise engineering achieves this role as an enabler by introducing a separation of enterprise construction and function, i.e. moving towards an ontological view of the enterprise by identifying the constructional essence of the enterprise independent of implementation (Dietz & Hoogervorst 2012).

This constructional essence of the enterprise can be described by a white-box model (Dietz et al. 2013). Such a model of the enterprise describes its architecture and includes a set of descriptive representations or models that are required in order to create and/or manage the enterprise (Kappelman & Zachman 2013). The architecture is also the baseline for changing the enterprise once it is implemented if the descriptive representations are retained and care is taken to ensure that the descriptions are maintained consistently with the implemented enterprise (Kappelman & Zachman 2013).

The earlier concept of design as a vehicle between current and future situations can now be interpreted in these architectural terms by stating that it is possible to migrate an enterprise from a current (‘as-is’) architectural state to a future (‘to-be’) architectural state through enterprise engineering (Vernadat 1996; Schekkerman 2004). This enterprise engineering process involves a multi-phase approach that coordinates strategic, operational, and organisational demands in migrating the enterprise from a current to a future state (refer to Figure 5.2). This thinking is influenced by the Master Planning approach presented in PERA (Williams et al. 1996) and relates to the concept of roadmaps. Roadmaps list individual increments of change according to a timeline to show progression from the current state to a future state of business processes, systems, information and technology (Vernadat 1996; DeBoever et al. 2010).

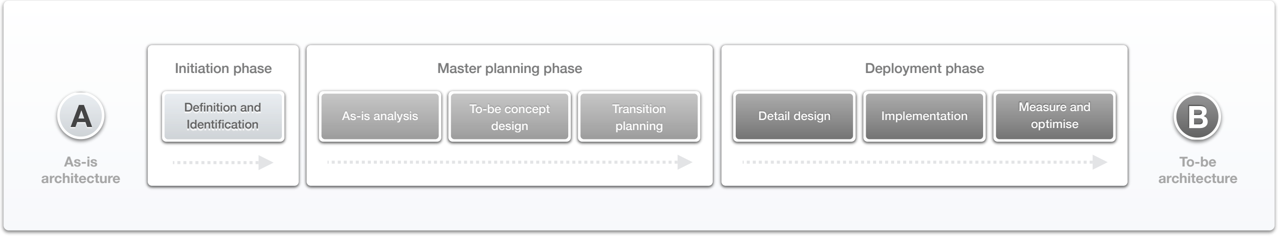


Figure 5.2: The enterprise engineering process

(adapted from Department of Industrial Engineering, Stellenbosch University 2013)

In the PERA-influenced paradigm of enterprise engineering, a typical enterprise engineering project includes an initiation phase, a master planning phase, where as-is analysis, to-be concept design and transition planning are performed, and a deployment phase, where detail design, implementation, measurement and optimisation are performed (refer to Figure 5.2).

It is possible to motivate that, from an epistemological point of view, enterprise engineering as a discipline has a pragmatic philosophical underpinning, with architectural knowledge generated about the enterprise as applied through action to bring about change (refer to section 3.2). When engineering an IKN by migrating it between states, two primary types of design scenarios can be encountered by the enterprise engineer (Department of Industrial Engineering, Stellenbosch University 2013):

1. **Greenfield design:** This is a design scenario where a totally new network needs to be developed or engineered. The current state is therefore empty and the new network is created through the design of a to-be state and migrating the network to this to-be state, i.e. network implementation.
2. **Redesign:** This is a scenario where a part or parts of an existing network need to be improved through re-engineering. A current state therefore exists and a to-be state is designed and the network is migrated to the to-be state, which results in network refinement.

Recurring versions of these scenarios are part of the life cycle of IKNs and these versions form part of the environment with which the reference architecture for IKNs will be tasked to interact. Another component of this environment is the stakeholders in the network engineering process. In the classic enterprise engineering context of the PERA Master Planning approach (Williams et al. 1996), a number of these stakeholders were identified:

- **Champion:** An individual, knowledgeable in enterprise integration technology, who is pro-active in promoting it, and who serves as a catalyst to push towards such applications.
- **Initiating sponsor:** A high-level management individual who lends support and prestige to the work of the champion and clears corporate obstacles.
- **Steering committee:** A group of stakeholders in the business unit for which the enterprise integration programme is being developed, which lends direct management guidance and support to that effort.
- **Enterprise integration planning team:** The team that performs the actual analysis and preparation of the plan under the guidance of the steering committee.

These stakeholders in the enterprise engineering process are reconsidered in the context of IKNs in section 5.6.5.6.

5.3.2 Life cycle of IKNs

This section discusses the life cycle of IKNs as a fundamental element of the scenarios in which the artefact is required to function.

The methodology for IKNs developed by Schutte (2010) describes the primary enterprise engineering activities that can be performed on an IKN as its life history unfolds. These activities represent the life cycle of such networks (Noran 2004) and five high-level phases can be distinguished. The phases are named after these prevalent activities, namely *design, implementation, operation, refinement* and *phase-out*. Such an activity-based view of the life cycle of IKNs as presented by the methodology is relevant to a reference architecture for IKNs, as it provides insight into the changing environment in which the artefact will be used to engineer IKNs. Each of these phases in the methodology for the development of IKNs is discussed in the following sections.

5.3.2.1 Design phase

The design phase is the most important phase in the establishment of an IKN because properly planning and designing the network enables the stakeholders to optimise the benefits of being involved. The design phase is divided into three sub-phases (Schutte 2010):

1. Determination of vision, strategy, domain and stakeholders;
2. Establishment of network requirements;
3. Detail design and planning of the network.

In order to determine the vision, strategy, domain and stakeholders of the new IKN, the need for knowledge creation and sharing to enable innovation has to be articulated as this defines the purpose of the network (Anklam 2007). A knowledge vision and strategy further enables an effective IKN as that ensures the relevance of future knowledge work to the innovative purpose of the network. Stakeholders are identified, analysed and selected according to their contribution to the success of innovation and knowledge management initiatives. Financial feasibility is often a measure of the success of an IKN, as an active network requires inputs that are linked to investment from stakeholders (Schutte 2010). When organisations create and share knowledge related to innovations with external stakeholders, as is the case with IKNs, the ownership and protection of intellectual property rights is a primary concern and suitable strategy has to be identified in this area as well.

Establishing the requirements for the IKN commences with an analysis of the requirements for knowledge in the network's innovation chain and linking that to knowledge forms, content and possible sources. Appropriate knowledge work processes and their facilitating conditions are also selected to align with these knowledge requirements (Back et al. 2005). Furthermore, performance measures and targets are identified and precautions are taken to ensure that it is possible to measure and achieve these targets. This high-level network design enables the documentation of financial support requirements, as well as the identification of contractual agreements that need to be put in place.

The detail design and planning of the network is initiated by the structural design of the network, indicating relationships amongst stakeholders throughout the innovation chain. The network's approach to governance is also planned in detail. The required knowledge work processes are designed, and include the supporting infrastructure in terms of organisational tools and ICT architecture. Further planning activities for the network include the development of planning documents that address network deliverables, milestones, communication and change management. Performance measures that were previously only considered at a high level are articulated in detail to include specific measures, targets, corrective actions and incentives. Financial and contractual models are also planned and developed in detail in this phase.

5.3.2.2 Implementation phase

The implementation phase is divided into two sub-phases (Schutte 2010):

- Planning and preparing for the implementation of the network
- Rolling out and implementing the network

Planning and preparing for the implementation of the network include approaching the final group of stakeholders to confirm their commitment, as well as planning the actual network rollout activities. Organisational procedures are defined in detail and the previously designed organisational and ICT architectures are built or procured. It is also important to verify that the necessary funding is available to implement the IKN.

Upon completion of the implementation planning and preparation, actual network rollout can be initiated. Stakeholder participation is facilitated through relationship development and internal role player identification to ensure that the identified knowledge vision can be achieved and innovation can be facilitated in the network. Kick-off workshops and meetings are held and the network is organised through public and private activities to develop a community amongst all participants. The built or acquired organisational and ICT architectures are rolled out to all network stakeholders and performance measures are implemented.

5.3.2.3 Operation phase

Innovation projects are executed in this phase and the operation of an IKN therefore requires that stakeholder expectations be managed and participation in network activities is ensured, with contracts being renewed appropriately (Schutte 2010). It is also necessary to ensure that continuous funding is available and that network activities are evolved, sustained and facilitated. The organisational architecture and ICT architecture should be maintained, as these assets are required for documenting, storing and categorising knowledge, searching for new knowledge, communicating with other network stakeholders, sharing experiences, amongst others.

5.3.2.4 Refinement phase

In the design phase, a vision with derived goals were identified for the IKN and it is important that the achievement of these goals is assessed and the operation of the network subsequently be adjusted and refined accordingly (Schutte 2010). This phase of the methodology for IKNs therefore represents a stage of the network life cycle that occurs concurrently with the operation stage. In the refinement phase, activities can be executed in parallel with, and be linked to, operational network activities. Network refinement activities may include performance measurement, the expansion of stakeholders and funding, and the enhancement of the network architecture.

Performance measurement plays an important role in the network refinement process , as it is necessary to assess the impact of the IKN on the business goals of stakeholders (Schutte 2010). This performance measurement could include a network health check, upon which ailing network symptoms are identified and corrective actions initiated (Back et al. 2005). Such a health check could for instance indicate that the discovery of new knowledge and progress with specific innovation projects requires knowledge sources and capabilities that are not present in the current network. This would necessitate the involvement of new network stakeholders that can introduce the required knowledge and capabilities. These enhancements to the network imply changes to the structural design of the network, with potential implications for funding contracts, intellectual property arrangements, as well as the organisational and ICT architectures.

5.3.2.5 Phase-out phase

IKNs that have achieved their goals and have a decreasing activity level, or that register a poor performance during a health check and show no improvement after subsequent corrective actions have been implemented, may decide to terminate their activities (Schutte 2010). In such cases the network is phased out while managing the assets that have been created during its operation.

The phase-out is divided into two sub-phases:

1. Plan phase-out
2. Execute phase-out

Successful IKNs can achieve their goals by executing the innovation project(s) that necessitated the institution of the network, thus rendering the network unnecessary (Schutte 2010). It could also be that a network that was created and operated around a specific theme or domain can reach the end of its life cycle (Back et al. 2005). In both these scenarios the network phase-out is initiated by planning the phase-out communication, i.e. how the motivations and planning for the phase-out will be conveyed to network stakeholders. Crucially, the planning of the architectural closeout follows, as the architecture often contains the captured benefits and knowledge created by the network during its operation (Schutte 2010). It is important to ensure that this value is not lost, but rather remains accessible to stakeholders for future innovation activities. The planning of the architecture closeout therefore needs to include the archiving and ensured future accessibility of the knowledge base. The contractual closeout of the network also has to be planned.

Execution of the network phase-out can commence once planning is completed and begins with documenting the results and performance of the network, including benefit analysis and financial closeout. The architecture is subsequently archived and migrated to some or all of the stakeholders, or closed altogether. The final element of the phase-out is contractual closeout based on prior planning. It must resolve any outstanding contractual issues that may still exist and close all contracts.

5.3.3 Behaviour of reference architectures

This section focuses on the behaviour of reference architectures that are relevant in the enterprise domain, as implied by the scenarios. For more background on reference architectures in general, refer to section 2.7.1.

Much can be learnt about the behaviour of reference architectures by examining the working definition of these constructs that is adopted by this study:

> “A reference architecture aims at structuring the design of architectures for a given domain by defining a unified terminology, describing the functionality and roles of components, providing template components, giving example architectures, and defining a development methodology.” (Camarinha-Matos & Afsarmanesh 2008a)

The primary behavioural trait of a reference architecture is therefore the provision of inputs for the design of particular architecture descriptions for instances in the class of enterprises covered by the reference architecture. This is done through the provision of reference models that serve as the basis for particular models that accurately describe the entity instance (Cloutier et al. 2010). In doing this, reference architectures represent a coherent set of design principles to be used in a given domain (Camarinha-Matos & Afsarmanesh 2008a).

The characteristics of reference architectures that are particularly relevant to the enterprise domain can be visualised along three dimensions, as illustrated in Figure 5.3. This framework for describing reference architectures shows similarities with the framework that forms the basis of GERAM (IFIP-IFAC Task Force 2003). The three dimensions for describing enterprise reference architectures are (IFIP-IFAC Task Force 2003; Department of Industrial Engineering, Stellenbosch University 2013):

1. **Life cycle:** This dimension describes the life cycle phases of an enterprise and the typical activities that are performed during each phase of the enterprise life cycle, e.g. identification, conceptual development, design, implementation and operation.
2. **Views:** These dimensions enable the reference architecture to look at the design or redesign of the enterprise from different perspectives. These perspectives can be based either on function (e.g. manufacturing, information or organisation) or role (e.g. planner, designer or manager).
3. **Instantiation:** This dimension defines whether the enterprise reference architecture is generic (applicable to all enterprise types or industries), partial (only applicable to a specific industry or enterprise type), or specific (customised for a specific enterprise).

The provision of reference models that serve as the basis for particular models that describe the enterprise instance (i.e. the fundamental behaviour of a reference architecture) can thereby be characterised according to the phase of the enterprise life cycle that the model addresses. Reference models can

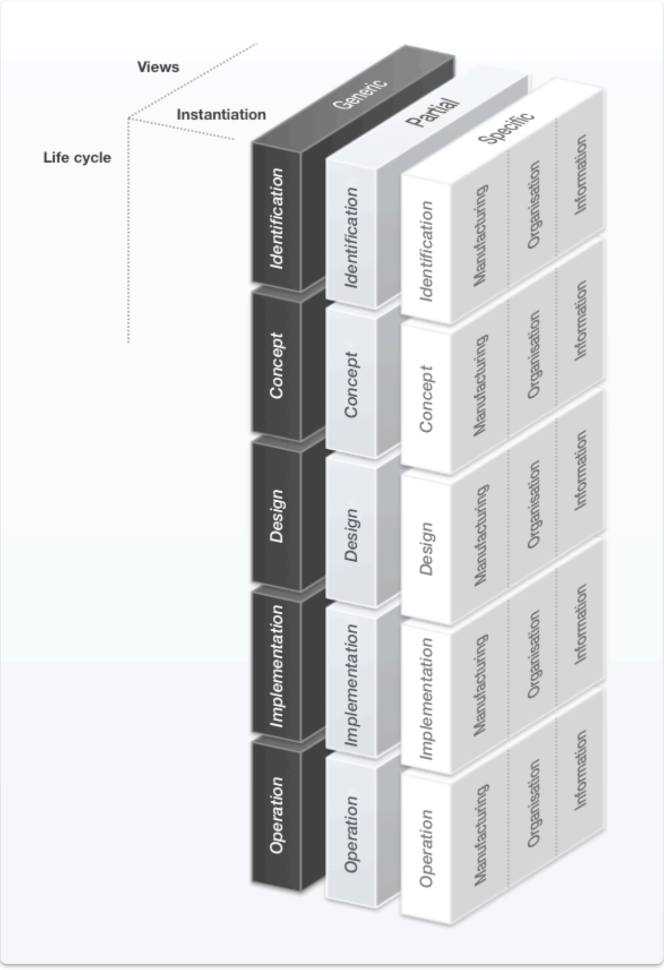


Figure 5.3: Enterprise reference architecture dimensions

(adapted from Department of Industrial Engineering, Stellenbosch University 2013)

furthermore vary in terms of their perspective on the enterprise's architecture, while the reference architecture itself exists at a level of abstraction for which the models represent a complete set. The detailed behaviour of an enterprise reference architecture is determined by its characteristics as described along these three axes. Two main classes of enterprise reference architectures can be identified (Williams 1994):

1. **Type 1:** These reference architectures only deal with the structural arrangement (design) of a physical system such as the manufacturing component of the enterprise, or the structure of the complete enterprise. The Zachman enterprise framework (Kappelman & Zachman 2013) is an example of such a reference architecture.
2. **Type 2:** These reference architectures deal with the structural arrangement (organisation) of the development and implementation of a project or programme such as an enterprise engineering programme. They illustrate the life cycle of the project developing the enterprise and therefore have

a specific reference model that addresses the flow of enterprise engineering efforts along the life cycle axis of Figure 5.3. PERA (Williams 1998) is an example of such a reference architecture.

The behaviour of enterprise reference architectures can be summarised by considering their role in the enterprise engineering process, as illustrated in Figure 5.4. This diagram shows the enterprise engineering process being used as a vehicle to migrate the enterprise from a current state with an as-is architecture to a preferred future state with a to-be architecture (refer to section 5.3.1). Depending on its intention, the enterprise reference architecture contains reference models and methods, and possibly a life cycle process or activity sequence, and provides a number of inputs to this process. Some of these inputs are generally associated with a Type 1 reference architecture (1) and others with a Type 2 reference architecture (2), as illustrated.

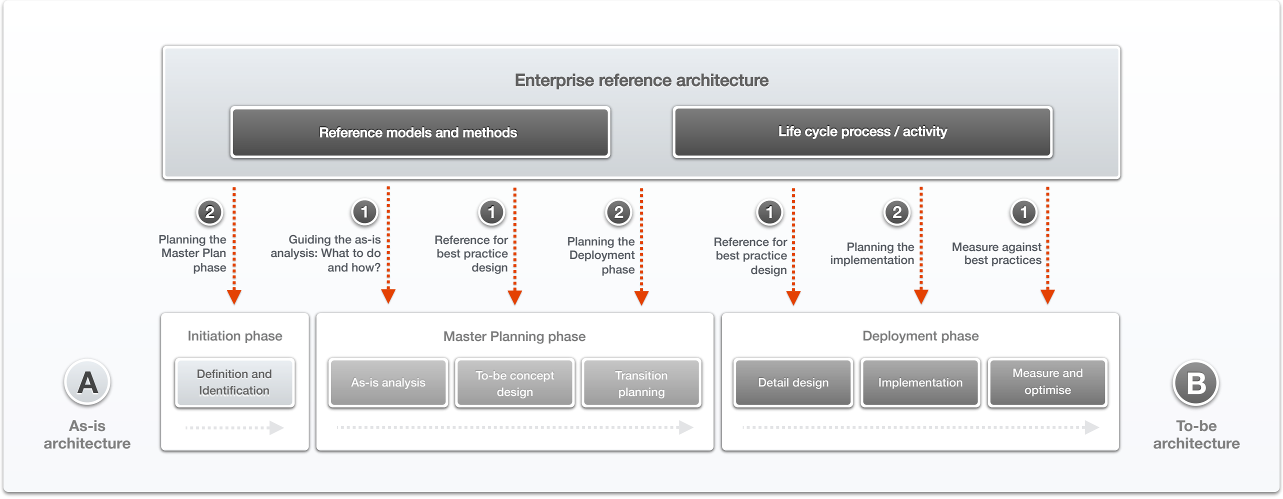


Figure 5.4: Role of a reference architecture in the enterprise engineering process

(adapted from Department of Industrial Engineering, Stellenbosch University 2013)

Behaviour typically associated with a Type 1 reference architecture includes guidance of the as-is analysis, references for best practice design of the to-be concept, references for best practice detail design and inputs for the measurement and optimisation of the architecture. Behaviour typically associated with a Type 2 reference architecture includes guidance for the definition and identification of the enterprise engineering project. It is also associated with inputs for the planning of the deployment phase and specifically the implementation of the newly engineered enterprise architecture. In both cases the enterprise reference architecture contributes to the enterprise engineering process through the provision of appropriate reference models that have as focus either the structural arrangement of the enterprise or the engineering project that aims to design and develop the enterprise.

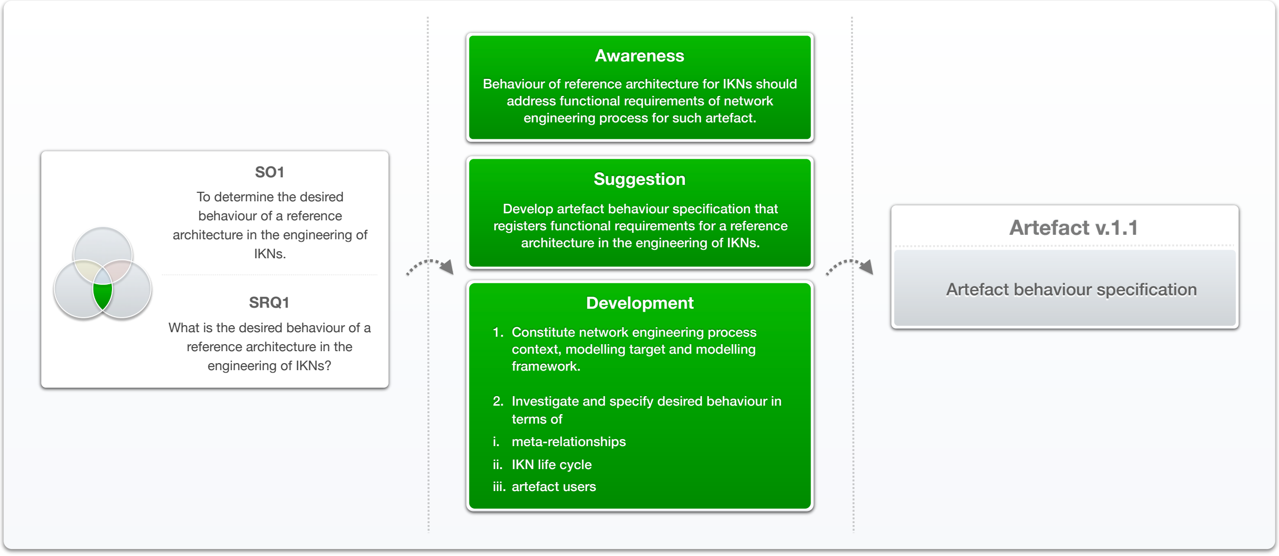
This section presented functional perspectives on the engineering of IKNs, as well as background information on the IKN life cycle and the behaviour of reference architectures. This discussion serves as context for the problem awareness that initiates design reasoning in the following section.

5.4 Awareness

This section presents the awareness step of the design reasoning in the first (green) design cycle.

Given SO1 and SQ1 (refer to section 5.2), the awareness of the problem considered in this design cycle can be stated as follows (refer to Figure 5.5):

> **The behaviour of a reference architecture for IKNs should address the functional requirements of the network engineering process for such an artefact.**



SO1
To determine the desired behaviour of a reference architecture in the engineering of IKNs.

SRQ1
What is the desired behaviour of a reference architecture in the engineering of IKNs?

Awareness
Behaviour of reference architecture for IKNs should address functional requirements of network engineering process for such artefact.

Suggestion
Develop artefact behaviour specification that registers functional requirements for a reference architecture in the engineering of IKNs.

Development

1. Constitute network engineering process context, modelling target and modelling framework.
2. Investigate and specify desired behaviour in terms of
   i. meta-relationships
   ii. IKN life cycle
   iii. artefact users

Artefact v.1.1
Artefact behaviour specification

Figure 5.5: Reasoning in first design cycle

This problem awareness articulates that the problem considered in the study can in part be solved with an artefact that exhibits a desired behaviour when implemented in its expected environment, i.e. it must function as a reference architecture in the engineering of IKNs. As discussed in section 5.3.3, the term ‘reference architecture’ is linked to a certain set of artefact behavioural traits, e.g. the provision of a unified terminology for a domain and the description of the functionality and roles of components in systems in the domain. The exact embodiment of these traits, however, differs between reference architectures, and it is not known what the exact desired behaviour of a reference architecture in the engineering of IKNs is. This desired behaviour may include elements that are specific to IKNs and it can therefore not be assumed that the typical behaviour of a reference architecture accurately addresses the requirements for a solution to the identified research problem. For this reason, a better understanding is required before further design and development can take place.

It should also be noted that the solution objective specifically states that “the desired behaviour of a reference architecture *in* the engineering of IKNs” must be determined, and not “the desired behaviour of a reference architecture *for* the engineering of IKNs”. This corresponds to the intention of the reference architecture for IKNs to deal with the structural arrangement of the network from an engineering point of view, rather than providing inputs for the planning of the programme to engineer it. Hence, the awareness of the problem articulates the need for a Type 1 reference architecture (refer to section 5.3.3), given the existence of the methodology for IKNs (Schutte 2010) which already provides the inputs to the network engineering process that would be expected from a Type 2 reference architecture.

5.5 Suggestion

This section presents the suggestion step of the design reasoning in the first (green) design cycle.

The awareness of the problem considered in this design cycle leads to the following suggestion for a possible solution (refer to Figure 5.5):

> **Develop an artefact behaviour specification that registers the functional requirements for a reference architecture in the engineering of IKNs.**

In order to ensure that the reference architecture for IKNs exhibits the desired behaviour for a solution when implemented in its environment, an artefact behaviour specification should be developed. This entails identifying a modelling target and, subsequently, constituting the network engineering process as context. Furthermore, a modelling framework should be specified to identify the perspectives from which the reference architecture for IKNs will be designed and developed. The artefact behaviour specification should be positioned within this modelling framework and describe the functionality of a reference architecture for use in the engineering of IKNs in terms of the following:

- Meta-relationships with other entities in the environment;
- The life cycle of IKNs;
- Users of the artefact.

An artefact behaviour specification featuring the above elements will answer SQ1 and thereby achieve SO1.

5.6 Development

This section presents the development step of the design reasoning in the first (green) design cycle.

5.6.1 Reference modelling target and enterprise engineering process

The target for reference models included in the artefact is the typical configuration of an IKN as viewed at the network level, i.e. the network is seen as a distinct entity of which the constituent member organisations are merely components. The reference architecture for IKNs therefore describes the commonalities that can be generalised from multiple instances of this type of network. The reference modelling target can also be seen as a specialisation of CNOs in general and the aim is accordingly to describe a certain subset of this larger pool of organisational forms (refer to section 2.2).

To assist in specifying the desired behaviour, the enterprise engineering process with its relation to reference architectures as described in section 5.3.3 and illustrated in Figure 5.4 will be used as a point of departure to describe the environment with which the artefact will be interacting. In this scheme, the reference modelling target is migrated from an as-is architecture to a to-be architecture with the help of an enterprise reference architecture that assists through the provision of reference models and methodological guidance.

5.6.2 Modelling framework

This section presents the perspectives from which the reference architecture for IKNs may be designed and developed. These perspectives are subsequently combined into a coherent modelling framework.

IKNs are regarded as having a highly complex nature that can be attributed to both their networked organisational configuration, as well as the collaborative phenomena they support. A reference architecture for IKNs requires a modelling framework to capture this complexity. Based on prior work in the domain of enterprise reference architectures and architecture frameworks (IFIP-IFAC Task Force 2003), as well as CNOs (Camarinha-Matos & Afsarmanesh 2008a; Vesterager et al. 2003), the modelling framework illustrated in Figure 5.6 is constituted for the reference architecture for IKNs.

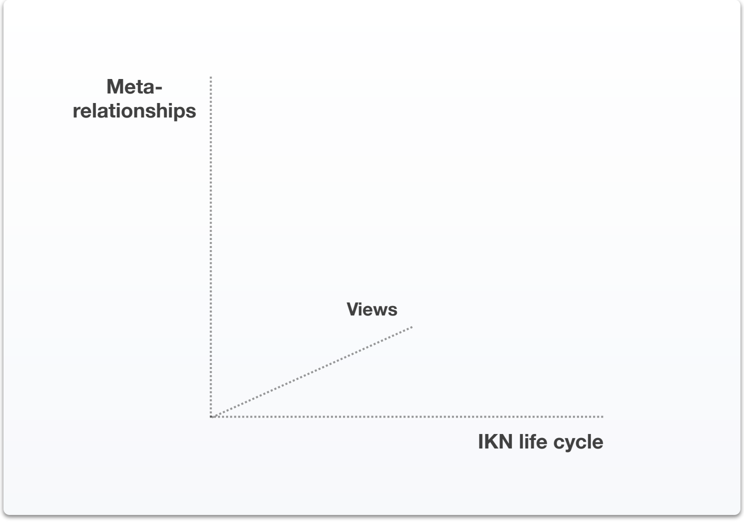


Figure 5.6: Modelling framework for a reference architecture for IKNs

This modelling framework forms the skeleton of the reference architecture for IKNs and its constitution is the starting point of the incremental development of the solution artefact developed in this study. As such, its detailed design forms the bulk of chapters 5, 6 and 7 before chapter 8 is populated with reference models to produce the final version of the solution artefact.

While the modelling framework that will be used to articulate the characteristics of the reference architecture for IKNs draws on modelling frameworks previously developed in the fields of reference modelling for collaborative networks or enterprise architecture, none of these frameworks are adopted verbatim. These frameworks are deemed to be too generic to structure the constituent parameters of the required reference architecture at the same level of maturity of other reference artefacts that are available for IKNs, e.g. the methodology developed by Schutte (2010).

The modelling framework addresses the complexity involved with modelling IKNs by describing the modelling space along three axes or perspectives:

- **Meta-relationships**. This perspective is used to indicate how the reference architecture for IKNs relates to other artefacts in its environment on a meta-level. This corresponds to the “genericity” axis found in the GERAM framework (IFIP-IFAC Task Force 2003) and the “modelling intent” axis included in the ARCON framework (Camarinha-Matos & Afsarmanesh 2008a), and is specifically instituted to describe the relationships between reference architectures, architectures and instantiations within the modelling space.
- **IKN life cycle**. The life cycle perspective is introduced to describe the diverse actions that contextualise the IKN engineering process. The GERAM modelling framework includes a generic enterprise life cycle (IFIP-IFAC Task Force 2003) and the ARCON framework includes an axis that more accurately describes the life cycle of CNOs in general (Camarinha-Matos & Afsarmanesh 2008a). It is, however, possible to substitute these more generic axis descriptions with a designation based on the life cycle description of IKNs as described in the methodology for IKNs (Schutte 2010).
- **Views**. The views perspective describes the abstraction of architectural characteristics of IKNs, i.e. the main set of engineering elements and properties that can be assembled to capture and represent IKNs. As discussed in section 1.6, the engineering focus of this study limits this perspective to an endogenous view on IKNs and these architectural views therefore only describe the internal components of IKNs and their relationships.

In order to develop the artefact behaviour specification in this chapter, meta-relationships and IKN life cycle axes are discussed in more detail in sections 5.6.3 and 5.6.4 respectively. The views axis is discussed in greater detail when an architecture framework is selected to populate it in chapter 6.

5.6.3 Meta-relationships

This section presents and populates the meta-relationships axis of the modelling framework.

The first dimension of the modelling framework describes the reference architecture for IKNs in terms of its meta-relationships with other architectural artefacts that exist in the solution space (refer to Figure 5.7).

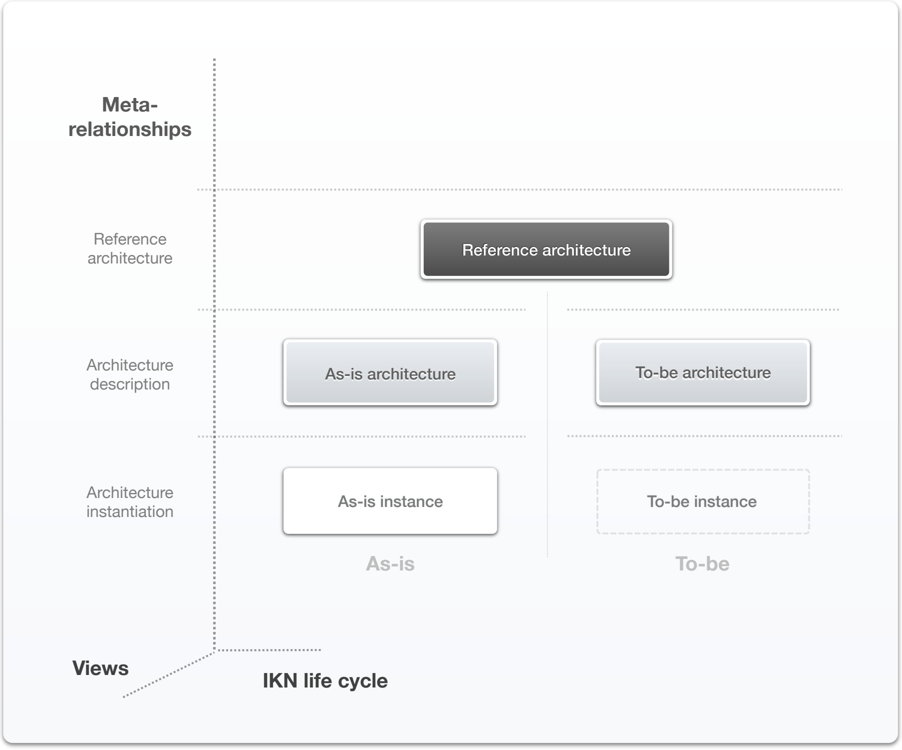


Figure 5.7: Modelling framework - Meta-relationships perspective

These architectural artefacts represent different levels of abstraction and include the following values along the axis:

- **Architecture instantiation.** This artefact is a real, instantiated version of a system in the form of an operational enterprise, i.e. a real-world IKN. The network engineering process is based on the design and instantiation of as-is and to-be versions of a network, and it is therefore possible to identify both as-is and to-be versions of IKN instantiations. It should, however, be noted that the to-be instance should be seen as an instantiation target rather than an instantiation in itself, as it never really exists. This is due to the fact that as soon as a to-be architectural instantiation exists, it is an as-is architectural instantiation. It is therefore inherently linked to a to-be architecture description, but is nevertheless a relevant artefact in the context of a reference architecture for IKNs as shown in section 5.6.4 below.

- **Architecture description.** An architecture description is an abstract description or representation of a specific system or instantiation, i.e. a particular model that indicates the system structure and the functions of its components, their interactions and constraints, and can be used to instantiate the system (Camarinha-Matos & Afsarmanesh 2008a). There is therefore a descriptive meta-relationship between this value on the axis and the architecture instantiation value. Particular architecture descriptions of both the as-is and to-be network instantiations are developed as part of the network engineering process, with migration between these architectures, along with their linked instantiations, representing organisational growth and change.

- **Reference architecture.** A reference architecture relates to an architecture description by serving as an input to structure the design of architecture descriptions in a given domain (refer to section 5.3.3). Reference architectures describe abstracted patterns that are common to the architecture descriptions of systems in the given domain and therefore there exists a descriptive meta-relationship between reference architectures and particular architecture descriptions. Reference architectures serve as the basis for the development of both as-is and to-be particular architecture descriptions and reference architectures are therefore relevant in both the as-is and to-be environments of the network engineering process, as indicated in Figure 5.7.

Traversing this axis of the modelling framework from instantiation to reference architecture would entail increasing levels of abstraction and replacement of constant configurations with variable configuration options. Conversely, moving from reference architecture to instantiation would entail increasingly assigning constant values to network configuration variables until a real-world network is instantiated where all values are constants.

5.6.4 IKN life cycle

This section presents and populates the IKN life cycle axis of the modelling framework.

The reference architecture for IKNs can also be described in terms of the life cycle of the organisations it is engineering. By adopting the life cycle of IKNs put forward by Schutte (2010) (refer to section 5.3.2), and with recognition to Back et al. (2004) and the ARCON framework contributed by Camarinha-Matos et al. (2008), the following values can be placed on the life cycle axis of the modelling framework (refer to Figure 5.8):

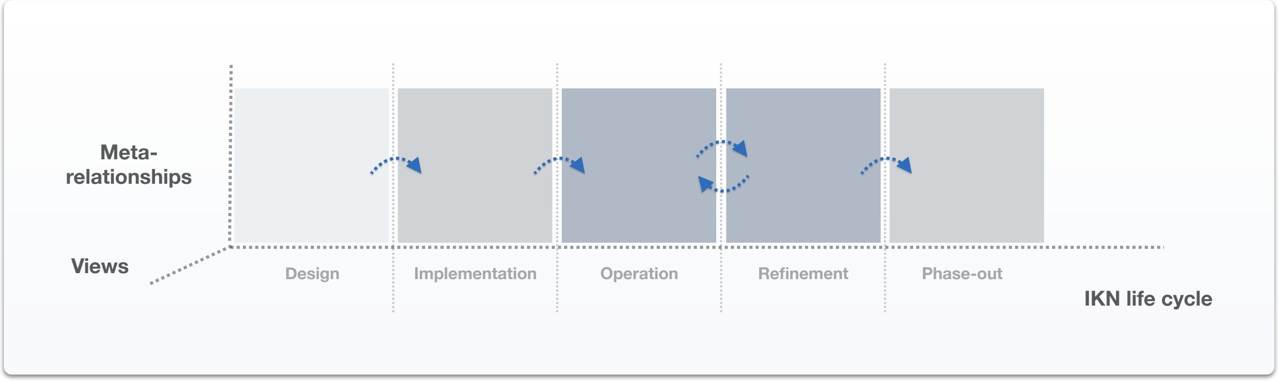


Figure 5.8: Modelling framework - IKN life cycle perspective

- **Design.** The network is designed at a conceptual level by determining its vision, strategy, domain and stakeholders. The network requirements are also established, followed by detail design and planning of the IKN.
- **Implementation.** The IKN design is implemented and includes planning and preparing, as well as executing the network implementation.
- **Operation.** The IKN is operated through creating and exchanging knowledge in order to facilitate innovation projects.
- **Refinement.** The IKN is refined by measuring performance, designing an improved version of the network and then migrating towards this new network design. This phase often occurs concurrently with the operation phase.
- **Phase-out.** Upon achieving its objectives or network failure, the IKN is phased out with attention to the management of its existing knowledge base.

5.6.5 Artefact behaviour specification

This section discusses the development of the artefact behaviour specification and investigates the relationships between the meta-relationships and IKN life cycle axes of the modelling framework.

Given the constituted meta-relationships and IKN life cycle perspectives included in the modelling framework, it is possible to plot the meta-relationships between architectural artefacts against the engineering actions in the life cycle of IKNs to arrive at an artefact behaviour specification for a reference architecture for IKNs. This is done through considering each phase of the IKN life cycle as a process wherein various architectural artefacts, including the reference architecture for IKNs, serve as inputs, are involved in transformations and also form the process outputs. The process-driven relationships between the architectural artefacts are then transposed onto the meta-relationship axis of the modelling framework to indicate how the artefacts interact in each of the life cycle phases. Specific emphasis is placed on all interactions that involve the reference architecture for IKNs, as a collation of these interactions forms the core of the artefact behaviour specification developed in this chapter.

5.6.5.1 Design phase

In the design phase of the IKN life cycle, an initial high-level design of the envisaged network architecture is done, followed by more detailed design and planning (refer to section 5.3.2.1). The design phase starts with no artefacts being present and concludes with detailed design artefacts that describe the network that is to be implemented.

In terms of architectural artefacts, the process therefore has an empty as-is architecture description and a lack of instantiation as an input and a designed to-be architecture description as an output, as illustrated in the process model in Figure 5.9a. The methodology for IKNs (Schutte 2010) guides the transformation

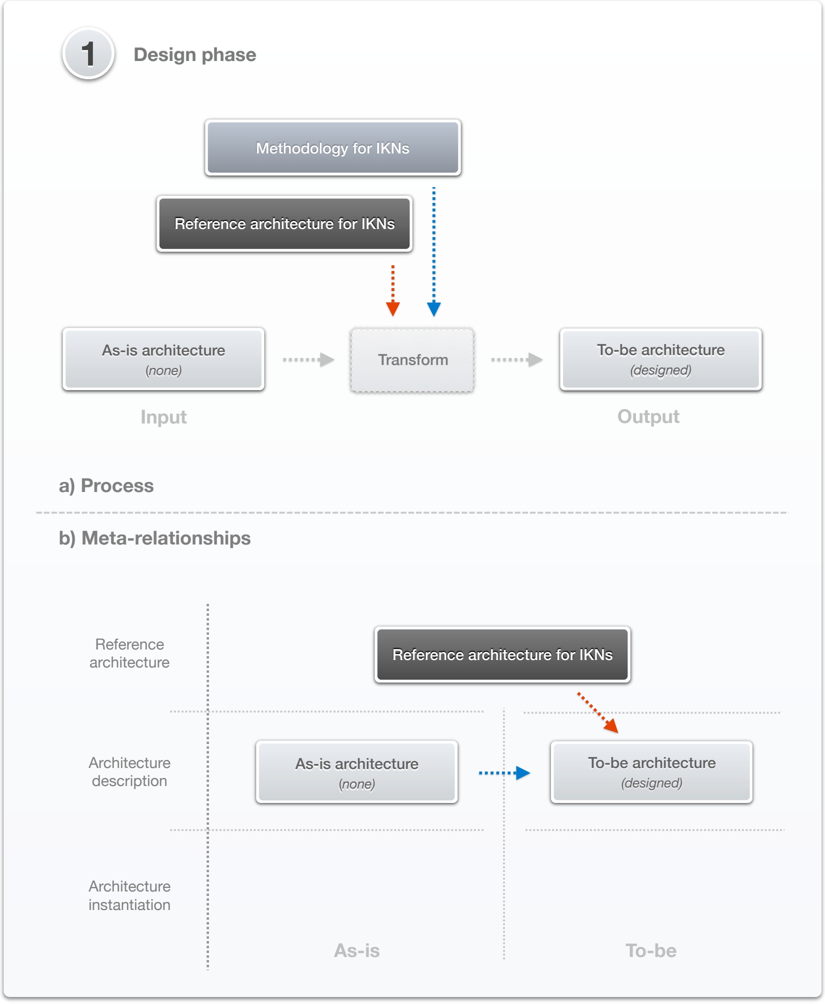


Figure 5.9: Behaviour of the reference architecture for IKNs in the design phase of the IKN life cycle

between these two artefacts by suggesting focus areas, planning documents and architecture descriptions that are to be developed, and this guidance is indicated with a blue arrow in Figure 5.9a.

In this phase of the IKN life cycle, the methodology makes specific reference to the design of the network structure, knowledge processes and organisational tools of the future IKN. Given the function of an enterprise reference architecture (refer to section 5.3.3), these are transformation activities in which the reference architecture for IKNs is required to supply reference models as the basis for the development of the particular models that comprise the to-be architecture description. The role of the reference architecture for IKNs in the transformation is indicated with a red arrow in Figure 5.9a.

These relationships between architectural artefacts are transposed onto the meta-relationships perspective of the modelling framework as shown in Figure 5.9b. The blue arrow indicates the migration from an empty as-is architecture to a designed to-be architecture via the methodology for IKNs, while the red arrow indicates the input the reference architecture for IKNs provides for the design of the to-be architecture description. No architectural instances are involved in the design phase of the IKN life cycle.

Figure 5.9b illustrates an element of the artefact behaviour specification for a reference architecture for IKNs by showing that the artefact is required to provide **inputs** for the **design** of a **to-be architecture description** when **no as-is architecture** description or **instantiation** is present.

5.6.5.2 Implementation phase

During the implementation phase of the IKN life cycle the network design developed in the design phase is implemented to constitute the IKN, i.e. migrating from a conceptual entity described in design documents to a tangible enterprise (refer to section 5.3.2.2).

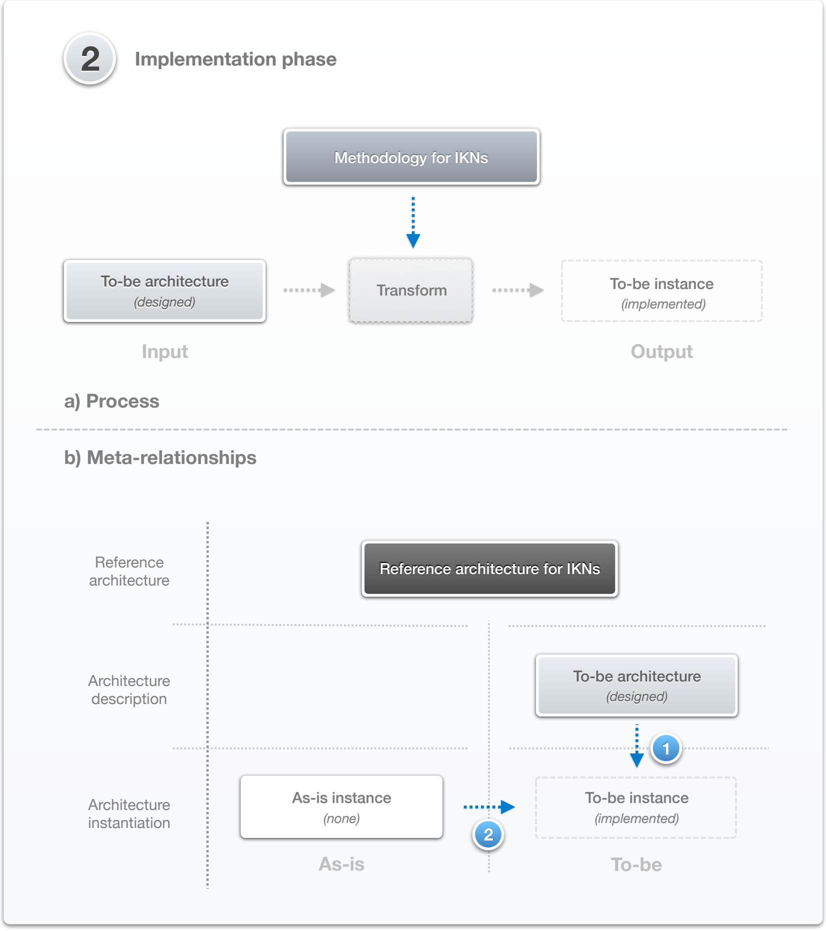


Figure 5.10: Behaviour of the reference architecture for IKNs in the implementation phase of the IKN life cycle

In architectural terms, this implementation process therefore has the designed to-be architecture description as an input and the implemented to-be architectural instance as an output, as illustrated in the process model in Figure 5.10a. The transformation in the process represents the instantiation of the to-be architecture description. However, as previously discussed, a to-be architectural instance cannot exist and is achieved by migrating the as-is instance (which was previously empty) to become the to-be instance. The methodology for IKNs informs the transformation between the to-be architecture description and the instance by providing guidance on planning and executing the network implementation. This is indicated by the blue arrow in Figure 5.10a.

Transposing the process model in Figure 5.10a to the meta-relationships perspective yields the diagram in Figure 5.10b. In this diagram, blue arrows indicate the instantiation of the to-be architecture description (1) and the migration of the previously empty as-is instance to the implemented to-be instance (2). These activities are, however, linked to the methodology for IKNs and the reference architecture is not involved in this phase of the life cycle. *The implementation phase of the life cycle of IKNs therefore does not contribute to the artefact behaviour specification.*

5.6.5.3 Operation phase

In the operation phase of the life cycle of IKNs, network stakeholders create and share knowledge in order to facilitate the execution of innovation projects and there are no meaningful structural alterations to the network (refer to section 5.3.2.3). The existing network design is therefore maintained for a period of time.

When considering this phase of the life cycle from an architectural point of view, the process can be modelled as shown in Figure 5.11a. The process has the operational as-is architecture description and instance as inputs, and the operational as-is architecture and to-be instance as outputs. This demonstrates that the same architecture description is maintained while transforming between architectural instances through time. Once again, the operational to-be instance is reached through migrating the as-is instance to achieve the identified to-be state. The methodology for IKNs guides the transformation in the process by identifying focus areas and placing emphasis on the maintenance of the network architecture. This is indicated by the blue arrow in Figure 5.11a.

Note that it is also possible to organically arrive at an operational IKN without ever having invested in accurate architectural descriptions of the network design, i.e. without the guidance of the methodology or reference architecture for IKNs to develop architecture descriptions. In such cases, the as-is architecture description will be absent from the process diagram in Figure 5.11a, with the operational as-is instance being the only input and the to-be operational instance the only output.

Figure 5.11b demonstrates this process model transposed to the meta-relationships perspective and blue arrows indicate the maintenance of the operational as-is architecture description (1) and the migration of the operational as-is instance to the operational to-be instance (2). These activities, however, are linked to the methodology for IKNs and the reference architecture is not involved in this phase of the life cycle. Therefore, similar to the implementation phase, *the operation phase of the life cycle of IKNs does not contribute to the artefact behaviour specification.*

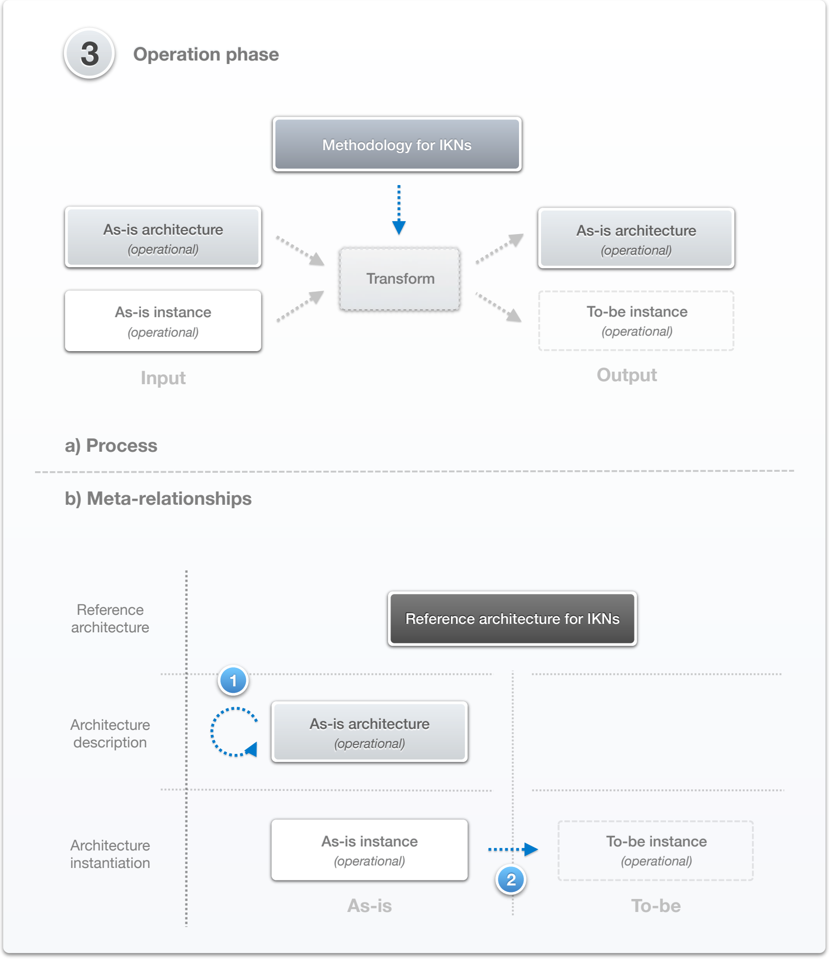


Figure 5.11: Behaviour of the reference architecture for IKNs in the operation phase of the IKN life cycle

5.6.5.4 Refinement phase

In the refinement phase of the life cycle of IKNs, which often occurs concurrently with the operation phase, the network is refined by measuring performance, designing an improved network and migrating towards this improved version (refer to section 5.3.2.4). This phase consequently assumes the existence of an operational IKN, with the possible existence of documentation and models that describe the network design.

In an architectural view on the process which is executed in the refinement phase, two possible points of entry exist. The first scenario consists of a network that is operational, i.e. has an operational as-is instance, but has no description of the operational as-is architecture. Such a description, however, is required to perform a viable health check on the network (refer to section 5.3.2.4), and needs to be developed before the design of a refined network is possible. Reference models are necessary to provide the basis for such a description. The input of the reference architecture for IKNs is therefore required for the first transformation from an operational as-is instance to an operational as-is architecture. This is indicated with a red arrow (1) in Figure 5.12a.

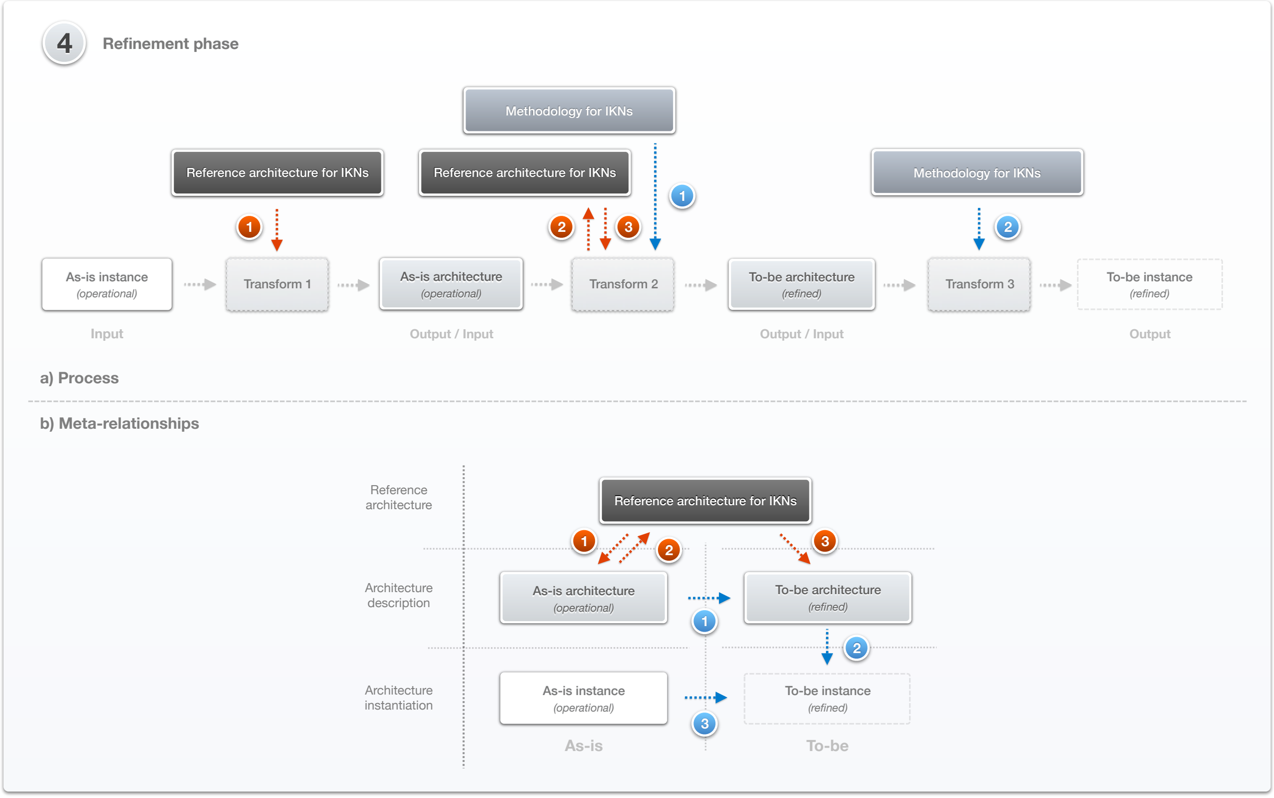


Figure 5.12: Behaviour of the reference architecture for IKNs in the refinement phase of the IKN life cycle

The second scenario for the refinement phase consists of an operational IKN that does have a description of the operational as-is architecture available. In this case the process starts with this as-is architecture as the input to the second transformation shown in Figure 5.12a.

The second transformation in the refinement process is driven by the methodology for IKNs’ guidance that the performance of the network should be measured, and this is indicated with a blue arrow (1) in Figure 5.12a. This transformation has the operational as-is architecture description as input and the refined to-be architecture description as output. Given the activities of measuring the performance of the network and designing a refined version, the reference architecture for IKNs is involved in the transformation in two ways. The first is in providing reference models that assist in assessing the operational as-is architecture, and this is indicated with a red arrow (2). Such an assessment results in the identification of certain problem areas in the network architecture that can now be used as targets for improvement.

The second role of the reference architecture for IKNs is to provide reference models as a basis for the design of the refined to-be architecture description that includes solutions to the previously identified problems, and this is also indicated with a red arrow (3) in Figure 5.12a. In this case, it would be expected that certain reference models that address the identified problem areas would be selected from within the reference architecture for IKNs as inputs for the development of specific architectural solutions.

Furthermore, the entire reference architecture for IKNs will facilitate the understanding of the interrelations between architectural elements and therefore assist in change management.

The final transformation in the refinement process involves the instantiation of the refined to-be architecture, and has the architecture description as input and the refined to-be instance of the network as output. This instantiation process is guided by the methodology for IKNs, and is indicated by a blue arrow (2) in Figure 5.12a.

In Figure 5.12b, the process diagram in Figure 5.12a is transposed to the meta-relationships perspective. The inputs that the reference architecture for IKNs provides via reference models for the development of an as-is architecture description are indicated with a red arrow (1). The same is done for the use of the reference architecture as a benchmarking tool to assess the as-is architecture (2), as well as for the inputs that the reference architecture provides for the development of a refined to-be architecture description (3). Following on this, the involvement of the methodology for IKNs is indicated with blue arrows as follows: guidance for the design of refined to-be architecture (1) and guidance for the instantiation of the refined to-be architecture (2) through the migration of the operational as-is instance to the refined to-be instance (3).

Figure 5.12b contributes three elements to the artefact behaviour specification of a reference architecture for IKNs by showing that the artefact is required to:

1. provide **inputs** for the **design** of an **operational as-is architecture** description when only an **operational as-is instance** is present.
2. assist in **assessing** the performance of an **operational as-is architecture** description by serving as a benchmarking tool.
3. provide **inputs** for the **design** of a **refined to-be architecture** description when a diagnosed, **operational as-is architecture** description is present.

5.6.5.5 Phase-out phase

In the final phase of the life cycle of an IKN the network is phased-out, as it has either failed or achieved its objectives (refer to section 5.3.2.5). Special attention is given to the management and archiving of the network's built-up knowledge base. This phase therefore starts with an operational network and concludes with an archived version of the enterprise from which value can still be extracted when required.

As was the case in the refinement phase of the IKN life cycle, an architectural view on the process that is executed during the network phase-out reveals two possible points of entry. The first scenario consists of a network that is operational, i.e. an operational as-is instance exists, but has no description of the operational as-is architecture. Such a description, however, is required to be able to assess the options for archiving the network and managing its knowledge base. It should be developed before the planning and execution of the network phase-out is possible. Reference models are required to provide the basis for

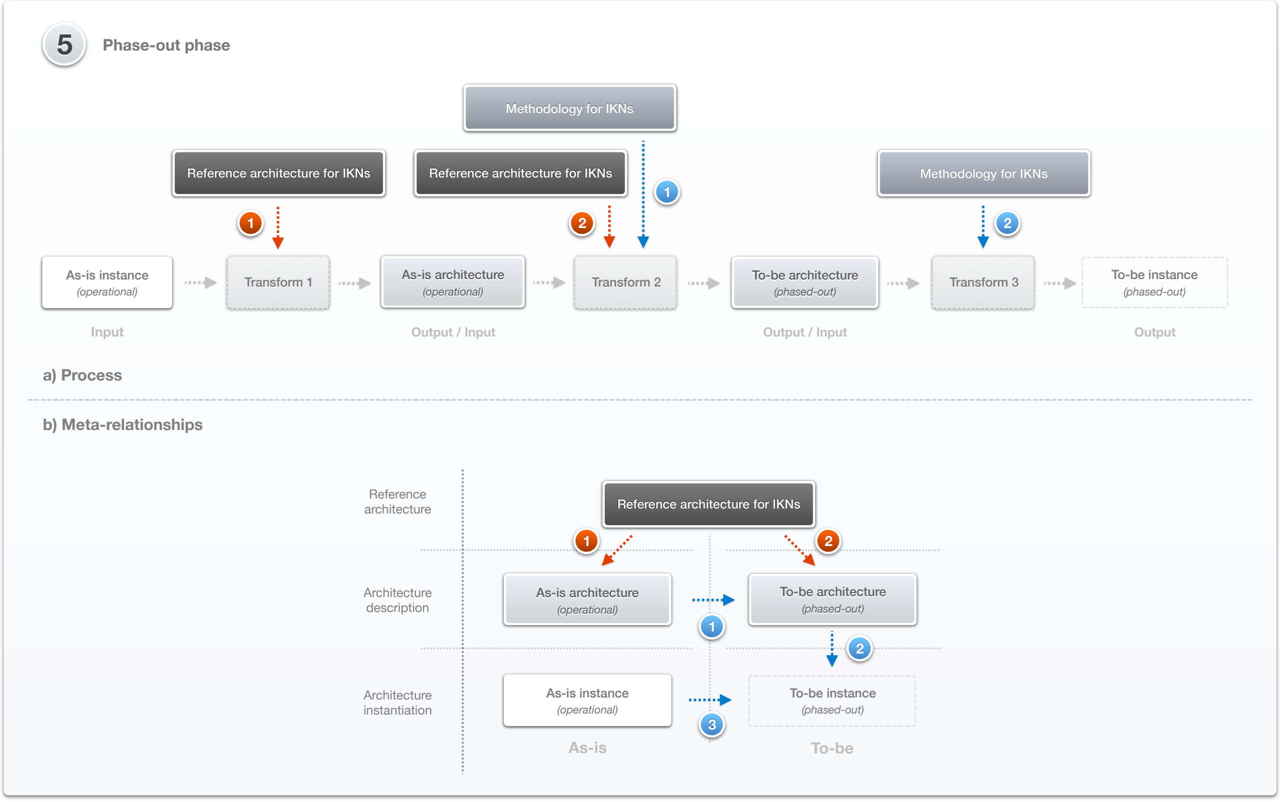


Figure 5.13: Behaviour of the reference architecture for IKNs in the phase-out phase of the IKN life cycle

such a description and the input of the reference architecture for IKNs is therefore required for the first transformation from an operational as-is instance to an operational as-is architecture. This is indicated with a red arrow (1) in Figure 5.13a. The second scenario for the network phase-out consists of an operational IKN that does have a description of the operational as-is architecture available. In this case the process starts with the as-is architecture description serving as the input to the second transformation shown in Figure 5.13a.

The second transformation in the phase-out process is driven by the guidance which the methodology for IKNs provides for approaching a network phase-out, and this is indicated with a blue arrow (1) in Figure 5.13a. This transformation has the operational as-is architecture description as input and the design of the phased-out to-be architecture description as output. Given this transformation, the reference architecture for IKNs is involved through the provision of reference models that form the basis of the phased out to-be architecture, as indicated by a red arrow (2) in Figure 5.13a. It is important to note that even though the network is phased out from its previous operational level of activity, the fact that a built-up knowledge base remains that should be managed and archived, means that it is not the case that the network is phased out to a point where no architecture exists. Some form of to-be architecture is therefore indeed necessary, albeit a minimal one.

The final transformation in the phase-out process involves the instantiation of the phased out to-be architecture, and has the architecture description as input and the phased out to-be instance of the network as output. This instantiation process is guided by the methodology for IKNs and is indicated by a blue arrow (2) in Figure 5.13a.

In Figure 5.13a the process diagram is transposed to the meta-relationships perspective in Figure 5.13b. The inputs that the reference architecture for IKNs provides via reference models for the development of an operational as-is architecture description are indicated with a red arrow (1). The same is done for the inputs that the reference architecture provides for the development of a phased out to-be architecture description (2). Blue arrows are used to indicate the involvement of the methodology for IKNs and represent the guidance for the design of phased out to-be architecture (1), along with guidance for the instantiation of this to-be architecture (2) through the migration of the operational as-is instance to the phased-out to-be instance (3).

Figure 5.13b contributes two elements to the artefact behaviour specification of a reference architecture for IKNs by showing that the artefact is required to provide **inputs** for the **design** of a **phased out to-be architecture** description when an **operational as-is architecture** description is present. It also repeats the requirement to provide **inputs** for the **design** of an **operational as-is architecture description** when only an **as-is instance** is present that was first identified in the refinement phase (refer to section 5.6.5.4).

5.6.5.6 Artefact users

This section discusses the expected users of the reference architecture for IKNs as an additional element of the relationships between the meta-relationships and IKN life cycle axes of the modelling framework.

The users of the reference architecture for IKNs are an important element of the environment with which the artefact will be interacting (Miedema et al. 2007). It is therefore important to consider who they are and what their respective agendas are in compiling the artefact behaviour specification.

Several stakeholders in the classic enterprise engineering process as proposed by Williams (1996) are presented in section 5.3.1. From descriptions of these stakeholders and the components of the artefact behaviour specification that have been identified thus far, it is deemed that the **enterprise engineering team** are the stakeholders in the engineering of IKNs that will interact with the reference architecture for IKNs. These are the individuals who perform the analysis, design and development work of the IKN architecture, as well as the planning of its rollout.

It should be noted that the reference architecture for IKNs is not being developed for any specific enterprise engineering team related to any specific IKN. In terms of the dimensions for describing an enterprise reference architecture in section 5.3.3, the artefact is therefore seen as a partial reference architecture (refer to Figure 5.3) as it is relevant to a specific enterprise class, but is not customised to any enterprise in particular.

Since no research exists on the exact composition of an enterprise engineering team for IKNs, a series of assumptions are made in order to anticipate the profile of the users of a reference architecture for IKNs.

The enterprise engineering team are primarily active in the design, refinement and phase-out phases of the IKN life cycle. For an IKN, this team likely includes representatives of multiple organisations that are initially included as founding members of the IKN. As soon as the development methodology for IKNs (Schutte 2010) leads to the identification of further stakeholders for inclusion in the network, representatives from these organisations are included as well, as it is crucial that all network stakeholders give their inputs in the design of the network architecture. This enforces the notion that care should be taken to ensure that the content of the reference architecture for IKNs is indeed agnostic of implementation, as its users represent a heterogeneous organisational background. The heterogeneous nature of this team also highlights the need for a common understanding of the network architecture amongst member organisations to allow coordinated implementation, operation, refinement and phase-out of the IKN in later phases of its life cycle.

Given the conceptual nature of the initial high-level planning of an inter-organisational network that aims to innovate through knowledge creation and transfer, the individuals that represent their respective network member organisations in this activity should be accustomed to thinking on a strategic level. The champion of the IKN network is also likely to be involved in the enterprise engineering team and will enforce the initial strategic focus. Furthermore, team members are likely to have experience with business architecture and be involved in top management or the business development function in their own organisations.

During the detailed design and subsequent refinement activities of the IKN life cycle, the team is also likely to include individuals who explicitly operate at an architectural level in their own organisations and in the network as a whole. Importantly, external consultants with knowledge regarding innovation network development or architectural issues may also form part of the enterprise engineering team at certain points during the network life cycle. These external consultants and architects are likely to have specialist knowledge regarding the architecting approach that is to be followed, including the reference architecture.

Stakeholders who are involved with the design of the network architecture at a technical level, e.g. the information systems architecture, may not necessarily be part of the enterprise engineering team that directly engage with the reference architecture for IKNs. They will, however, be required to align their efforts with the business architecture produced by the above-mentioned stakeholders.

The current discussion of the likely users of the reference architecture for IKNs contributes the following two elements to the artefact behaviour specification:

1. The artefact content must be **free from implementation bias** with regard to any specific network or network member organisation.
2. The artefact content must be aimed at stakeholders that are active in the **design**, **refinement** and **phase-out** phases of the IKN life cycle.

5.6.5.7 Summary of artefact behaviour specification

This section summarises the artefact behaviour specification as the product of the investigation into the relationships between the meta-relationships and IKN life cycle axes of the modelling framework.

By collating all the elements of the artefact behaviour specification for the reference architecture for IKNs that was identified by considering architectural transformations throughout the IKN life cycle and adding that to the artefact user identification, the complete specification is achieved. It is illustrated in Figure 5.14 and consists of five scenarios in which the reference architecture for IKNs is required to exhibit a desired behaviour when employed by its identified users. This behaviour is classified into two functional modes later in this section.

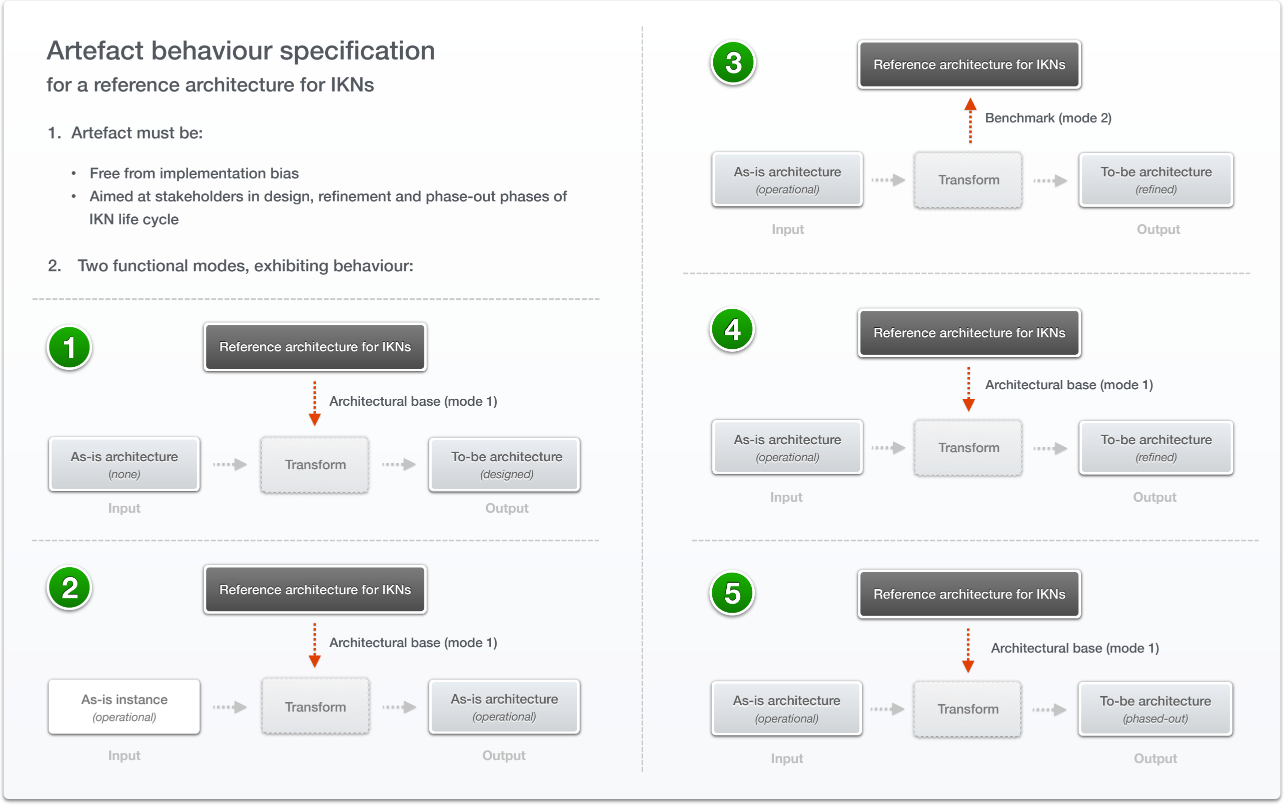


Figure 5.14: Artefact behaviour specification for a reference architecture for IKNs

1. The reference architecture for IKNs must be:

  1.1. *free from implementation bias*.
  1.2. aimed at stakeholders that are active in the *design*, *refinement* and *phase-out* phases of the IKN life cycle.

2. The artefact must be able to perform the following actions:

  2.1. Provide *inputs* for the *design* of a *to-be architecture* description when *no as-is architecture* description or *instantiation* is present.
  2.2. Provide *inputs* for the *design* of an *operational as-is architecture* description when only an *as-is instance* is present.
  2.3. Assist in *assessing* the performance of an *operational as-is architecture* description by functioning as a benchmarking tool.
  2.4. Provide *inputs* for the *design* of a *refined to-be architecture* description when a diagnosed, *operational as-is architecture* description is present.
  2.5. Provide *inputs* for the *design* of a *phased-out to-be architecture* description when an *operational as-is architecture* description is present.

Amongst these behavioural elements, two *functional modes* for the reference architecture can be observed. The first mode involves the *provision of inputs* for the design of particular architecture descriptions, and is required in four different scenarios (see 2.1, 2.2, 2.4 and 2.5 above) that include interactions with various architecture descriptions and instances. The second mode is to assist in assessing an existing architecture description during the refinement phase of the IKN life cycle by serving as a *benchmarking tool* (2.3 above). These multiple functional modes are achieved by including reference models that are agnostic of the artefact's functional mode in a certain scenario, while ensuring that neither mode is prohibited by their content. The functional modes of the artefact are therefore brought about through its interaction with its users in the specific scenario.

5.7 Conclusion

This chapter presented the first cycle in the incremental design and development process of a reference architecture for IKNs. The cycle therefore considered the compound requirement perspective that combines the scenarios and the functional requirement (refer to the green intersection in Figure 5.15).

The design cycle addressed the first solution objective (SO1), and the search process in this chapter was directed by the first secondary research question (SRQ1). This question asked what the desired behaviour of a reference architecture in the engineering of IKNs would entail.

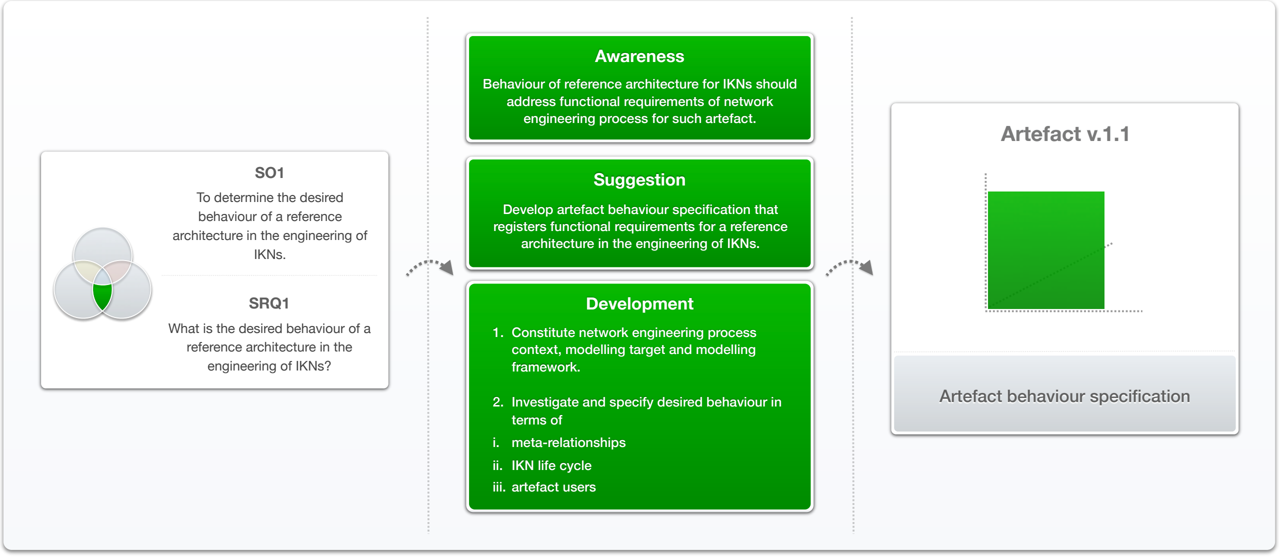


Figure 5.15: Conclusion of first design cycle

Following a concise presentation of relevant literature, the *awareness* of the problem outlined that the behaviour of a reference architecture for IKNs should address the functional requirements of the network engineering process for such an artefact. Subsequently, the *suggestion* was made that an artefact behaviour specification should be developed to register the functional requirements for a reference architecture in the engineering of IKNs.

The *development* of this artefact behaviour specification was initiated through the identification of the enterprise engineering process as a point of departure, specifying the typical IKN as the modelling target and constituting a modelling framework. The artefact behaviour specification was then populated through the investigation and specification of the desired behaviour of the reference architecture for IKNs in terms of meta-relationships between architectural artefacts throughout the life cycle of IKNs. The likely users of the artefact were also taken into account in the final artefact behaviour specification.

The artefact behaviour specification contributed by this chapter represents the *relationships between the meta-relationships and IKN life cycle dimensions of the modelling framework*, as illustrated by the green plane to the right of Figure 5.15. This artefact behaviour specification is seen as the first version of the reference architecture for IKNs (v.1.1). It answers SQ1 and achieves SO1 by determining the *desired behaviour of a reference architecture in the engineering of IKNs*, and the first design cycle is therefore circumscribed.

The generated knowledge about the final artefact, gained through design decisions and the development of a first version, can now be used to update the requirement frame of reference and leads to the initiation of a next design cycle. The artefact is further developed by means of a second design cycle in chapter 6 that selects an architecture framework to employ in the reference architecture for IKNs.

6. Architecture framework selection

6.1 Introduction

This chapter presents the second cycle in the incremental design and development process of the reference architecture for IKNs. The cycle considers the compound requirement perspective that combines the scenarios and the technical requirements (refer to the yellow intersection in Figure 6.1).

The design-and-development cycle aims at achieving the second solution objective (SO2) by stating that the artefact must employ an architecture framework that is suitable to the engineering of IKNs. The search process in this chapter is directed by the accompanying second secondary research question (SRQ2), which asks whether such an architecture framework is indeed available.

This chapter builds on the work done in chapter 5 and contributes the *selection of a suitable architecture framework* that constitutes and populates the views axis of the modelling framework for the reference architecture for IKNs. The search for an architecture framework that satisfies the requirements for the engineering of IKNs furthermore represents the investigation of the *relationships between the views axis and the axis indicating the engineering of IKNs throughout their life cycle*.

The architecture framework selection achieves the solution objective (SO2), and therefore circumscribes and restricts the activities of the design cycle, which leads to subsequent design cycles that further develop the artefact in chapters 7 and 8.

The solution objective and associated secondary research question that drives the design cycle is discussed in section 6.2. Section 6.3 presents background selected from existing literature that forms the basis for the ensuing discussion in this chapter. Design reasoning follows, with section 6.4 discussing the awareness that not all architecture frameworks are suitable to the engineering of IKNs, and that the requirements for suitability can be expressed in terms of architecture framework dimensions. Section 6.5 makes the suggestion to investigate and evaluate the Zachman framework for suitability to the engineering of IKNs, as expressed in terms of the architecture framework dimensions mentioned above.

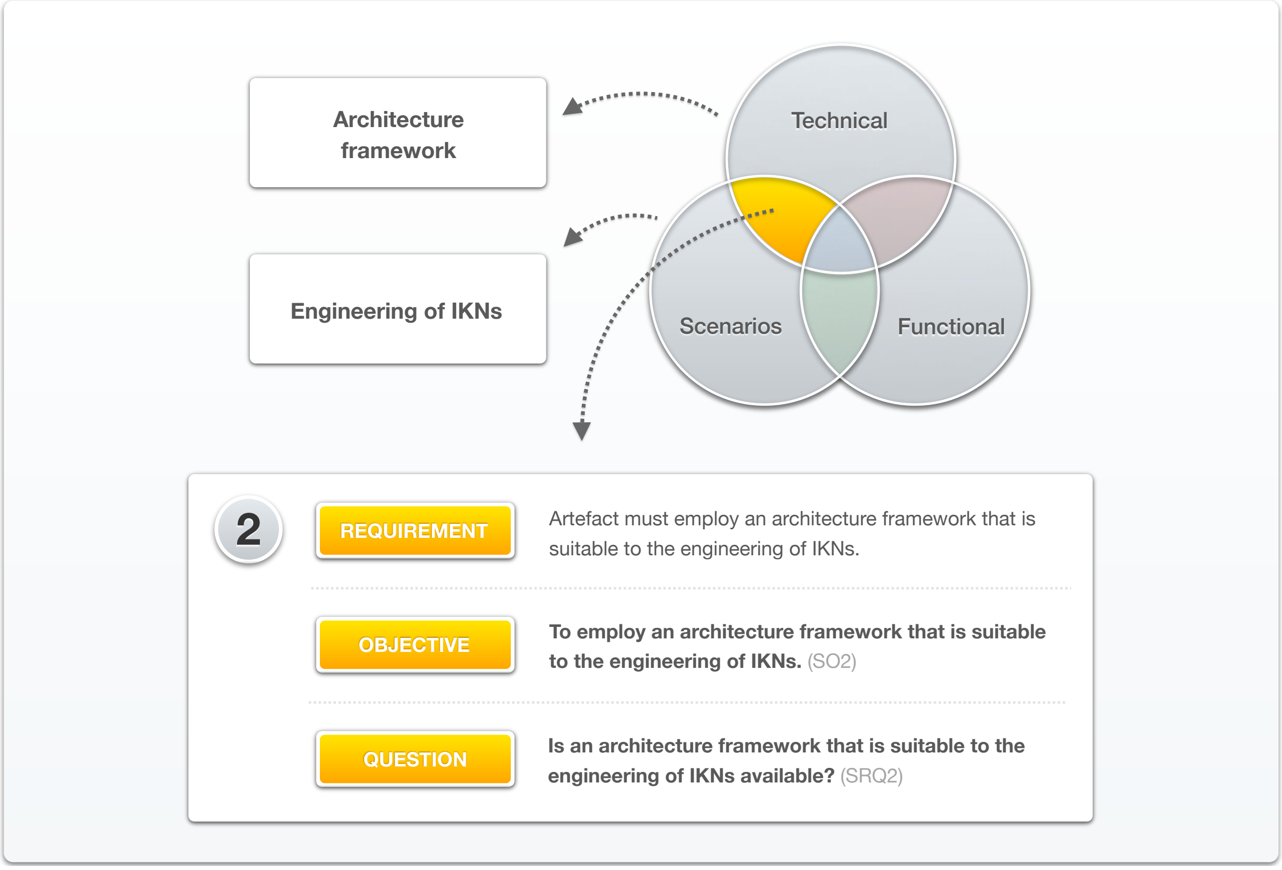
Section 6.6 documents the development of the selection of a suitable architecture framework. This development starts with the investigation of the Zachman framework in terms of the architecture framework dimensions used to describe the suitability to the engineering of IKNs. The Zachman framework's fit to these requirements is then evaluated, and either the Zachman framework is selected or the process is repeated with a different framework. Section 6.7 concludes this chapter.

6.2 Design cycle drivers

This section presents the solution objective and secondary research question that drives this design cycle. This is done in order to position the design cycle (and chapter) in the context of the overall design and development work performed in Part 3 of the research design.

The design cycle is driven by SO2 and SRQ2 as derived by considering the requirements intersection of scenarios and the technical requirement (refer to Figure 6.1).

Compound requirement:	The artefact must employ an architecture framework that is suitable to the engineering of IKNs.
Solution objective:	To employ an architecture framework that is suitable to the engineering of IKNs. (SO2)
Research question:	Is an architecture framework that is suitable to the engineering of IKNs available? (SRQ2)



Artefact must employ an architecture framework that is suitable to the engineering of IKNs.

To employ an architecture framework that is suitable to the engineering of IKNs. (SO2)

Is an architecture framework that is suitable to the engineering of IKNs available? (SRQ2)

Figure 6.1: Drivers of second design cycle

6.3 Background

This section presents literature that directly informs the awareness for the second (yellow) design cycle.

6.3.1 Technical perspectives on the engineering of IKNs

This section discusses elements of the scenarios that specifically pertain to the technical requirement for the artefact.

Enterprises, and therefore by extension also IKNs (refer to section 2.5), are intentionally created entities of human endeavour (Daft 2012; Dietz et al. 2013) and as such are examples of *organised complexities* (Weinberg 2001). This means that they are highly complex, while simultaneously exhibiting a high degree of organisation. These entities pose different challenges than ‘organised simplicities’ that can be dealt with through analytical approaches, or ‘unorganised complexities’ that can be addressed statistically (Weinberg 2001). Furthermore, modern enterprises are required to be in a continuous state of flux, being able to adapt to their surroundings. This requirement to accommodate change further compounds the complexity of the enterprise environment (Dietz et al. 2013). EE is proposed as an approach to make the organised complexity of changing enterprises *intellectually manageable* (Dietz et al. 2013) (refer to section 2.6), and in this study EE is applied to the domain of IKNs with the same objective.

The focus on the engineering of IKNs emphasises the point of view that they are purposefully designed, engineered and implemented *systems*. To this end, two distinct system notions were discussed in section 2.6, namely the teleological system notion and the ontological system notion. The teleological system notion emphasises the function and behaviour of a system and is adequately described with a black-box model (refer to section 2.6.2.1). A black-box model is effective when the aim is to use or control a system. However, when the aim is to build or change a system, i.e. to engineer it, as is the case in the scenarios that the reference architecture for IKNs are designed for, the ontological system notion has to be adopted (Dietz & Hoogervorst 2008). The ontological system notion emphasises the construction and operation of a system and is therefore prominent in the engineering sciences. It can be related to the functional focus of the teleological system notion by noting that the behaviour of a system is brought about as an effect of its construction and operation. Ontological systems are described with the use of white-box models, which indicate the components of a system and their interaction relationships (refer to section 2.6.2.2).

The ontological view of the enterprise described above should be seen as a systemic ontology (Bunge 1979), where the goal is to understand the *essence of the construction and operation of complex systems*, especially with regard to enterprises (Dietz & Hoogervorst 2008). The adoption of this view introduces transparency to the enterprise domain, and allows for a separate understanding of enterprise design and implementation.

Dietz et al. (2008) refer to the “collective services that an enterprise provides to its environment” as the business of the enterprise, and equate that to the functional perspective. Likewise, they refer to the “collective activities of an enterprise in which these services are brought about and delivered, including the human actors that perform these activities”, as the organisation of the enterprise. The constructional or ontological view of the enterprise describes this organisational element, and designing and achieving this view lies at the heart of the EE effort.

By considering the engineering of IKNs in the above context, two key elements of the scenarios, in which the reference architecture for IKNs will operate, can be identified:

- The ontological focus on the construction and operation of an IKN as a system in a way that is fully independent of its implementation, while exhibiting the essence of the system in a comprehensive, coherent, consistent and concise fashion.
- The pursuit of a unified and integrated design that enables effective network performance, while addressing mission and strategy related initiatives and areas of concern that are valid at some point in time.

By contributing these two elements to the domain, the engineering of IKNs allows individuals involved in the design and development of such networks to master the high level of organised complexity they are faced with.

6.3.2 Architecture frameworks

This section presents some background on the characteristics of architecture frameworks in order to enable the selection of a suitable framework to employ in the reference architecture for IKNs.

The *architecture* of a system indicates its high-level structure and describes the fundamental aspects of the system in an integrated fashion (Jonkers et al. 2006). The architecture depicts the fundamental organisation of a system embodied in its components and their relationships to each other from a specific perspective, and can therefore be used to guide the building or instantiation of the system (Schekkerman 2004). The architecture of a system is described in artefacts commonly referred to as architecture descriptions. These architecture descriptions need to be structured to ensure their comprehensiveness and coherence at the specific level of abstraction. *Architecture frameworks* provide guidance on how to structure and standardise architecture descriptions, and several such frameworks exist with varying levels of applicability to certain architectural environments (Greefhorst et al. 2006).

Architecture frameworks generally include a model for architecture descriptions, as well as a method to produce them, with some frameworks emphasising either the descriptions or the method (Greefhorst et al. 2006). Furthermore, architecture frameworks can be roughly categorised as either enterprise-class frameworks or application-class frameworks (Greefhorst et al. 2006). Enterprise-class frameworks are often referred to as *enterprise reference architectures* and focus on enterprise-level architecture descriptions that describe business units, entire organisations or even industry sectors (Schekkerman 2004; Greefhorst et al. 2006). Application-class frameworks are also referred to as software reference architectures and relate to application-level architecture descriptions that describe the architecture of a specific software application or a group of similar applications (Schekkerman 2004; Greefhorst et al. 2006).

6.3.2.1 Dimensions of architecture frameworks

This section introduces a range of objective architecture framework dimensions that are used in section 6.4 to articulate the requirements for suitability to the engineering of IKNs. These dimensions also structure the investigation of the Zachman framework in section 6.6.

Greefhorst et al. (2006) investigated architecture frameworks in an effort to discover their fundamental nature. It was found that architecture frameworks structure architecture descriptions along one or more axes, with this *n*-dimensional space denoting an architecture description that corresponds to the characteristics of the accompanying column and row (Greefhorst et al. 2006). The study refers to these axes as “dimensions” and to the columns or rows as “values”. Further analysis of the dimensions commonly used in architecture frameworks led to the identification of nine “base dimensions”, which form the foundation for the composite dimensions found in architecture frameworks (Greefhorst et al. 2006). The authors also note that “architecture frameworks represent attempts by their creators to enable clustering of architectural information in a way that suits a particular context and goal” (Greefhorst et al. 2006).

Greefhorst et al. (2006) defines an architectural dimension as follows:

> *An architectural dimension is a criterion to partition an architectural description into a set of segments, where each segment is identified by a unique value within a list of values associated with the dimension.*

The non-exhaustive list of the nine base dimensions of architecture frameworks are listed and described in Table 6.1. Illustrative values are supplied in parentheses, but are not standardised or formalised.

The nine base dimensions of architecture frameworks are now discussed in more detail, with reference to Greefhorst et al. (2006) throughout:

- **Type of information.** This is the most prevalent dimension in architecture frameworks and describes the subject of architecture information and therefore consists of the concepts that exist in domain-specific languages. Within this dimension a high-level distinction can be made between values such as *business*, *organisation* and *technical*. This dimension therefore breaks a complex situation down into more or less independent aspects through the separation of concerns.

Table 6.1: Base dimensions of architecture frameworks

(Greefhorst et al. 2006)

Dimension	Description
Type of information	The topic of the information. *(business, organisation, technical)*
Scope	The extent of the information covered. *(industry sector, organisation, domain, system family, system, component)*
Detail level	The amount of detail. *(high, medium, low)*
Stakeholder	The target audience. *(client, end-user, architect, analyst, developer)*
Transformation	The transformation phases that the architecture needs to cover. *(current situation, short-term, medium-term, long-term)*
Quality attribute	The quality attribute that is being addressed. *(functionality, reliability, usability, efficiency, maintainability, portability)*
Meta level	The amount of abstraction. *(instance, model, meta-model, meta-meta-model, meta-meta-meta-model)*
Nature	The nature of the information. *(policy, principle, guideline, model or standard)*
Representation	The way architectural information is represented. *(formal, semi-formal, informal)*

- **Scope.** This dimension describes the scope or extent of information covered and can be populated with the values *industry sector, organisation, organisational domain, system family* and *system component*. Different interpretations of the dimension are possible from specific points of view, with the general intention being that the dimension relates to ownership and levels of constraint.
- **Detail level.** This dimension is based on the amount of detail, allowing for levels with more information to be defined. A characteristic of this dimension is that between *low, medium* and *high* levels of detail all the information of the level above is retained, with new information added to it. The primary goal in this dimension is to leave out those details that are not relevant or known in a particular context or at a particular moment in time. As it is possible to add different types of detail, this dimension comes in various types, with other meanings regularly being attached in architecture frameworks.
- **Stakeholder.** This dimension focuses on the stakeholders that are addressed as a primary criterion, with different stakeholders typically only interested in certain parts of the architecture.

Illustrative values such as *client, end user, architect, analyst* and *developer* can be identified. The pure intention of this dimension is often obscured by other meanings attached to it in architecture frameworks.

- **Transformation.** This dimension generally uses change in time as the criterion, and typically distinguishes between the *current* situation and *short-term, medium-term* and *long-term* situations, including the transitions between them. It is, however, also possible to view this dimension not in terms of specific moments in time, but rather on the grounds of characteristics of the situation that can exist in time, e.g. maturity levels such as initial, repeatable, defined, managed and optimised.
- **Quality attribute.** A number of dimensions in existing frameworks mention quality characteristics, such as security, performance and usability. These characteristics can be considered as a separate dimension, with segments that highlight certain quality characteristics, e.g. *security, performance, usability, reliability, efficiency, maintainability* and *portability*.
- **Meta level.** This dimension addresses architecture frameworks that, instead of domain-specific models, provide general classifications and relationships, i.e. meta-models or generalised reference models. The meta-level dimension resembles the detail level dimension, the difference being that rather than describing less information, meta-models describe different information and that levels are separated through abstraction. Values such as *instance, model, meta-model, meta-meta-model* and *meta-meta-meta-model* can be identified.
- **Nature.** This dimension determines the nature of the architectural information, i.e. a *policy, guideline, model* or *standard*. Inherent to this dimension is the extent to which designers need to comply with the architectural information supplied by the framework, with e.g. a policy more important to follow than a guideline.
- **Representation.** This dimension uses the way architectural information is presented as criterion, with *informal, semi-formal* and *formal* values being illustratively identified. Informal representations use natural language which leaves room for interpretation. Semi-formal representations use some form of common syntax or format to improve the definition of architectural information, e.g. UML. Formal approaches employ description languages such as C2 and Rapide.

These base dimensions of architecture frameworks can be used in various ways, including as a communication vehicle to convey the intention of an architecture framework, or as a checklist to assist in the framework selection process. The dimensions can also be used in the construction of a new architecture framework or reference architecture by selecting the most applicable dimensions and values within those dimensions, and translating those to the required structure (Greefhorst et al. 2006).

This section presented technical perspectives on the engineering of IKNs, as well as background information on architecture frameworks. This discussion serves as context for the problem awareness that initiates design reasoning in the following section.

6.4 Awareness

This section presents the awareness step of the design reasoning in the second (yellow) design cycle.

Given SO2 and SQ2 (refer to section 6.2), the awareness of the problem that is considered in this design cycle can be stated as follows (refer to Figure 6.2):

> **Not all architecture frameworks are suitable to the engineering of IKNs. The requirements for suitability, however, can be expressed in terms of architecture framework dimensions.**

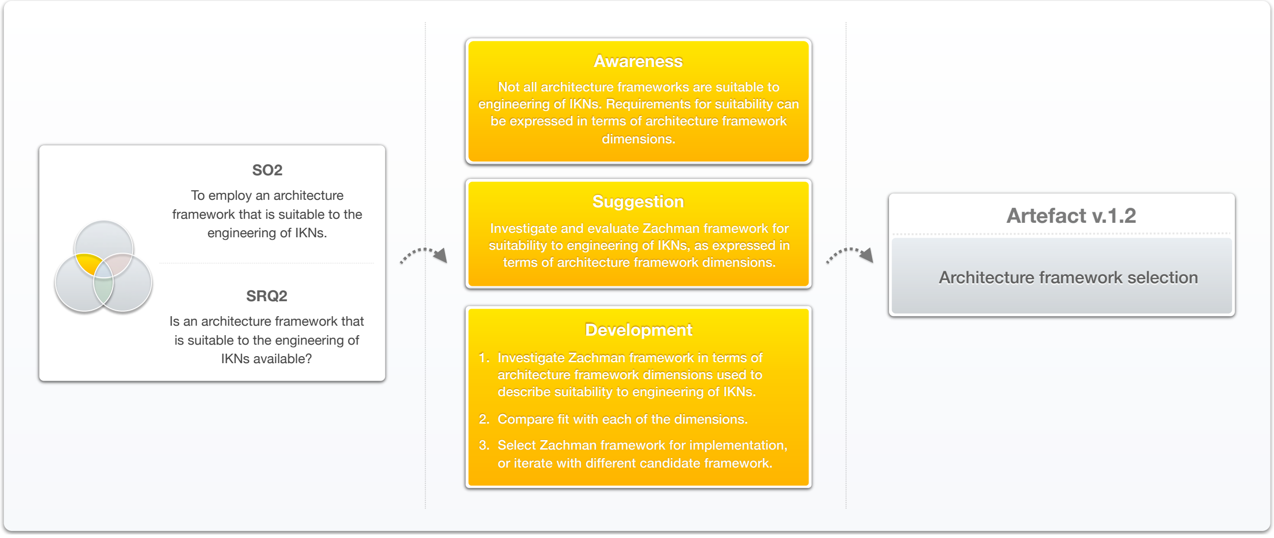


Figure 6.2: Reasoning in second design cycle

This problem awareness articulates that the problem that is considered in the study can in part be solved with an artefact that exhibits the appropriate characteristics when implemented in its expected environment, i.e. it employs an architecture framework that is suitable to the engineering of IKNs. The awareness relates to the observation made by Greefhorst et al. (2006) that “architecture frameworks are in essence attempts to enable clustering of architectural information in a way that is suited to a particular context and goal”. This supports the approach for selecting a suitable architecture framework for a given scenario proposed by Schekkerman (2004), which starts with carefully evaluating and understanding the enterprise business environment. Thereafter the goals and objectives that the framework will serve have to be identified and defined. Only then is it possible to check if an existing framework fits the enterprise business environment, i.e. goals and objectives, which in this case is the engineering of IKNs.

Reusing and adapting an existing framework to the specific scenario reduces the amount of effort that is required to structure the content of the reference architecture for IKNs. This is, however, only possible if the original effort by the creators of the framework to cluster architectural information aligns with the environment of engineering of IKNs.

The requirements for suitability of an architecture framework to the engineering of IKNs can be expressed in terms of the base dimensions of architecture frameworks (refer to section 6.3.2.1), and are presented in Table 6.2. Note that only the dimensions that are required to sufficiently express the requirements for suitability are employed. Some dimensions primarily relate to how the architecture framework can be adapted and implemented in the reference architecture for IKNs, and these dimensions are discussed in section 7.3.

From the discussion in section 6.3.1, it follows that the goal of the architecture framework which is selected for implementation in the reference architecture for IKNs, should enable the artefact to serve as a white-box model for the typical configuration of IKNs. The architecture framework should furthermore enable the reference architecture for IKNs to produce case-specific architecture descriptions that serve as white-box models for the specific IKNs involved. There is therefore a focus on the architecture framework's ability to provide an ontological view that describes the construction of IKNs.

In terms of **type of information** (refer to Table 6.2), the architecture framework and the architecture descriptions it produces must be able to describe architectural information that is of *business*, *organisational* and *technical* nature, as all of these elements are necessary to describe the construction of an IKN. Within the **scope** dimension, the framework needs to describe architectural information pertaining to the IKN as an *enterprise* and *network*, since the enterprise-wide construction of an IKN has to be described. The primary **quality attribute** which is mandatory for the framework is the focus on an *ontological view* on IKNs which is independent of implementation, resulting in *flexibility* of construction and *reusability* of parts.

The artefact behaviour specification that was developed in chapter 5 informs the architecture framework selection on the specification of **stakeholders** in that they are primarily users that engage with the business architecture of IKNs, such as *enterprise architects, strategists* and *concept owners*. The architecture framework should, however, also allow for secondary group of stakeholders that indirectly engage with the artefact through the alignment of their technical efforts with the business architecture of IKNs, such as *system architects, engineers* and *technicians*. The artefact behaviour specification furthermore shows that architecture framework has to handle both *current* and *future* architectural **transformations.** The **nature** of the architectural information that has to be structured is *models,* as these models are to combine to provide the white-box model that describes the construction of an IKN.

Table 6.2: Awareness of requirements for suitability of architecture framework to engineering of IKNs

Dimension	Description
Type of information	Business, organisation, technical
Scope	Enterprise, network
Quality attribute	Ontological view, flexibility, reusability
Stakeholders	Enterprise architects, strategists, concept owners, system architects, engineers, technicians
Transformations	Current, future
Nature	Models

6.5 Suggestion

This section presents the suggestion step of the design reasoning in the second (yellow) design cycle.

In light of the awareness of the problem considered in this design cycle, the following suggestion for a possible solution is made (refer to Figure 6.2):

> **Investigate and evaluate the Zachman framework for suitability to the engineering of IKNs, as previously described in terms of architecture framework dimensions.**

The Zachman framework (Zachman 1987) (refer to section 2.7.2) is an intuitive candidate for suitability to the engineering of IKNs, especially the 2013 interpretation of the framework as an enterprise ontology (Kappelman & Zachman 2013) which matches the quality attributes required by Table 6.2. The Zachman framework has also previously been used successfully to describe inter-organisational innovation networks in order to enable comparisons between their various constructional characteristics (Gous et al. 2011).

In order to ensure that the Zachman framework can enable the reference architecture for IKNs to describe the construction of these networks, the framework should be investigated in terms of each of the architecture framework dimensions used to express the requirements for suitability to the engineering of IKNs. As the Zachman framework is investigated in terms of each dimension, it should be compared with the requirements for suitability and the fit should be evaluated. If the Zachman framework complies with the requirements, it should be selected as an architecture framework for implementation in the artefact. If no compliance is found, the process should be repeated with a different candidate architecture framework.

6.6 Development

This section presents the development step of the design reasoning in the second (yellow) design cycle.

6.6.1 Type of information

The Zachman framework describes architectural information with a number of different subjects along its vertical axis, as it passes through the stages of reification from idea to physical reality. Zachman argues that, in passing through the stages of reification, all industrial products (and by extension also enterprises, being intentional entities of human endeavour) have certain following architectural perspectives (Kappelman & Zachman 2013; Zachman 2011; Zachman 1997). These perspectives are represented by the following rows in the framework (refer to Figure 6.3):

- Scope (boundaries)
- Requirements (concepts)
- Design (logic)
- Plan (physics)
- Part (configurations)
- Product (instantiations)

The manifestations of these architectural perspectives determine the types of information described in the Zachman framework, namely scope contexts, business concepts, system logic, technology physics, tool components and operations instances, as indicated by the row labels to the far right in Figure 6.3.

In the top row of the Zachman framework, *scope contexts* identify the architectural items that form part of the architectural description at a strategic level, thereby identifying the boundaries of the architecture. The definition of *business concepts*, including the relations between them, in the second row defines what certain architectural terms mean in the context of the specific architecture which is described. These business concepts form the requirements for the supporting lower architectural levels. In the third row, the *design logic* of the enterprise is represented, thereby focusing the description of the enterprise architecture on the system level and indicating its internal organisation. The specification of *technology physics* in the fourth row describes the architecture on a technical level by specifying the technological plan which will be configured to instantiate the enterprise. In the fifth row, the *tool components* that form the constituent parts of the technological specification are configured to produce the *operational instance* of the enterprise in the sixth row.

From the discussion above, it follows that the top two rows of the Zachman framework, with their focus on *scope contexts* and *business concepts* respectively, address architectural information of a *business* nature. The third row addresses *organisational* information by representing the system logic, with the fourth and fifth rows describing *technical* architectural information through their respective focus on technology physics and tool components. The Zachman framework therefore **satisfies** the requirements for suitability to the engineering of IKNs in terms of the **types of information** it structures and describes (refer to Table 6.2).

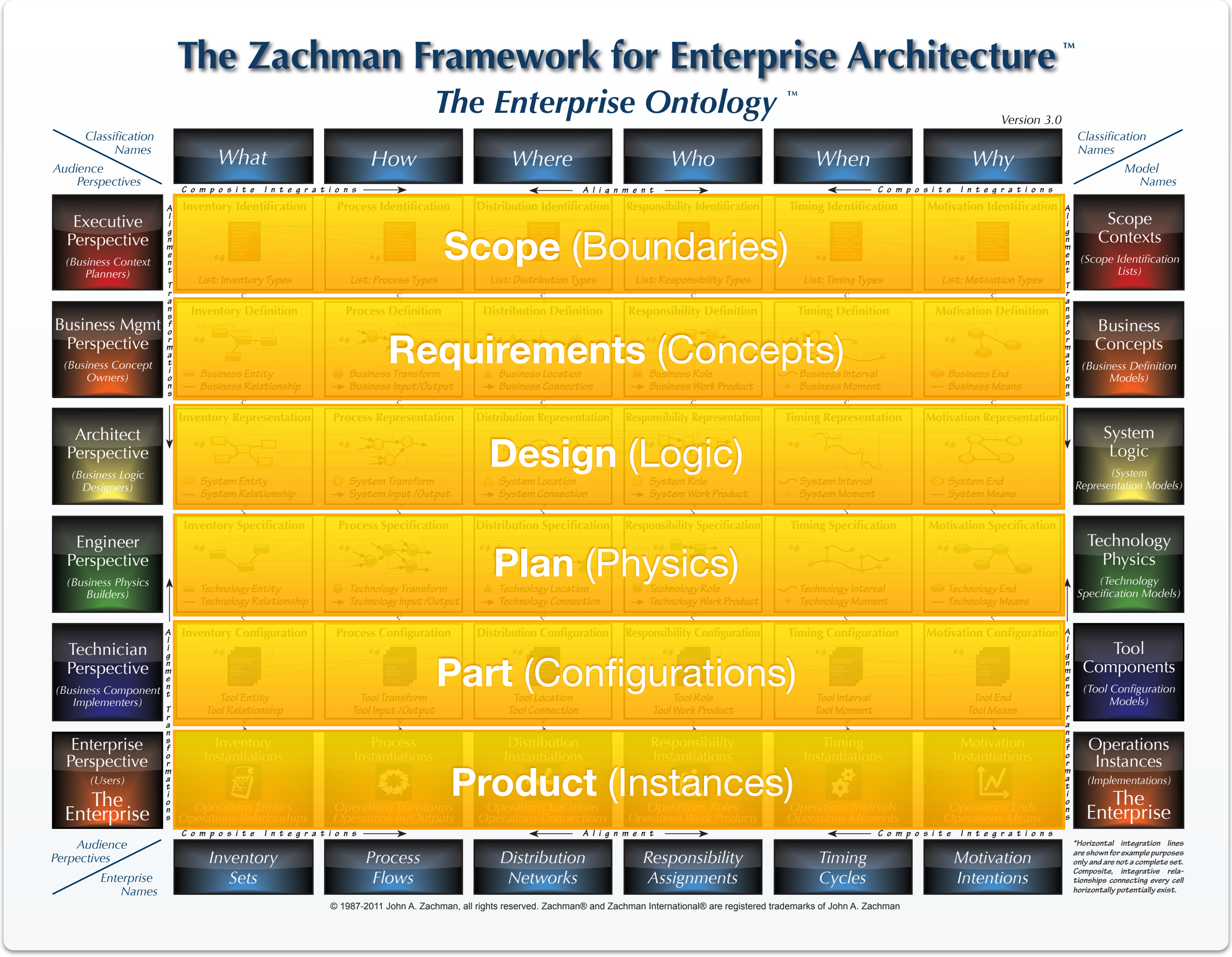


Figure 6.3: Types of information described by the Zachman framework

(adapted from Kappelman & Zachman 2013)

6.6.2 Scope

The scope of information covered by the Zachman framework is variable, since the underlying logic of the framework can be applied to any object (Kappelman & Zachman 2013; Noran 2003). This means that the framework can be scaled to fit any of the values along the scope dimension of architecture frameworks, including industry sector, organisation, organisational domain, system family and system component. The precise scope of an implementation of the Zachman framework is determined by the architectural boundaries identified in the first row. For this reason there is a link between the business information the framework describes and the scope of the rest of the architecture (refer to section 6.6.1). The columns of the framework are labelled as “abstractions” that combine to provide the complete set of relevant descriptive characteristics of the object, thus providing evidence for their comprehensiveness (Kappelman & Zachman 2013) (refer to Figure 6.4). These abstractions are universal and are common to all industrial products (Kappelman & Zachman 2013), including all known enterprise forms.

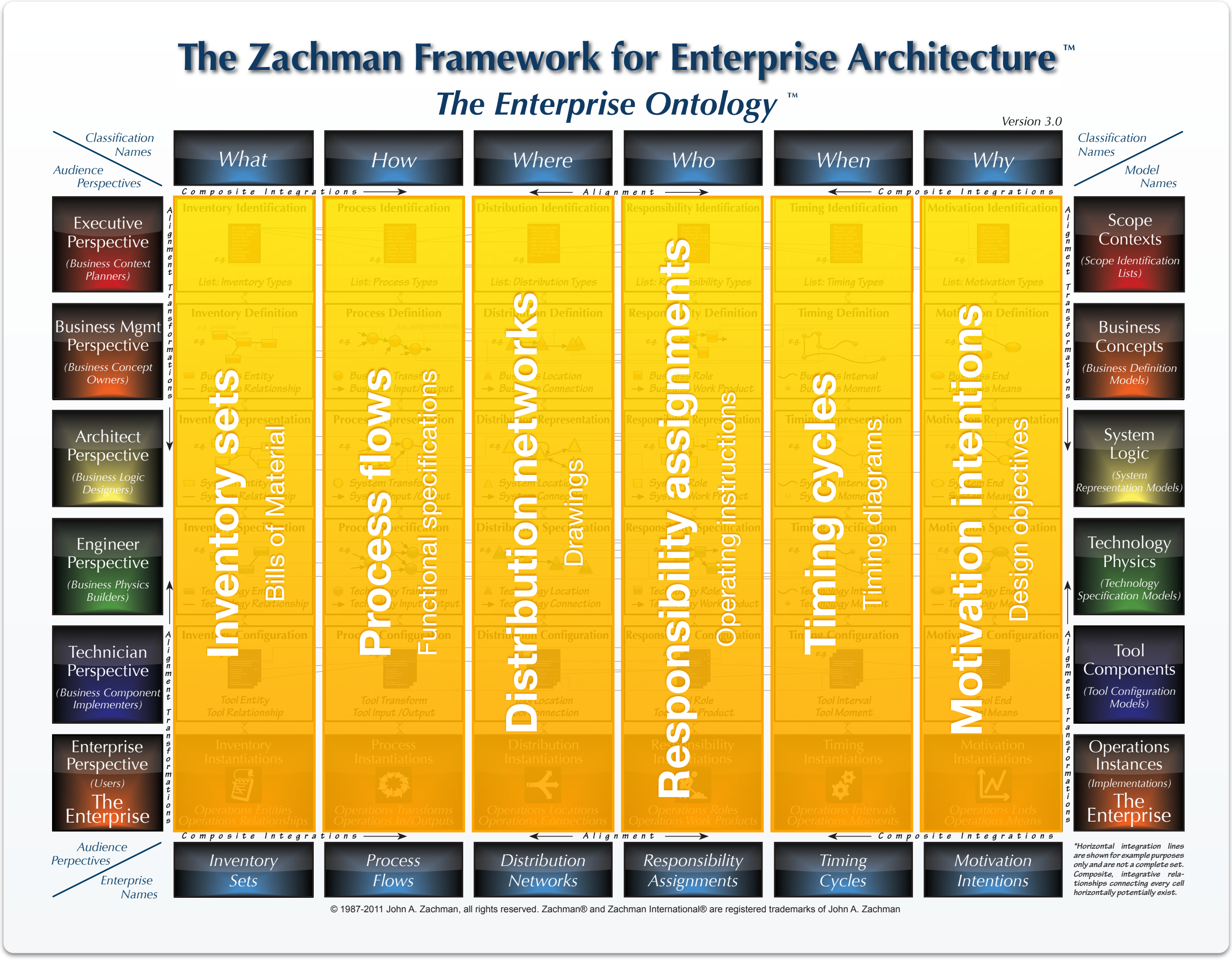


Figure 6.4: Scope of the Zachman framework

(adapted from Kappelman & Zachman 2013)

The abstraction columns answer the six basic interrogative questions: “What?”, “How?”, “Where?”, “Who?”, “When?” and “Why?”. The columns correspond to “the universal set of descriptive representations for describing any and all complex industrial products” (Kappelman & Zachman 2013) and are, from left to right:

- Inventory sets – described in bills of material
- Process flows – described in functional specifications,
- Distribution networks – described in drawings
- Responsibility assignments – described in operating instructions
- Timing cycles – described in timing diagrams
- Motivation intentions – described in design objectives

These descriptions may have industry-specific variations in terminology, but combine to indicate that the Zachman framework has the required architectural scope to describe the construction of an *enterprise*. The framework therefore satisfies the first requirement for suitability to the engineering of IKNs in terms of scope.

Including the complete set of primitive (single-variable) abstractions in the artefact allows for the construction of composite (multi-variable) models. These composite models represent concepts in the way they would be encountered in the real world, i.e. as multi-variable compounds in the functioning *enterprise*. The relationships that exist between elements from different abstractions that allow for the construction of composite models are known as *integrations*.

The complete set of descriptions listed above along with its flexible scope and integrations enable the framework to describe the architecture of an enterprise consisting out of multiple, geographically dispersed (Where?), constituent organisations (Who?) that join forces for a specific reason (Why?). The framework also recognises that these organisations can collaborate to produce and transform (how) a range of objects (What?) according to a number of timing cycles (When?). This means that the Zachman framework also has the ability to describe the construction of a *network* as part of its architectural scope.

The Zachman framework therefore **satisfies** the requirements for suitability to the engineering of IKNs in terms of the **scope** of the architectural information it describes (refer to Table 6.2).

6.6.3 Quality attribute

The Zachman framework is a classification scheme or taxonomy for descriptive representations of objects, with the added context of enterprise-related labels for the descriptions (Kappelman & Zachman 2013). Kappelman and Zachman (2013) describe the framework as “an ontology about organisations; an information model of what you need to ‘know’ about an enterprise so you can better manage, monitor, change, understand and communicate about it”. The Zachman framework therefore describes the essence of an enterprise.

As presented in section 2.7.2, the Zachman framework is represented in two dimensions as a matrix consisting of six columns and six rows, with connections between the cells being indicated in some places (Zachman 2011). Structuring the architectural description of an enterprise according to this schema produces an architecture that is *normalised*, i.e. a description where no one fact is included in more than one cell. This distinguishes the Zachman framework from other architecture frameworks, which allow redundancies in their basic models and often at least combine process and data into their high-level meta-models (Kappelman & Zachman 2013). The Zachman framework describes architectural information in terms of primitive, single-variable models that form the fundamental building blocks of the enterprise and indicate its construction at the most essential level. In this context, the terms ‘primitive’ and ‘single-variable’

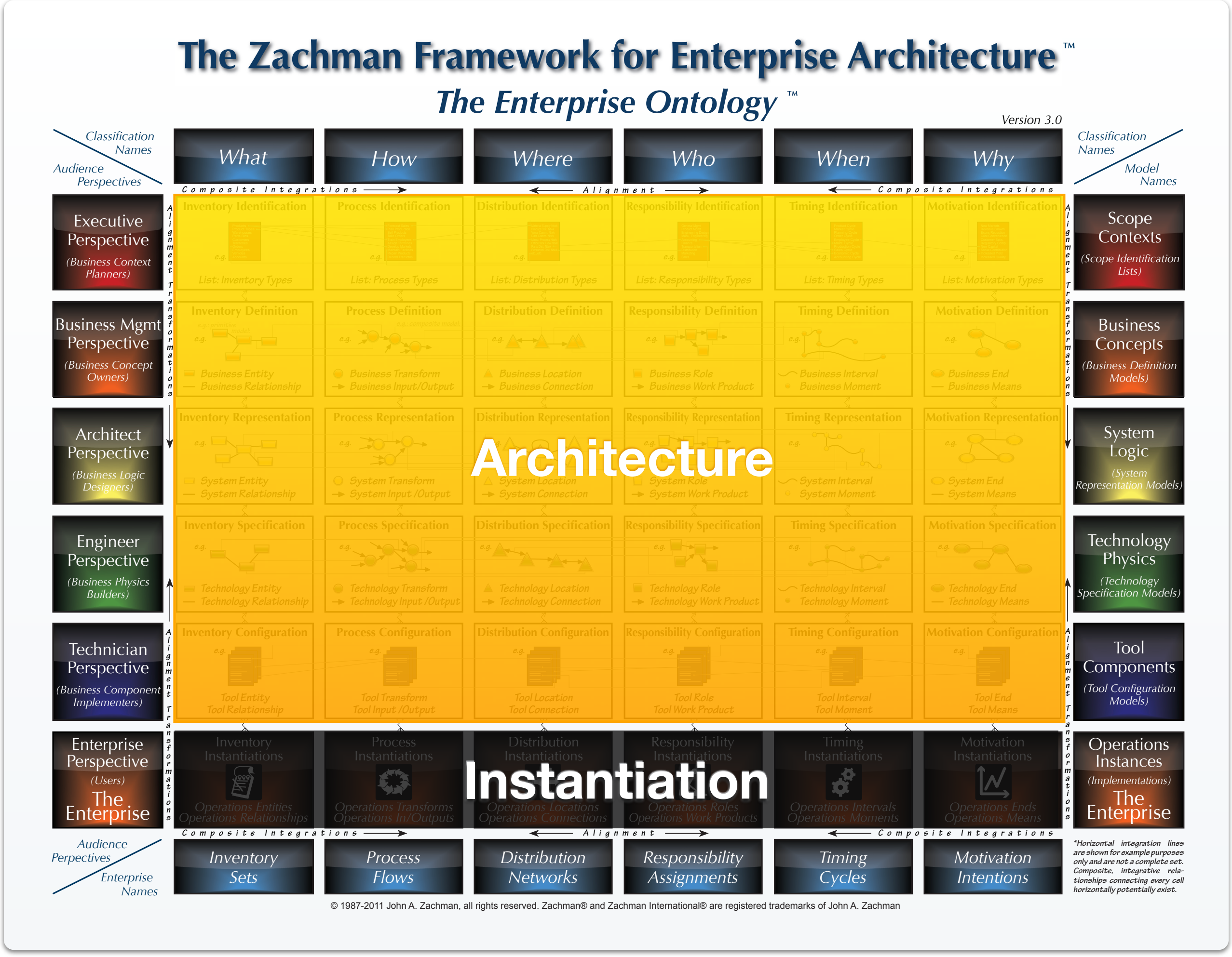


Figure 6.5: Ontological view of the enterprise according to the Zachman framework

(adapted from Kappelman & Zachman 2013)

are used to signify models that only address a single abstraction or interrogative, e.g. only considering the “What” question, regarding the modelling subject.

The primitive, single-variable models may be combined or integrated to form composite, multi-variable models that can be used to construct or describe the functional, real-world version of the enterprise. The Zachman framework therefore creates an architecture that shows an understanding of the construction and operation of the enterprise in a way that is fully independent of its implementation or instantiation (refer to Figure 6.5).

This amounts to an *ontological view* of the enterprise that only shows the essential features of the system (Dietz & Hoogervorst 2008). Furthermore, since infinite options exist for the integration of primitive models to form composite models, much like the way a finite number of elements can combine to form an infinite number of compounds (Kappelman & Zachman 2013), the Zachman framework produces an architecture that has great *flexibility* as a quality characteristic. The primitive models may also be reused in multiple compound configurations, allowing for *reusability* of architectural parts.

The Zachman framework therefore **satisfies** the requirements for suitability to the engineering of IKNs in terms of the **quality attribute** of the architectural information it describes (refer to Table 6.2).

6.6.4 Stakeholders

The Zachman framework explicitly links different stakeholders to the various perspectives or rows. The different stakeholders form the primary audiences of the architectural information presented in these respective rows. These stakeholders are from the top row to the bottom row (Zachman 2011) (refer to Figure 6.3):

- Executives, as business context planners;
- Business management, as business concept owners;
- Architects, as business logic designers;
- Engineers, as business physics builders;
- Technicians, as business component implementers;
- Operators, as business users.

These stakeholders are also seen as the owners of the architectural information in the given perspective and are therefore positioned the best to develop the models that populate the particular row of the framework. Executives have a *strategic* focus that helps to identify the business context and boundaries, with business management being the *owners* of business *concepts* and are positioned the best to define these concepts. These two groups are seen as the stakeholders that directly engage with the business architecture of the enterprise by actively participating in its design and engineering.

Architects, in turn, operate as logic designers that help to represent the enterprise as a *system*. These system architects should, however, not be confused with enterprise architects that also consider the business layer of the enterprise from an architectural point of view. *Engineers* build the physics of the enterprise by specifying the technology that is required to support the business concepts and system logic. *Technicians* implement the components of the enterprise by configuring specific tools as specified by engineers. These three groups are seen as the stakeholders that indirectly engage with the business architecture of the enterprise by aligning their technical efforts, e.g. information systems, with it to ensure that the business architecture is accurately supported.

The framework itself is a classification schema for enterprise architecture artefacts and therefore speaks to *enterprise architects* that are attempting to architect, describe and integrate the entire enterprise.

The Zachman framework therefore **satisfies** the requirements for suitability to the engineering of IKNs in terms of the **stakeholders** involved in the architectural information it describes (refer to Table 6.2).

6.6.5 Transformations

As was discussed in section 6.6.3, the Zachman framework is inherently an ontology that describes the essence of the enterprise. As such, the framework itself is entirely process and method agnostic and implies no temporal or characteristic-based preference in terms of transformations (Kappelman & Zachman 2013).

The objective of the framework is to serve as a schema for structuring architectural information and since processes or methods do not bound the nature of this architectural information, it is possible to use the framework to structure multiple architecture transformations. In other words, it is possible to use the Zachman framework to describe both the *current* architecture of the enterprise, as well as *future* versions of the architecture, be it of a short, medium or long-term nature. It is then possible to migrate the enterprise from a current architecture to a future architecture through enterprise engineering.

The Zachman framework therefore **satisfies** the requirements for suitability to the engineering of IKNs in terms of **transformations** of the architectural information it describes (refer to Table 6.2).

6.6.6 Nature

With regards to the nature of the architectural information it includes, the Zachman framework specifies that it is *models* that are stored in each of its matrix cells. The nature of these models do however differ between perspectives. They are from row one to row five (Zachman 2011):

- Scope identification lists
- Business definition models
- System representation models
- Technology specification models
- Tool configuration models

The framework does not specify which models to use or how to use them, allowing for flexibility in their application and interpretation. A meta-model, however, is provided for each cell. These meta-models provide a representational language that ensures coherency between the architecture information contained in the framework, both within a row and throughout a column (Kappelman & Zachman 2013). It is also noteworthy that the models comprising the framework structure are normalised, meaning that each cell contains a primitive, single-variable model. These models are therefore ideally suited to describe the construction of the enterprise, independent of its implementation (refer to section 6.6.3), and together form a white-box model of the enterprise.

The Zachman framework therefore **satisfies** the requirements for suitability to the engineering of IKNs in terms of the **nature** of the architectural information it describes (refer to Table 6.2).

6.6.7 Architecture framework selection and modelling framework

The investigation and evaluation of the Zachman framework in terms of the dimensions that were used to express the requirements for suitability to the engineering of IKNs (refer to Table 6.2) have shown that the framework meets all the requirements. The **Zachman framework** is therefore selected as the architecture framework to be employed by the reference architecture for IKNs.

A modelling framework was created as the foundations of the reference architecture for IKNs in chapter 5 (refer to section 5.6.2). This modelling framework consists of three axes that constitute a three-dimensional solution space wherein the artefact is developed (refer to Figure 6.6). The three axes describe meta-relationships between the artefact and its architectural environment, the role it plays in the engineering of IKNs throughout their life cycle, and the architectural views the artefact produces. The meta-relationships and IKN life cycle axes, as well as the interactions between them, were discussed in chapter 5.

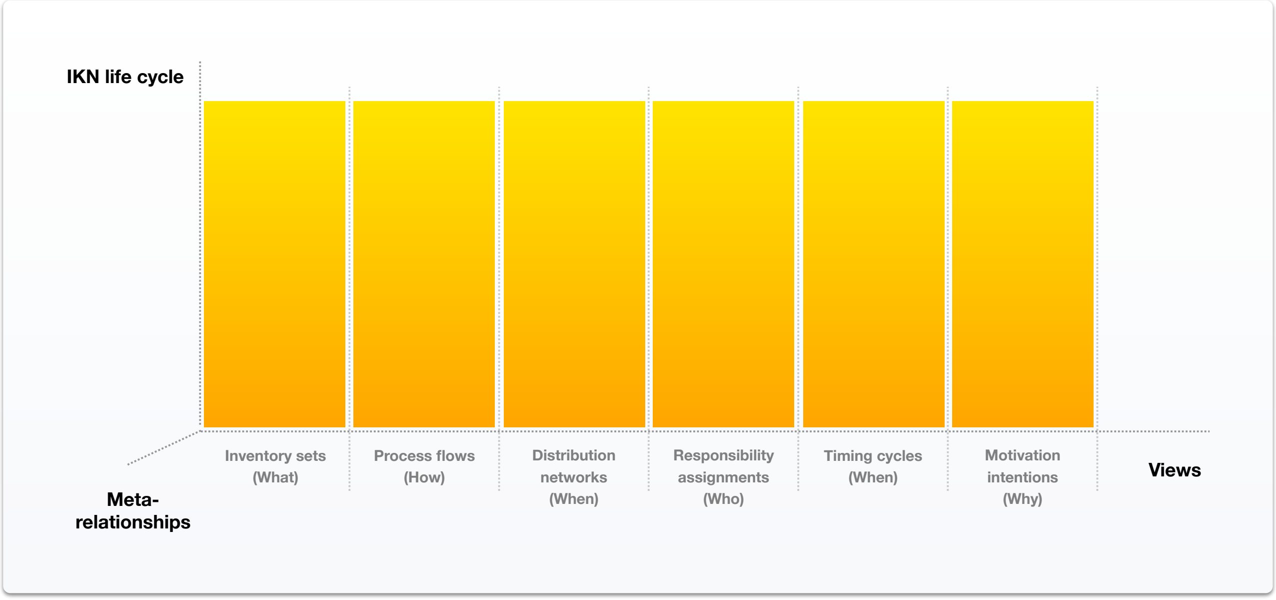


Figure 6.6: Modelling framework - Architectural views perspective

The views perspective describes the abstraction of architectural characteristics of IKNs, i.e. the main set of engineering elements and properties that can be assembled to capture and represent IKNs. The focus of this study on the creation of a white-box model that describes the construction of IKNs limits this perspective to an endogenous view on IKNs and these architectural views therefore only describe the internal components of IKNs and their relationships.

Given the selection of the Zachman framework as an architecture framework that is suitable to IKNs, it is now possible to populate this third axis of the modelling framework with the architectural views provided by the Zachman framework. These views correspond to the columns of the Zachman framework as they represent the framework's version of the abstraction of architectural characteristics (Kappelman & Zachman 2013). They are, in no particular order, as in the Zachman framework itself:

- Inventory sets (What?)
- Process flows (How?)
- Distribution networks (Where?)
- Responsibility assignments (Who?)
- Timing cycles (When?)
- Motivation intentions (Why?)

These abstractions combine to form a complete set of descriptive representations of the essence of IKNs in order to enable the engineering of these networks throughout their life cycle. As these architectural views are not linked to any specific process or method in the Zachman framework, they remain constant throughout the life cycle of IKNs. However, the particular manifestations of these views as produced through reification will differ, as addressed in the current chapter.

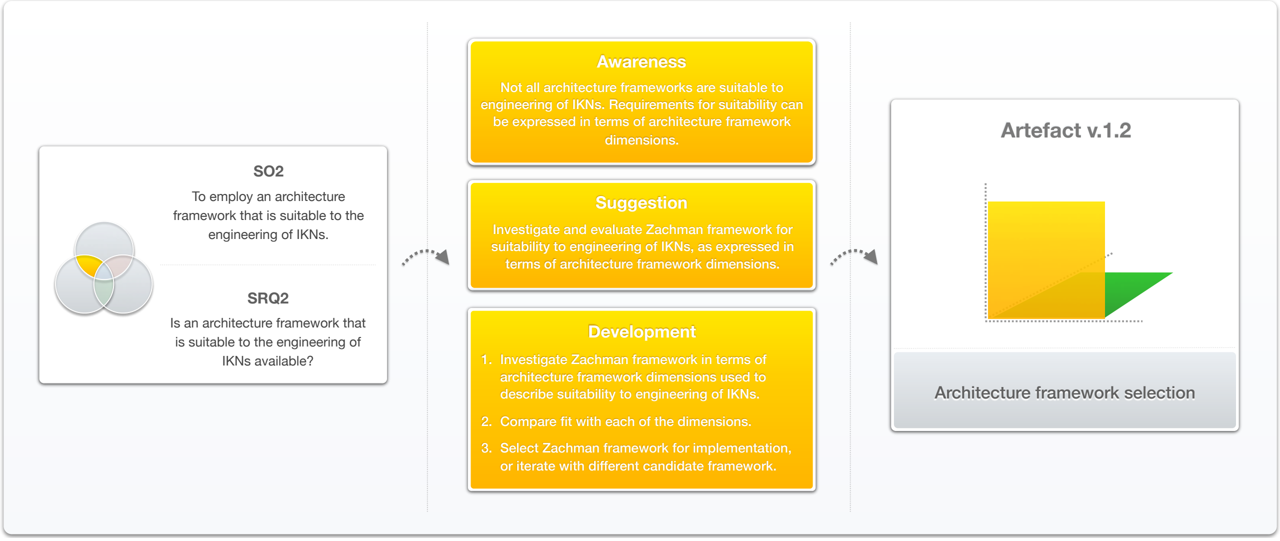
6.7 Conclusion

This chapter presented the second cycle in the incremental design and development process of a reference architecture for IKNs. The cycle therefore considered the compound requirement perspective that combines the technical requirement and scenarios (refer to yellow intersection in Figure 6.7).

The design cycle addressed the second solution objective (SO2). The investigation in this chapter was directed by the second secondary research question (SRQ2). This question asked whether an architecture framework that is suitable to the engineering of IKNs is available.

Following a concise presentation of relevant literature, the *awareness* of the problem outlined that not all architecture frameworks are suitable to engineering of IKNs, and that the requirements for suitability can be expressed in terms of architecture framework dimensions. The *suggestion* was subsequently made that the Zachman framework should be investigated and evaluated for suitability to the engineering of IKNs. The Zachman framework is an intuitive candidate for suitability to the engineering of IKNs, since the interpretation of the framework as an enterprise ontology (Kappelman & Zachman 2013) matches the quality attributes required.

The *development* of the architecture framework selection was done through investigation and evaluation of the Zachman framework in terms of the dimensions used to describe the requirements for suitability in



SO2

To employ an architecture framework that is suitable to the engineering of IKNs.

SRQ2

Is an architecture framework that is suitable to the engineering of IKNs available?

Awareness

Not all architecture frameworks are suitable to engineering of IKNs. Requirements for suitability can be expressed in terms of architecture framework dimensions.

Suggestion

Investigate and evaluate Zachman framework for suitability to engineering of IKNs, as expressed in terms of architecture framework dimensions.

Development

1. Investigate Zachman framework in terms of architecture framework dimensions used to describe suitability to engineering of IKNs.
2. Compare fit with each of the dimensions.
3. Select Zachman framework for implementation, or iterate with different candidate framework.

Figure 6.7: Conclusion of second design cycle

Table 6.2. The Zachman framework and the architectural information it describes were therefore investigated in terms of type and scope of information, quality attributes, stakeholders, transformations and nature. For each of the dimensions, the level to which the framework satisfies the requirements was evaluated. As the Zachman framework satisfied all the requirements, it was selected as the architecture framework to be employed by the reference architecture for IKNs.

The selection of the Zachman framework implies that the architectural views provided by the framework, i.e. inventory sets, process flows, distribution networks, responsibility assignments, timing cycles and motivation intentions as manifested through a number of reified perspectives, *populate the third axis of the modelling framework embedded in the reference architecture for IKNs* (refer to section 5.6.2). The search for an architecture framework that satisfies the requirements for the engineering of IKNs represented the investigation of the *relationships between the views axis and the axis indicating the engineering of IKNs throughout their life cycle* (refer to the yellow plane to the right of Figure 6.7).

The architecture framework selection contributed by this chapter is seen as the second version of the reference architecture for IKNs (v.1.2). It answers SQ2 and achieves SO2 *by identifying an architecture framework that is suitable to the engineering of IKNs*. The second cycle is therefore terminated through circumscription.

The generated knowledge about the final artefact, gained through design decisions and the development of a second version, can now be used to update the requirement frame of reference and leads to the initiation of a next design cycle. The artefact is further developed by means of a third design cycle in chapter 7, which develops an artefact development roadmap outlining how the Zachman framework can be implemented in the reference architecture for IKNs.

7. Artefact development roadmap

7.1 Introduction

This chapter presents the third cycle in the incremental design and development process of the reference architecture for IKNs. The cycle considers the compound requirement perspective that combines the technical and functional requirements (refer to the red intersection in Figure 7.1).

The design cycle addresses the third solution objective (SO3) by indicating how the selected architecture framework can be implemented to enable the artefact to function as a reference architecture. The search process in this chapter is directed by the third secondary research question (SRQ3), which asks how such an implementation of the architecture framework can be achieved.

The chapter therefore builds on the selection of the Zachman framework as the architecture framework for the reference architecture for IKNs in chapter 6. The chapter's contribution is delivered in the form of an *artefact development roadmap* that indicates *how the Zachman framework can be implemented to enable the artefact to function as a reference architecture*. The development of this artefact development roadmap represents the investigation of the *relationships between the views and meta-relationships axes of the modelling framework* that forms the foundation of the reference architecture for IKNs. This roadmap achieves the solution objective (SO3), and therefore circumscribes, and restricts, the activities of the design cycle, which leads to the initiation of a subsequent design cycle that develops the final artefact in chapter 8.

The solution objective and associated secondary research question that drives the design cycle is discussed in section 7.2. Section 7.3 presents background selected from existing literature that forms the basis for the ensuing discussion in this chapter.

Design reasoning follows thereafter, with section 7.4 discussing the problem awareness that the implementation of the selected architecture framework must enable the artefact to function as a reference architecture. Section 7.5 makes the suggestion to compile an artefact development roadmap indicating how this functionality can be enabled.

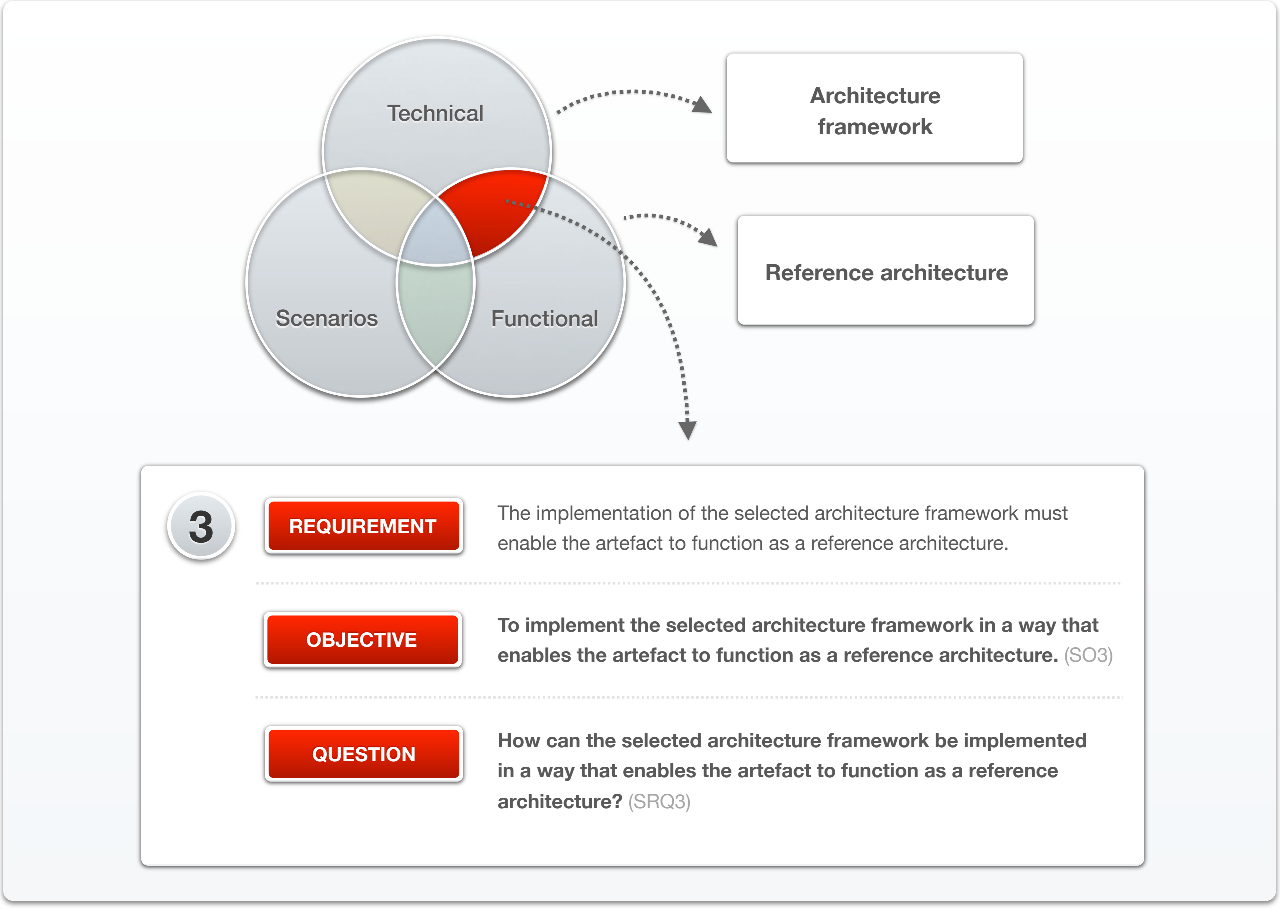
Section 7.6 documents the development of the artefact development roadmap, starting with an investigation of the implications in terms of architecture framework dimensions of enabling the artefact to function as a reference architecture. It is then determined how the selected architecture framework can be implemented within the artefact through specification of appropriate values for various architecture framework dimensions. These dimensions and values are then used to compile the artefact development roadmap before section 7.7 concludes this chapter.

7.2 Design cycle drivers

This section presents the solution objective and secondary research question that drive this design cycle. This is done in order to position the design cycle (and chapter) in the context of the overall design and development work performed in Part 3 of the research design.

The design cycle is driven by SO3 and SRQ3 as derived by considering the red requirements intersection of the technical and functional requirements (refer to Figure 7.1).

Compound requirement:	The implementation of the selected architecture framework must enable the artefact to function as a reference architecture.
Solution objective:	To implement the selected architecture framework in a way that enables the artefact to function as a reference architecture. (SO3)
Research question:	How can the selected architecture framework be implemented in a way that enables the artefact to function as a reference architecture? (SRQ3)



The implementation of the selected architecture framework must enable the artefact to function as a reference architecture.

To implement the selected architecture framework in a way that enables the artefact to function as a reference architecture. (SO3)

How can the selected architecture framework be implemented in a way that enables the artefact to function as a reference architecture? (SRQ3)

Figure 7.1: Drivers of third design cycle

7.3 Background

This section presents literature that directly informs the awareness for the third (red) design cycle.

The following section reviews the paper by Noran (2003), in which the author presents an extensive analysis of the Zachman framework from the GERA perspective. In this analysis, the Zachman framework is investigated from several points of view, showing how the framework can be implemented as a reference architecture. Of particular interest are the views on genericity (refer to section 7.3.1), life cycle phases (refer to section 7.3.2) and life history (refer to section 7.3.3) in the Zachman framework.

7.3.1 Genericity in the Zachman framework

This section presents a discussion of genericity and the subsequent implementation of the Zachman framework to achieve a reference architecture. It is therefore highly relevant with regard to the solution objective of this chapter and is the basis of the thinking in sections 7.6.1 and 7.6.2 in which the artefact development roadmap is set up to produce a reference architecture.

A common characteristic or dimension with which to differentiate between various reference architectures is genericity or meta-level (Kosanke et al. 1999; Williams & Li 1999; IFIP-IFAC Task Force 2003; Greefhorst et al. 2006). Varying levels of genericity refer to levels of aggregation in the reference architecture, and therefore also the varied scope of applicability thereof in EE efforts. While the Zachman framework does not explicitly contain a genericity dimension, separate versions of the framework, however, may be used for the three main areas of genericity as defined in GERA, i.e. generic, partial and particular (refer to Figure 7.2) (1). In this approach to genericity, the sixth row representing the operating instance of the enterprise is omitted from the generic and partial frameworks (Noran 2003) (represented with red bars in Figure 7.2).

In such a configuration of the Zachman framework, the particular version of the framework contains fully developed models of the enterprise instance, while the partial framework contains reference models (Noran 2003). These reference models may be models of prototypes, class models or models of patterns that are common to the architecture of a set of enterprises. The generic version of the framework then contains meta-models, i.e. modelling constructs and their relationships, ontologies and glossaries.

Referring to Figure 7.2 (2), there is another implicit genericity dimension along the perspectives (vertical) axis of the Zachman framework. The perspectives of the stakeholders involved in the life cycle phases of the enterprise differ, but they refer to the same enterprise and are therefore connected to the same version of the framework (Zachman 1987). For instance, a business requirement from the business management perspective (second row) may transform into several system requirements in the architect perspective. These system requirements become physical through the specification of a combination of technologies in the engineer perspective and are implemented through various tool configurations in the technician perspective.

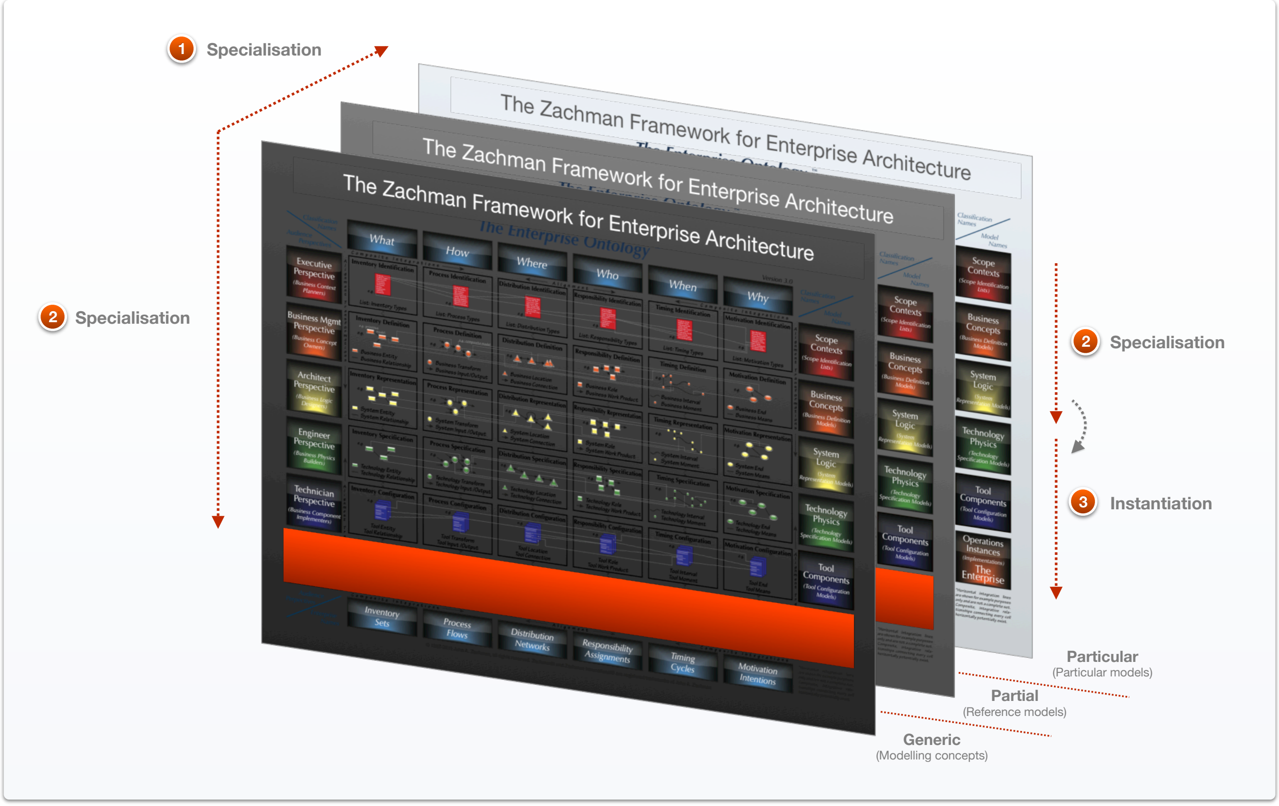


Figure 7.2: Genericity dimensions in the Zachman framework

(adapted from Noran 2003)

Each of these representations may use specific means of expressions, but in essence they refer to the same artefact from different perspectives. The degree of specialisation of the perspectives and their models gradually increases through the process of reification, i.e. approaching the operational instance in the sixth row of the framework (Noran 2003). This increase in the degree of specialisation is a result of decisions that are made regarding various enterprise parameters, e.g. enterprise boundaries, concept definitions, design logic and technologies. Instantiation occurs when abstract representations become reality in the engineer and technician perspectives of the particular version of the Zachman framework (refer to Figure 7.2) (3). This instantiation is associated with the implementation phase of the GERA life cycle (refer to section 7.3.1).

7.3.2 Life cycle phases in the Zachman framework

This section presents a mapping of the perspectives (rows) of the Zachman framework to a generic enterprise life cycle, as found in GERA. This mapping forms the basis of the thinking in section 7.6.4, where these perspectives are mapped to the IKN life cycle to inform the specification of stakeholders in the artefact development roadmap.

As was noted in section 5.3.3, two main types of architecture can be identified in systems engineering and enterprise integration literature: architectures that represent the structure of a system at a given point in time (type 1), and architectures which describe the possible phases and artefacts involved in the life cycle of the system or enterprise (type 2) (Kappelman & Zachman 2013; Vernadat 1996; Noran 2003; Cloutier et al. 2010). These life cycle phases are understood as a set of partially ordered possible enterprise processes or activities, which may be performed once, several times or not at all in the enterprise during its existence (IFIP-IFAC Task Force 2003).

The Zachman framework does not include an explicit life cycle dimension and leans towards being a type 1 architecture. The framework, however, does take an indirect approach towards the life cycle, with life cycle phases being related to perspectives of the various stakeholders involved in the enterprise engineering effort (Noran 2003). These perspectives have different levels of abstraction to consider the enterprise entity in question and can therefore be mapped to the GERA life cycle phases. In such a mapping, GERA life cycle activities are matched with the deliverables that certain Zachman framework stakeholders produce (refer to Figure 7.3).

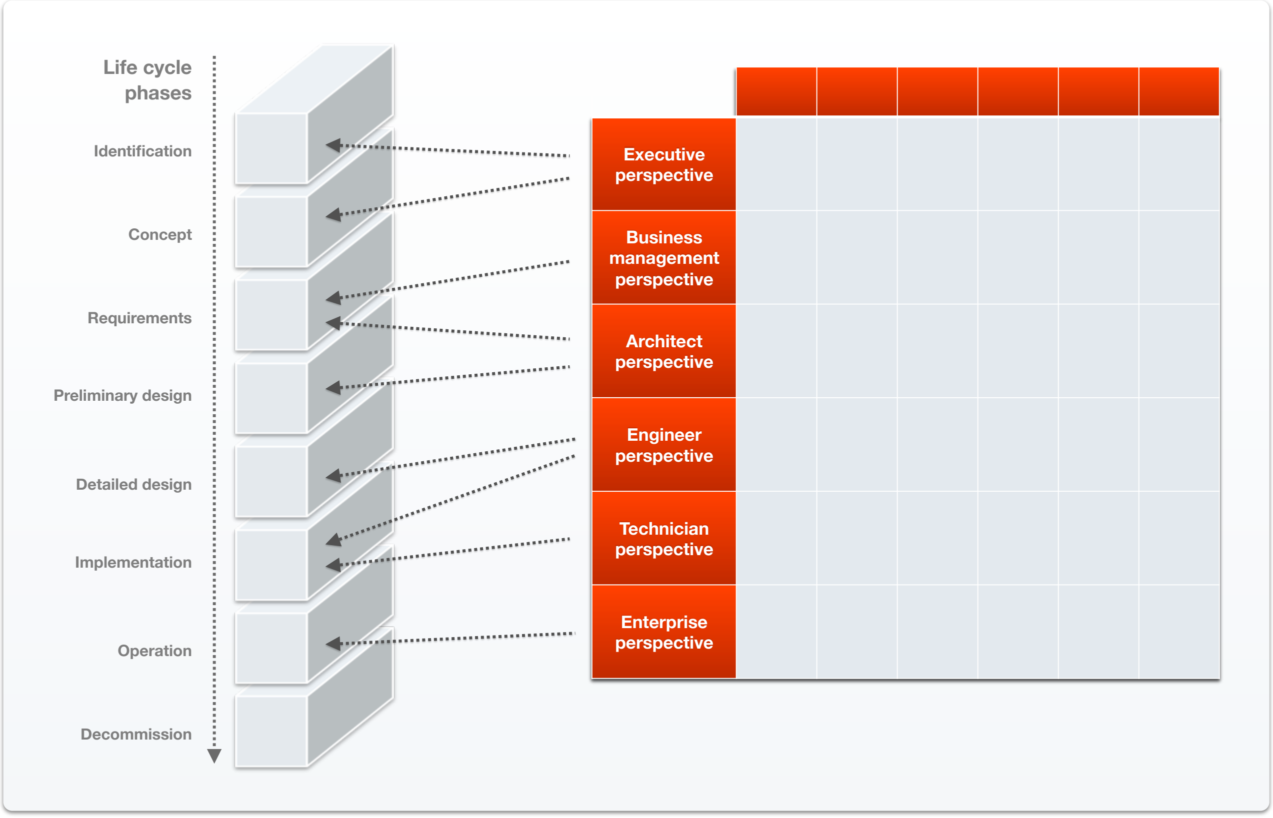


Figure 7.3: Mapping of Zachman framework perspectives to GERA life cycle phases

(adapted from Noran 2003)

The executive perspective (row 1) of the Zachman framework identifies the boundaries of the enterprise, matching the deliverables of the GERA identification phase (Noran 2003). The executive perspective, however, also describes the enterprise at a strategic level and is therefore associated with the concept phase as well. By defining concepts in the second row of the Zachman framework, the business management perspective describes the user requirements for the enterprise and is matched with the requirements phase. The requirements phase also delivers system requirements and is therefore also relevant to the architect perspective. The architect perspective furthermore produces the preliminary design at a system level, before the engineer perspective extends this preliminary design to a detailed design through the specification of technologies. In the implementation phase, the technician perspective implements these technologies to produce the enterprise perspective in the operation phase.

Although the GERA life cycle does not include a refinement phase, this action is included through the concept of life history (refer to section 7.3.3), in which sections of the life cycle are iterated to produce a refined version of the enterprise.

The Zachman framework does not explicitly include a perspective that can be associated with the decommission phase of the GERA life cycle. As was shown in section 5.6.5.5, however, this phase can be viewed as a special case of the refinement action resulting in a migration towards a specific decommissioned version of the enterprise. If this approach is taken, the Zachman framework does indeed include the necessary perspectives to cover the entire GERA life cycle.

7.3.3 Life history in the Zachman framework

This section presents a view on how the GERA life history concept is supported in the Zachman framework. Life history of an enterprise and the time-based versioning of models relates to the transformation element of the artefact development roadmap in section 7.6.6.

Enterprises are required to constantly adapt to their changing environments through the implementation of change processes. These change processes that occur during the life of an enterprise are concurrent, interact with one another, and can be suitably modelled through the GERA life history concept (IFIP-IFAC Task Force 2003). Whereas a life cycle is seen as the finite set of generic phases and steps a system *may* go through in its life span, its life history is the actual sequence of steps an enterprise *has* gone through (or *will* most likely go through) (Noran 2003).

The Zachman framework does not have an explicit life history concept, i.e. an independent timeline for the succession of enterprise versions. For successive or non-concurrent change processes, the life history of the enterprise may be represented through the "When?" column, which implies temporality and succession (Noran 2003). However, more flexibility is achieved by adding an independent time axis that shows how the architecture evolves over time.

Each change process may be modelled in its own timing representation. Alternatively, multiple change processes could be simultaneously represented on a combined timing diagram, thus indicating their possible interactions. In such a model of representing the enterprise life history there would be an integration with the “Why?” column to indicate the purpose of each change process and interaction (Noran 2003).

Another temporal aspect in the Zachman framework relates to versioning as a possible recursive effect in the framework. In such a versioned implementation of the framework there may be ‘as-is’ and ‘to-be’ versions of each of the models in the framework (Sowa & Zachman 1992) (refer to Figure 7.4) (1, 2). Versioning may be considered a temporal concept, as it identifies certain stages in the evolution or life history of an enterprise (Noran 2003; Álvares-Ribeiro et al. 2004). A historic repository of the enterprise’s architecture over time can also be constructed through versioning. This repository can be used to make appropriate selections from the life cycle phases to influence the future of the enterprise.

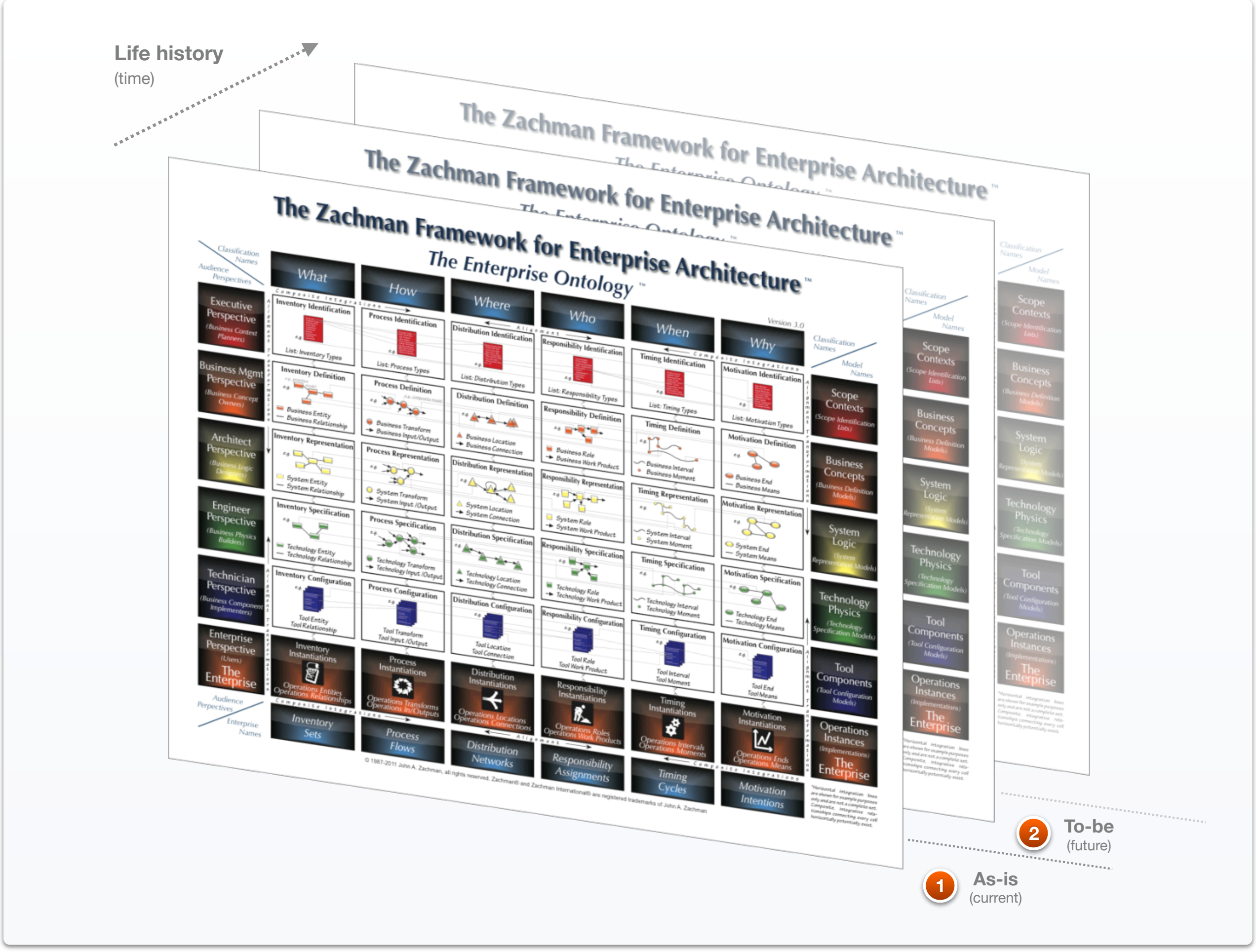


Figure 7.4: Versioning and life history in the Zachman framework

This section presented various perspectives on the Zachman framework, namely its approach to genericity, life cycle phases and life history. This discussion serves as context for the problem awareness that initiates design reasoning in the following section.

7.4 Awareness

This section presents the awareness step of the design reasoning in the third (red) design cycle.

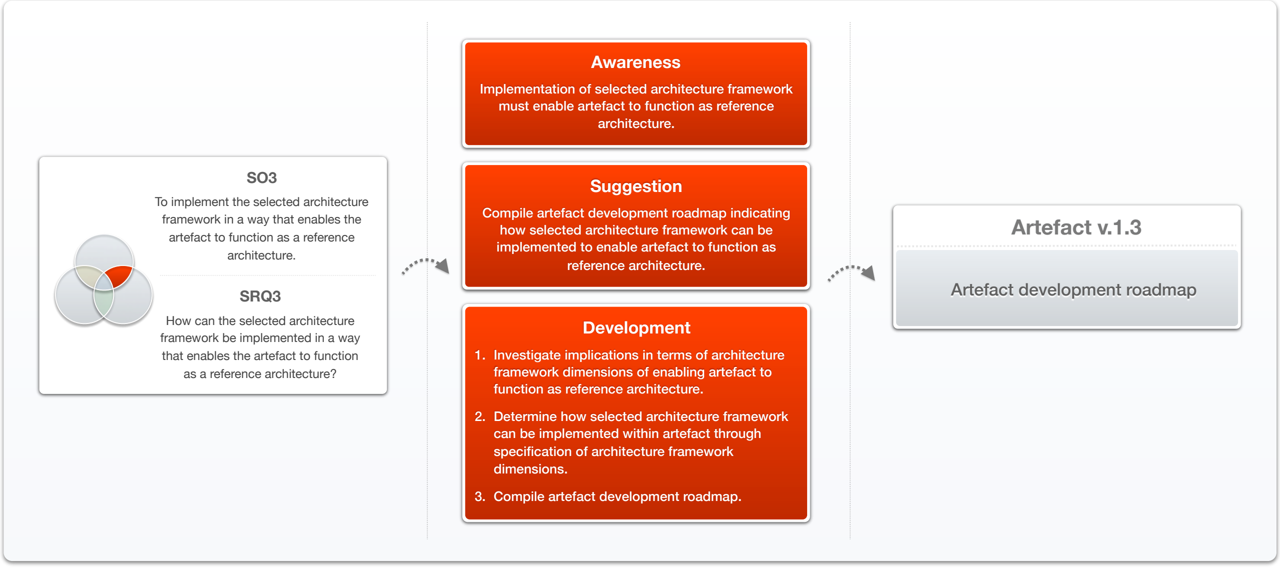


Figure 7.5: Reasoning in third design cycle

Given SO3 and SQ3 (refer to section 7.2), the awareness of the problem that is considered in this design cycle can be stated as follows (refer to Figure 7.5):

> **The implementation of the selected architecture framework must enable the artefact to function as a reference architecture.**

This problem awareness articulates that to achieve the solution objective considered in this chapter (SO3), an *implementation* of the *Zachman framework* that enables the solution artefact to *function as a reference architecture* is required. The underlying principle expressed in this awareness is that in the design of any artefact, its *behaviour is brought about, and consequently explained, by its construction* (Dietz & Hoogervorst 2008).

The awareness is furthermore in keeping with the approach for selecting and subsequently adapting or implementing an architecture framework, as proposed by Schekkerman (2004). Schekkerman notes that architecture frameworks can be leveraged to provide a starter set of the issues and concerns that have to be addressed in architecture development. When the decision is to adapt such an existing framework, rather than to invent a new one, the selected framework needs to be customised to suit the particular needs of the situation (Schekkerman 2004). In this chapter, an implementation of the Zachman framework is specifically adapted to the *functional requirements* of the reference architecture for IKNs, as illustrated by the red intersection in Figure 7.1. In chapter 8 this implementation of the framework is also adapted to the scenario-based requirements of the IKN domain, as illustrated by the greyscale blue (centre) intersection in Figure 7.1

7.5 Suggestion

This section presents the suggestion step of the design reasoning in the third (red) design cycle.

In light of the awareness of the problem presented in the previous section, the following suggestion for a possible solution is made (refer to Figure 7.5):

> **Compile an artefact development roadmap indicating how the selected architecture framework can be implemented to enable the artefact to function as a reference architecture.**

The objective of this *artefact development roadmap* is to show how the Zachman framework can be implemented in the reference architecture for IKNs in such a way that its construction brings about its function. The required implementation of the Zachman framework which will enable the artefact to function as a reference architecture can be expressed in terms of the *architecture framework dimensions* contributed by Greefhorst et al., as they can be used as a checklist for the construction of a new architectural artefact (Greefhorst et al. 2006).

Initially, the implications of enabling the artefact to *function as a reference architecture* should be investigated in terms of the architecture framework dimensions. Thereafter it should be determined how the *required implementation* of the Zachman framework can be coherently delivered through the specification of values in other relevant architecture framework dimensions. These values should then be collated to compile the artefact development roadmap that is to be followed to create the artefact itself in chapter 8.

7.6 Development

This section presents the development step of the design reasoning in the third (red) design cycle and therefore documents the compilation of the artefact development roadmap.

7.6.1 Meta-level

This section specifies the function of the required implementation of the Zachman framework to be a reference architecture.

The primary function of a partial architecture or reference architecture is the provision of inputs for the design of particular architecture descriptions for instances in the class of systems or enterprises covered by the reference architecture (refer to section 5.3.3). This is done through the provision of reference models that serve as the basis for particular models that accurately describe the entity instance (Noran 2003; Cloutier et al. 2010). A reference architecture is therefore viewed as a set of generalised or partial models for particular architecture descriptions, which in turn serve as models for the instantiation of the functioning system or enterprise (refer to Figure 7.2). This role of a reference architecture is illustrated by the placement of the reference architecture for IKNs in the solution space created by the modelling framework of the artefact (refer to Figure 7.6).

Figure 7.6: Modelling framework – Abstraction vs. generalisation

When interpreting this function of a reference architecture in terms of architecture framework dimensions, it follows that the major implication is along the *meta-level* dimension. This dimension indicates architecture frameworks that, instead of domain-specific models, provide general classifications and relationships. Greefhorst et al. (2006) refer to such generalised models as meta-models. This study, however, adopts the notion of a meta-model provided by Kühne (2005). This model characterises 'meta-ness' as a two-level detachment from the original and achieves this detachment through the double application of an abstraction operation.

This double application of an abstraction operation, however, does not strictly apply for the reference models contained in a reference architecture (refer to Figure 7.6). While an *abstraction* operation does occur between the architecture instantiation and the architecture description (1), a different operation occurs between the architecture description and the reference architecture, namely that of *generalisation* (2). It is therefore not a case of the same operation being performed twice. The reference models contained in the reference architectures do not describe *the models* contained in particular architecture descriptions, as would be the case with a true meta-model. The reference models rather describe *patterns* that are common to the architectural instances that are described by various architecture descriptions (Cloutier et al. 2010; Noran 2003).

To enable the artefact to function as a reference architecture, the required implementation of the Zachman framework is to contain partial or reference models in the sense that they are discussed above. The value of *reference architecture* is assigned to the meta-level dimension and the artefact is therefore positioned in the partial (middle) layer of Figure 7.2.

It is necessary to ensure that the coherent construction of the reference architecture for IKNs brings about its required function. The impact of the allocation of this value to the meta-level dimension on the other relevant architecture framework dimensions is therefore now investigated. Values are accordingly allocated to these dimensions.

7.6.2 Nature

This section specifies that, in order to enable the artefact to function as a reference architecture, the required implementation of the Zachman framework should be populated with reference models, rather than particular models. Possible sources for the development of such reference models are also discussed.

This dimension determines the nature of the architectural information structured by the architecture framework. Provided that the meta-level of the required implementation of the Zachman framework is set to *reference architecture* (refer to section 7.3.1), the value of the *nature dimension* is specified as *reference models*.

The objective of these reference models is to capture generalised patterns of components and relationships that exist in the architecture descriptions of enterprise construction and to represent them in such a way that they can be reused as the basis for the development of particular architectural models (Noran 2003; Cloutier et al. 2010). The intention of these reference models is therefore not to describe every possible combination of relationships that can feasibly be constructed in more specialised models. Since these reference models represent generalised patterns and are seen as the starting point for the development of particular models, they are not seen as limiting to the design of the particular models. While the reference models are seen as a complete set at their specific level of abstraction, it is possible to extend the content of particular models beyond the content of the reference models.

Inputs for the development of reference models include existing reference models for sections of the reference architecture, e.g. innovation process models. Particular models from architecture descriptions from various enterprises that form part of the range of the reference architecture, which can be obtained by modelling multiple IKN instances, can also serve as inputs. Another input is further generalised reference models that can be specialised for the particular range of the reference architecture, e.g. the reference models for CNOs provided by the ARCON framework (Camarinha-Matos & Afsarmanesh 2008a).

The reference models included in the required implementation will be single-variable or primitive models, according to the Zachman framework classification schema, and can be combined to form multi-variable or composite models (Zachman 2011). When the inputs obtained from the above-mentioned sources provide architectural information that is of a composite (multi-variable) nature, as is the case in most modelling environments, this information needs to be deconstructed into primitive (single-variable) elements.

7.6.3 Scope

This section specifies that the required implementation of the Zachman framework describes a network as an enterprise and therefore includes all six columns of the framework.

As was noted in section 6.6.2, the scope of architectural information covered by the Zachman framework is variable, due to the fact that the underlying logic of the framework can be applied to any object (Sowa & Zachman 1992; Kappelman & Zachman 2013). The framework can therefore be scaled to fit a variety of values or entity types along the scope dimension of architecture frameworks. The scope of specific implementation of the framework is determined by the architectural boundaries identified in the first row, along with the definition of these concepts in the second row (Sowa & Zachman 1992; Noran 2003).

The artefact behaviour specification indicates that the artefact will be used in the design, refinement and phase-out phases of the IKN life cycle to engineer IKNs (refer to section 5.6.5.6). In these phases with their respective stakeholders (refer to section 7.6.4) the artefact is functioning as a reference architecture with an *enterprise* scope. Given this context, the reference architecture will produce first-row architectural information that specifies the scope of the architecture as that of an entire *network* seen as an enterprise.

The Zachman framework is by default configured to handle this enterprise-wide scope of architectural information, given its purpose as a framework for enterprise architecture. The Zachman framework provides a complete set of abstractions or descriptive representations of any object that serves as the modelling subject, which are by default labelled to describe an enterprise scope. These abstractions answer six basic interrogatives regarding the subject, namely “What?”, “How?”, “Where?”, “Who?”, “When?” and “Why?” (refer to Figure 7.7).

In order to comprehensively describe the construction of IKNs as enterprises, the complete set of abstractions are employed in the reference architecture for IKNs through their inclusion as the architectural views in the modelling framework for the artefact (refer to section 6.6.7). The required implementation of the Zachman framework therefore includes all *six columns* of the framework. They are as follows:

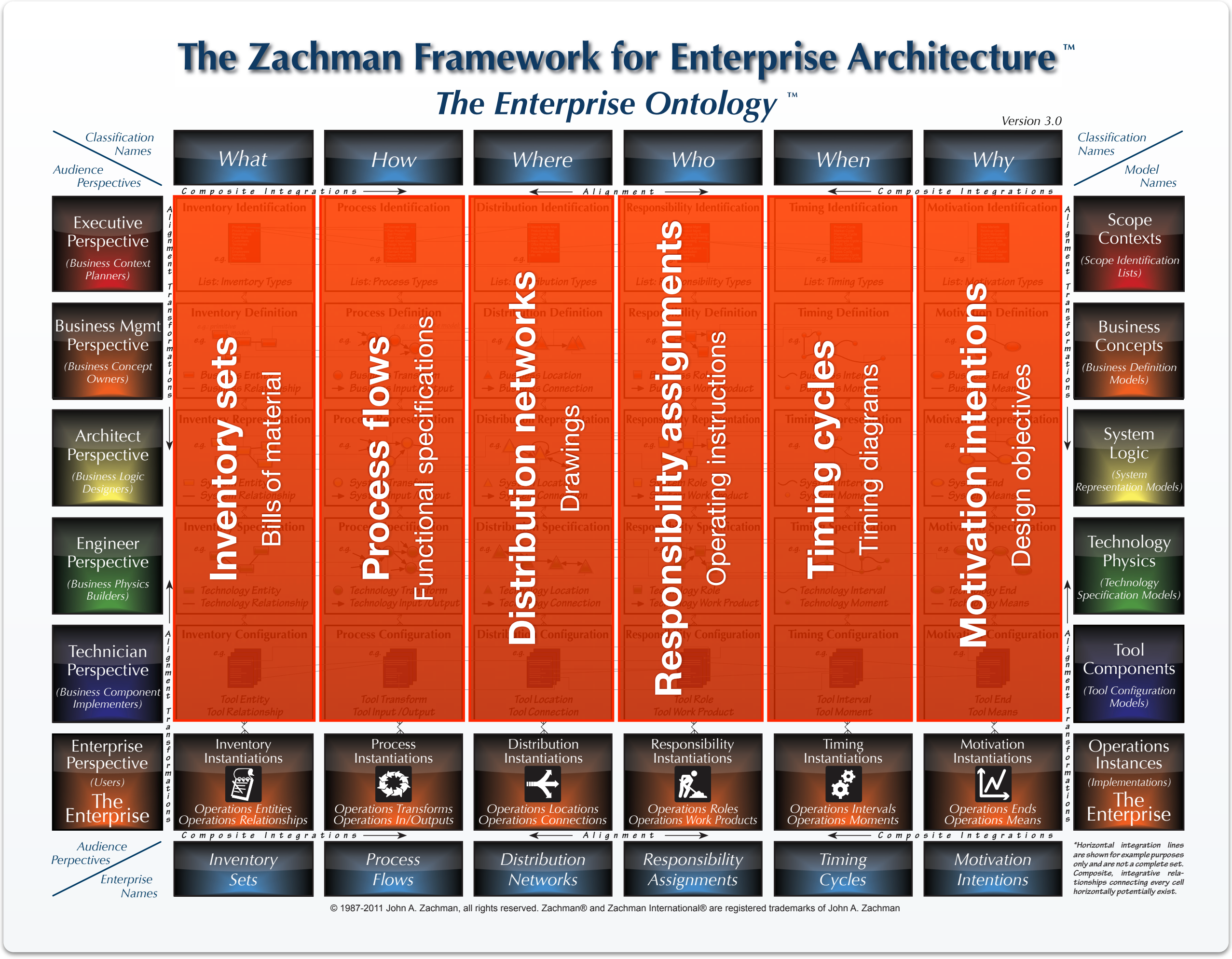


Figure 7.7: Scope of the required implementation of the Zachman framework

- Inventory sets – described in bills of material;
- Process flows – described in functional specifications;
- Distribution networks – described in drawings;
- Responsibility assignments – described in operating instructions;
- Timing cycles – described in timing diagrams;
- Motivation intentions – described in design objectives.

Including the complete set of primitive (single-variable) abstractions, the artefact allows for the construction of composite (multi-variable) models. These composite models represent concepts in the way they would be encountered in the real world, i.e. as multi-variable compounds in the functioning *enterprise*. The relationships that exist between elements from different abstractions that allow for the construction of composite models are known as *integrations*.

7.6.4 Stakeholders

This section specifies a primary and secondary audience for the required implementation of the Zachman framework.

The stakeholders that are addressed by an enterprise-class reference architecture range from strategists that consider the context of the enterprise to technicians that configure tools to achieve alignment with business requirements (Sowa & Zachman 1992).

In section 5.6.5.6 the artefact behaviour specification specified that the primary audience of the reference architecture for IKNs are stakeholders that are directly involved in the design, refinement and phase-out phases of the life cycle of IKNs (phases indicated in red in in Figure 7.8).

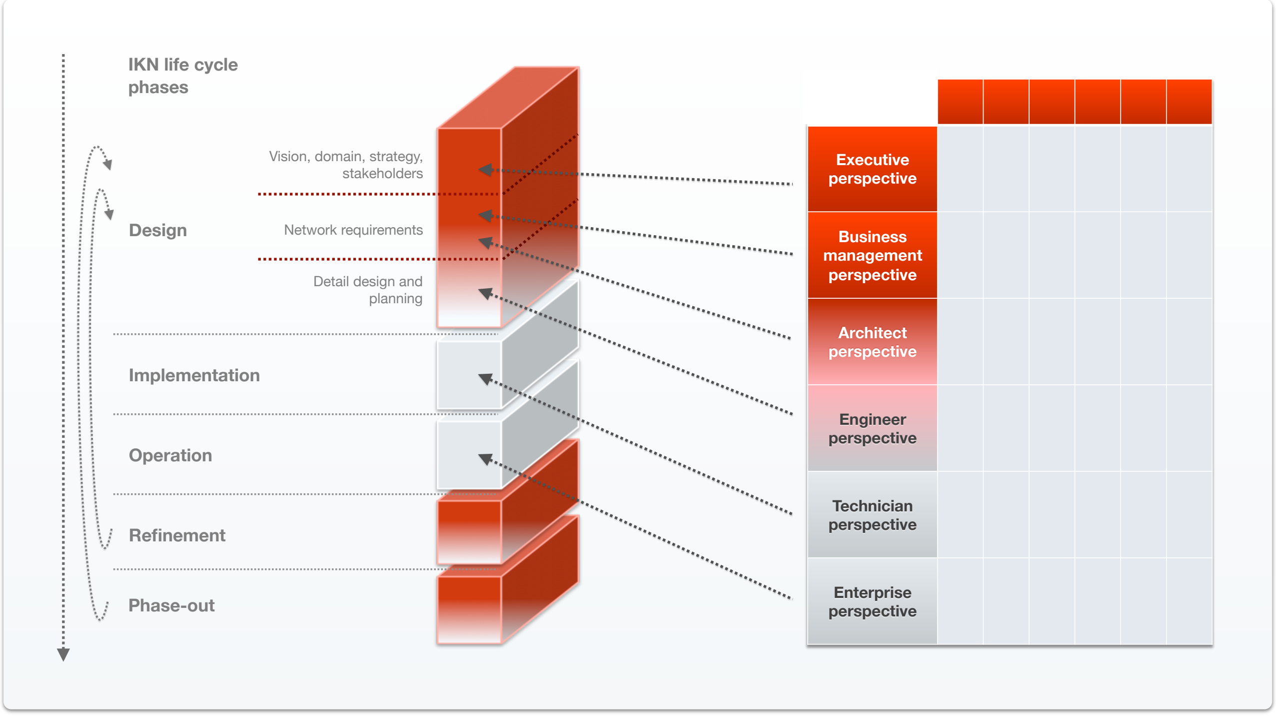


Figure 7.8: Mapping of Zachman framework to IKN life cycle phases

A mapping of the perspectives of the Zachman framework to the life cycle of IKNs will illuminate the specification of stakeholders for the required implementation of the framework, and is presented in Figure 7.8. This mapping is based on the mapping of the perspectives to the generic enterprise life cycle of GERA presented in section 7.3.2.

Figure 7.8 illustrates that the *executive* perspective (row 1) is mapped to the deliverables of the vision, domain, strategy and stakeholders sub-phase of the design phase of the IKN life cycle, as it identifies the boundaries and context of the network. The *business management* perspective (row 2) is mapped to the network requirements sub-phase, as it defines the concepts and user requirements that form part of the

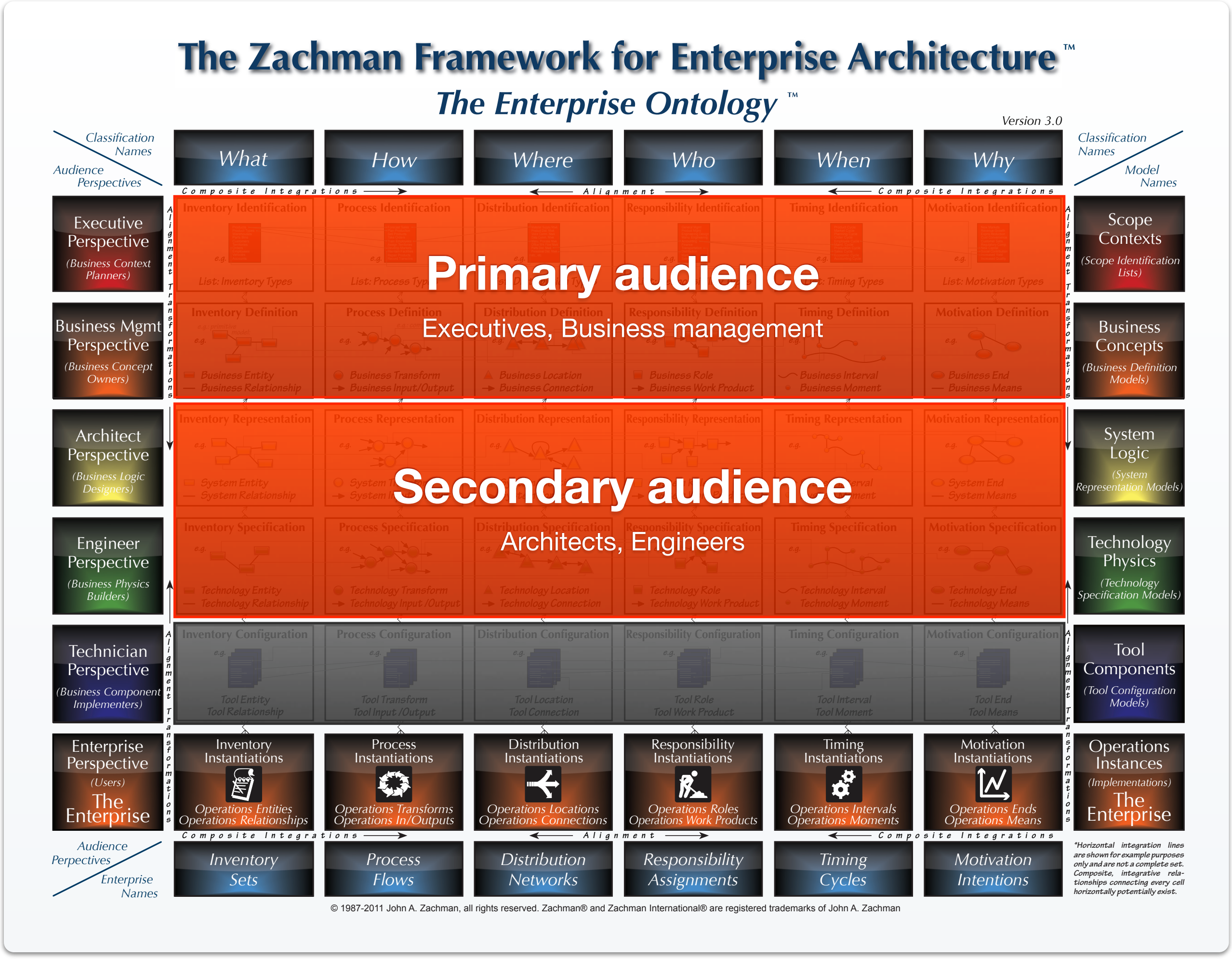


Figure 7.9: Stakeholders of the required implementation of the Zachman framework

network design. Both these stakeholders are actively architecting the business layer of the enterprise by designing and describing its fundamental organisation. These perspectives therefore rely on the reference architecture for inputs that serve as the basis for their particular models and are identified as the *primary audience* of the required implementation of the Zachman framework (refer to Figure 7.9).

Along with the user requirements that are associated with the business management perspective, the network requirements sub-phase of the design phase of the IKN life cycle additionally produces system requirements. It is therefore also associated with the *architect* perspective (row 3). The *engineer* perspective (row 4) is also associated with the detail design and planning sub-phase through the extension of the preliminary network design with the specification of technologies.

In the implementation phase, the *technician* perspective (row 5) implements the technologies specified in the engineer perspective to produce the *enterprise* perspective (row 6) and the functional IKN in the operation phase. The technician perspective is only linked to the implementation phase of the IKN life cycle, which was not identified as part of the artefact behaviour specification. This perspective is therefore not a stakeholder in the required implementation of the Zachman framework (refer to Figure 7.9).

As illustrated with the recursive arrows to the left of Figure 7.8, the refinement phase of the IKN life cycle resembles an iteration of the design phase. Certain design parameters, however, are fixed, primarily through the existence of a functional network that influences the redesign (refer to section 5.6.5.4). Network phase-out, in turn, is viewed as a special case of the above refinement phase, where a functional network exists and a phased out network is pre-specified as the migration target (refer to section 5.6.5.5). For both of these life cycle phases, however, the mapping of the Zachman framework perspectives to the design activities remains the same. These phases of the IKN life cycle, therefore, do not alter the view on the stakeholders in the required implementation of the framework. The values allocated to the stakeholder dimension of the required implementation of the Zachman framework are illustrated in Figure 7.9.

7.6.5 Type of information

This section limits the required implementation of the Zachman framework to the top two rows of the framework.

The type of information that is included in an enterprise-class reference architecture ranges from business-oriented information that identifies the boundaries of the enterprise and defines business concepts to technical information that describes the logic, technologies and configurations that constitute the technical architecture (Zachman 1987; Sowa & Zachman 1992).

In the Zachman framework the vertical process of reification that occurs in the transformations between the rows reifies the architectural information, thereby making the information progressively more concrete and case-specific (Noran 2003; Kappelman & Zachman 2013). This results in more generalisable architectural information at the top of the framework, i.e. the business information in the top two rows (refer to Figure 7.10). More particular or case-specific architectural information is found lower in the framework, i.e. the technical information in the bottom three rows. This means that a reference architecture implementing the Zachman framework is able to generalise more common patterns in the architectural information at the top of the framework than at the bottom (refer to section 7.3.1, Figure 7.2). The reference architecture can therefore provide more meaningful inputs through reference models that form the basis of particular models in the business layer of the architecture.

This view on the feasibility of reference models in the various rows in the Zachman framework supports the specified value of the of the stakeholders dimension (refer to section 7.6.4) to determine the type of information to be included in the reference architecture. The reference architecture for IKNs should include only *business* information, as is presented in the *top two rows* of the Zachman framework. The models in the top row of the required implementation identify the boundaries of the common architectural patterns

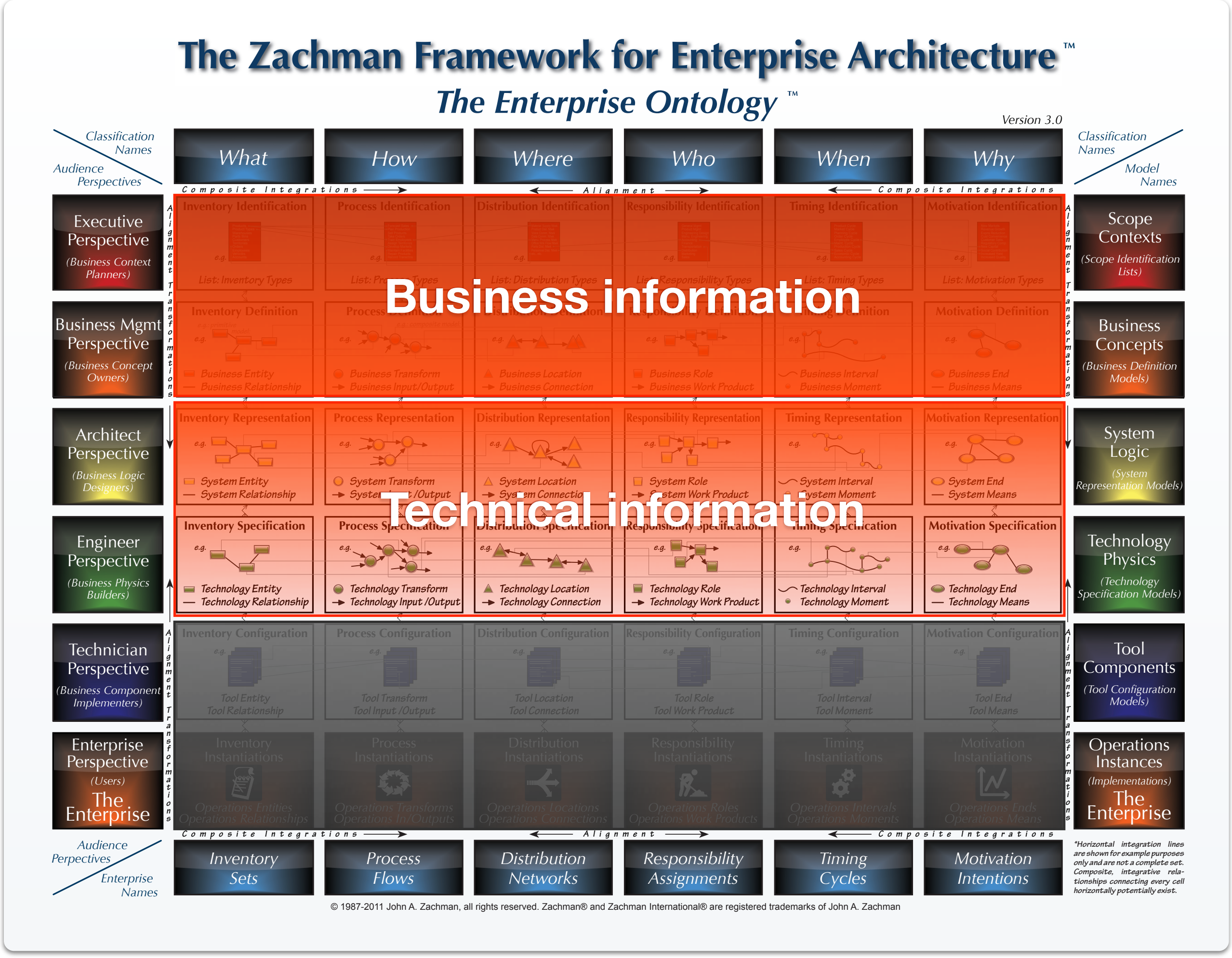


Figure 7.10: Type of information in the required implementation of the Zachman framework

that are found in the enterprises that fall in the range of the reference architecture. These models indicate the types of inventories, processes, distribution, responsibilities, timing and motivations that exist in the reference architecture.

The models in the second row of the required implementation define the business concepts that fall within the boundaries of enterprises in the range of the reference architecture, as identified in the first-row models. A transformation therefore occurs between these rows – from the identified boundaries of the common architectural patterns in the top row to the definitions of concepts that constitute the common architectural patterns that fall within these boundaries in the second row.

The architectural information in these top two rows is aimed at the primary audience of the artefact, who are directly engaging with the business layer of the enterprise architecture to create a particular business architecture description in the design, refinement and phase-out phases of the IKN life cycle. This particular business architecture description serves to provide the support requirements for the secondary audience of the artefact who are looking to achieve alignment of their technical efforts with the business layer. By retaining the vertical reification process of the Zachman framework, the reference architecture also provides directives for the development of particular technical models in the third and fourth rows albeit no reference models are provided.

As it is not included in the stakeholders of the required implementation of the Zachman framework, the fifth row is entirely omitted in terms of information type. The reification process that links this row with the rest of the reference architecture is however retained. Given the meta-level of the implementation of the framework (reference architecture), the sixth row of the framework is also removed (refer to section 7.3.1).

It should be noted that a case can be made for the feasibility of reference models for the third row of the framework that describe patterns in the system logic that are commonly found in IKNs. The decision to exclude these models from the reference architecture and solely focus on the business information in the top two rows is made with the study's research scope in mind. Such third-row reference models would directly address members of the secondary audience of the artefact and would therefore be valuable to improve the alignment of business and technology in IKNs at a system level. The creation of third-row reference models from common patterns in the system logic of IKNs, however, is a non-trivial exercise. The extension of the reference architecture through the development of these models is discussed amongst the opportunities for further research in section 11.3.

7.6.6 Transformations

This section specifies that the required implementation of the Zachman framework should be able to produce both as-is and to-be versions of particular architecture descriptions.

This architecture framework dimension generally uses changes in time as the criterion and typically distinguishes between multiple states, including the transitions between them. As the (time-based) life history of an IKN unfolds, the appropriate phases of the IKN life cycle with their associated activities are selected for execution (refer to section 7.3.3). The artefact behaviour specification specifies transformational environments in the design, refinement and phase-out phases of this life cycle in which the reference architecture for IKNs is required to function (refer to section 5.6.5.7). These environments, and the specific functional mode of the reference architecture in these phases of an IKN's life cycle, are as follows (refer to Figure 7.11) (1-5):

The reference architecture for IKNs must:

1. In the design phase, provide inputs for the design of a to-be architecture description when no as-is architecture description or instantiation is present.
2. In the refinement phase, provide inputs for the design of an operational as-is architecture description when only an as-is instance is present.

Figure 7.11: Transformations of the required implementation of the Zachman framework

3. In the refinement phase, assist in assessing the performance of an operational as-is architecture description by serving as a benchmarking tool.
4. In the refinement phase, provide inputs for the design of a refined to-be architecture description when a diagnosed, operational as-is architecture description is present.
5. In the phase-out phase, provide inputs for the design of a phased out to-be architecture description when an operational as-is architecture description is present.

The functional mode of the reference architecture for IKNs differs in the above environments as they unfold during the life history of an IKN. They are, however, all described as interactions with either *current* (as-is) or *future* (to-be) temporal *versions* of architectural artefacts. To accommodate this variation, reference models that form the content of the reference architecture should be transformation-agnostic.

No specific reference is made to short-term, medium-term or long-term transformations in this dimension of the required implementation of the Zachman framework. The type of information contained in the reference models along with the stakeholders (refer to sections 7.6.4 and 7.6.5), however, alludes to transformations that range between medium-term and long-term.

7.6.7 Detail

This section specifies that the required implementation of the Zachman framework should allow for varying levels of detail in each cell.

This dimension specifies the amount of detail and allows for levels with more information to be defined. A characteristic of this dimension is that between low, intermediate and high levels of detail all the information of the level with less detail is retained, with additional information being added. The goal of this dimension is therefore to omit details that are irrelevant or unknown in a particular context or moment in time.

Greefhorst et al. (2006) incorrectly remark that this is the intention of the vertical reification or perspectives dimension of the Zachman framework, thereby implying that the top row contains a low level of detail and the fifth row a high level of detail. While models in the first and fifth rows do indeed show large anatomical differences that may be interpreted as different levels of detail, *this is not the case*. Varying levels of detail, from low to excruciating, can be provided in each cell of the Zachman framework without changing the architectural perspective, thereby allowing for multiple models with varying levels of detail in each cell.

In order to comprehensively model the generalised patterns that form the reference models while maintaining usability of the artefact, multiple models with *varying levels* of detail should be included in cells of the required implementation of the Zachman framework.

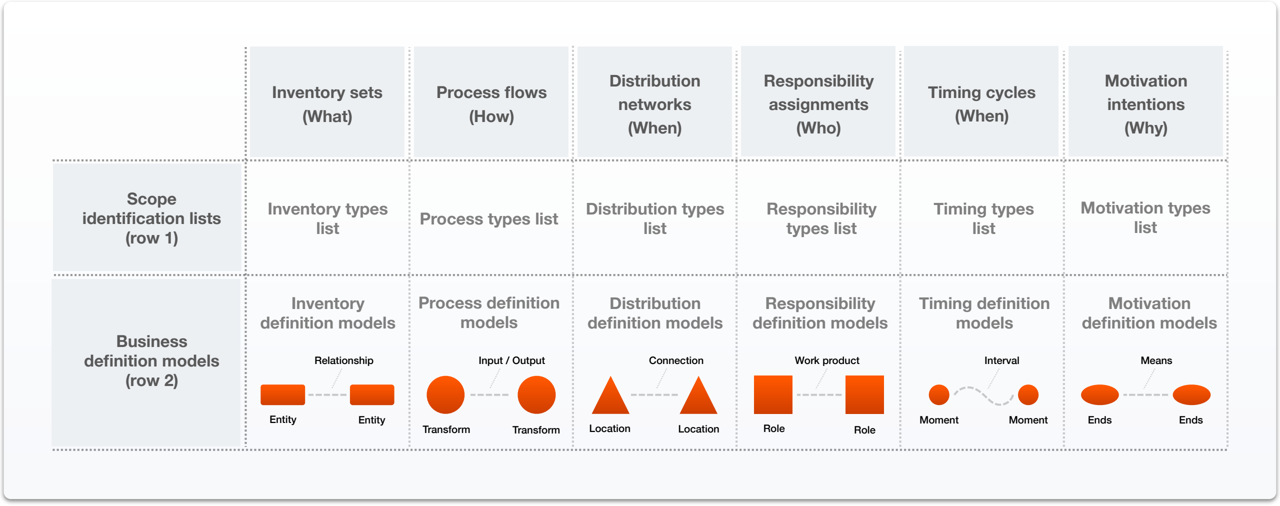
7.6.8 Representation

This section specifies representation formats for the reference models that populate the required implementation of the Zachman framework. The representation of second-row definition models with varied levels of detail is discussed at length.

This dimension specifies the way architectural information is presented in the artefact. Informal representations employ natural language, which leaves room for interpretation, while semi-formal representations use some form of common syntax to improve the definition of architectural information (Greefhorst et al. 2006).

The Zachman framework specifies that architectural information should be expressed *informally* in the first row, by creating primitive (single-variable) lists as a form of structured natural language. These lists identify the boundaries of the architecture description by identifying the types of inventories, processes, distribution, responsibilities, timing and motivations that exist in the architecture (refer to Figure 7.12). Hierarchical structures and indentations are used to embed the required different detail levels into the same language-based model (refer to section 7.6.7).

A *semi-formal* representation format is specified for the other cells of the framework through the implementation of meta-models that provide a representation language for each of the cells in the framework (refer to Figure 7.12). These meta-models are normalised to ensure that the framework can only produce primitive (single-variable) models and follow the basic relational structure of '*element-link-element*', with more appropriate labels for each of the cells. Provided that only business information is included in the



	Inventory sets (What)	Process flows (How)	Distribution networks (When)	Responsibility assignments (Who)	Timing cycles (When)	Motivation intentions (Why)
Scope identification lists (row 1)	Inventory types list	Process types list	Distribution types list	Responsibility types list	Timing types list	Motivation types list
Business definition models (row 2)	Inventory definition models Relationship Entity Entity	Process definition models Input / Output Transform Transform	Distribution definition models Connection Location Location	Responsibility definition models Work product Role Role	Timing definition models Interval Moment Moment	Motivation definition models Means Ends Ends

Figure 7.12: Representation of the required implementation of the Zachman framework

required implementation of the framework (refer to section 7.6.5), the second row meta-models for business definition models are of specific relevance.

The objective of the models in the second row of the Zachman framework is to define business concepts that fall within the architectural boundaries that were identified in the first-row lists. Hence, there exists a transformation from the identified types to defined concepts between the first and second rows. The framework provides an ‘*entity-relationship-entity*’ meta-model to define inventory concepts, ‘*transform-input/output-transform*’ for processes, ‘*location-connection-location*’ for distribution, ‘*role-work product-role*’ for responsibilities, ‘*moment-interval-moment*’ for timing, and ‘*ends-means-ends*’ for the definition of motivation concepts.

In section 7.6.7 multiple versions of business definition models with varying levels of detail were specified for each cell in the second row. These versions of the business definition models follow a structured increase in detail in each of the following cells:

1. *Basic business definition model* (low level of detail)
2. *Business concept definition models* (Intermediate level of detail)
3. *Extended business definition model* (high level of detail)

In the following sections, the meta-models provided by the Zachman framework are extended in a number of ways to assist in achieving the required variance in detail between these models. The characteristics of these meta-models and of each of the resulting model types are now discussed.

7.6.8.1 Basic business definition model

This section discusses the characteristics of the models that define business concepts in the second row at a low level of detail. One version of this model is created for every column of the artefact in section 8.5. While no example model is provided in this section, Figure 8.8 can be used as a reference.

The intention of the basic business definition model is to define an abstraction’s identified types (from the corresponding identification model in the first row) as concepts. This model therefore features a transformation from the top row *types* to the second row *concepts*. The model has a low level of detail and represents concepts visually according to the meta-model presented in Figure 7.13.

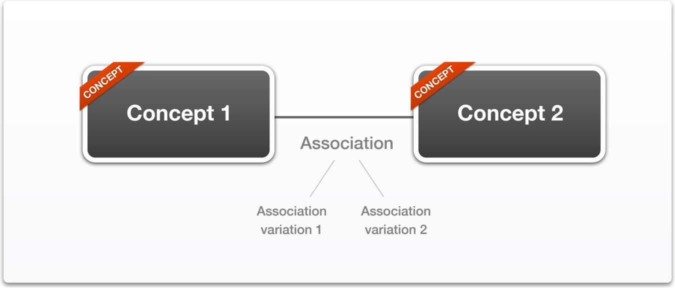


Figure 7.13: Meta-model for basic business definition models

The elements from the meta-models provided by the Zachman framework are labelled according to the various concepts and could be inventory sets, process flows, distribution networks, responsibility assignments, timing cycles, or motivation intentions, depending on the abstraction (column) to which the model belongs (refer to section 7.6.3). The associations between these concepts are also labelled appropriately, with variations in the nature of associations being possible depending on which concept is used as the basis for considering the association.

This low-detail business definition model can be seen as an aggregation of the more detailed elements and links that are defined in the business concept definition models (intermediate detail, refer to section 7.6.8.2) and extended business definition model (high detail, refer to section 7.6.8.3). All of the information in this model is therefore retained in the models with a higher level of detail.

7.6.8.2 Business concept definition models

This section discusses some of the characteristics of the models that define business concepts in the second row at an intermediate level of detail. Given that there are multiple concepts to define in each abstraction, multiple versions of these models are created for each column of the artefact in section 8.5, each focusing on a different business concept. While no example model is provided in this section, Figure 8.9 can be used as a reference.

Business concept definition models each define a single business concept in a given abstraction (column) at an intermediate level of detail from different points of view, through both textual and visual representations. The models add new information for each concept, while retaining all the information from the less detailed basic business definition model (refer to section 7.6.8.1).

In order to achieve the required increased level of detail in the business concept definition models, the relationships provided by the default Zachman framework meta-models (refer to Figure 7.12) are extended. Along with the associative relationships that are specified by the meta-models (refer to Figure 7.14) (1), two types of semantic relationships from the domains of knowledge representation and linguistics are added.

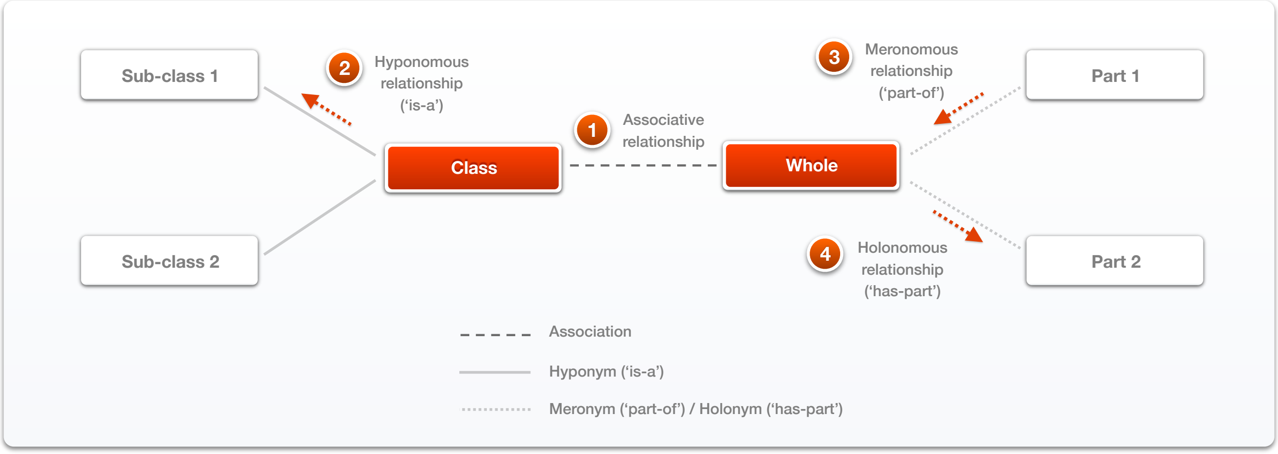


Figure 7.14: Relationships in the representation of the required implementation of the Zachman framework

The first is *hyponymy* (2), which indicates an '*is-a*' relationship between two concepts, e.g. red is a colour. In this relationship, concepts stand in the relation of class to sub-class (Brinton 2000). The second is *meronymy* (3) along with its opposite, *holonymy* (4)*,* which indicates a '*part-of/has-part*' relationship between two concepts where one is the whole and the other is the part (Cruse 1986; Brinton 2000). For example, in the case of a finger and a hand, a finger is a meronym (part) of a hand (whole), as it is a part of a hand. The hand (whole), however, is a holonym of the finger (part), as it has fingers as parts.

These semantic relationships do not violate the abstraction (column) meta-models, as they still produce primitive (single-variable) models. The semantic relationships also do not imply transformations to a different perspective (row) of the framework, and merely contribute the ability to include more detail in the definition of concepts in the second row. These semantic relationships are thus employed to produce the meta-model for business concept definition models in the required implementation of the Zachman framework.

This meta-model (refer to Figure 7.15) builds on the meta-model for basic business definition models (refer to Figure 7.13), as shown in the scope indication (refer to Figure 7.15) (1). The meta-model therefore reiterates that at the lowest level of detail business *concepts* may be *associated* with each other, and that

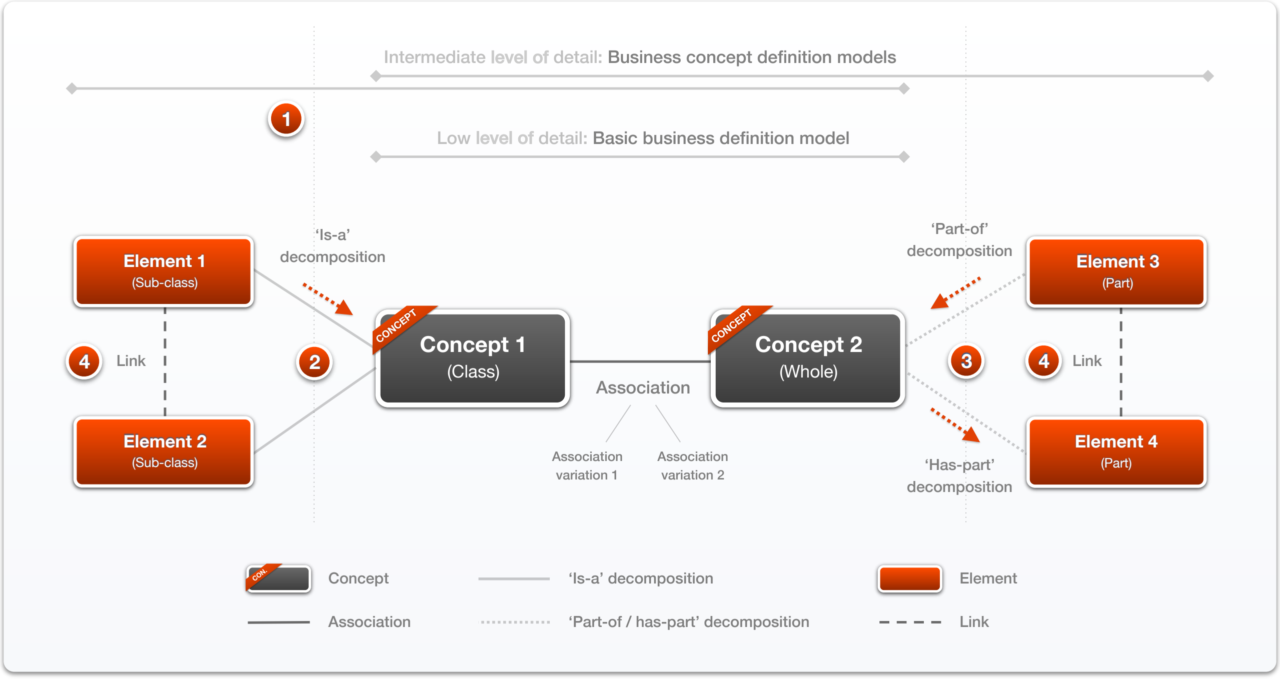


Figure 7.15: Meta-model for business concept definition models

more than one variation of such associations between concepts may exist. From there, each business concept definition model focuses on a single concept and its deconstruction, while keeping the concept's associations with other concepts in mind.

In order to assist in defining it, the concept in focus is deconstructed into its constituent *elements* via either 'is-a' (2) or 'part-of/has-part' (3) relationships. These elements represent entities, transforms, locations, roles, moments or ends, depending on the abstraction that the business concept definition model belongs to.

Elements relate to each other through links (4) that could be in the form of relationships, inputs/outputs, connections, work products, intervals or means, depending on the abstraction that the concept definition model belongs to.

As mentioned earlier, these definition models define business concepts from various points of view. Along with the above visual models, these points of view furthermore include a textual *definition* of the concept, along with descriptions of the concept's *deconstruction* into elements through either 'is-a' or 'has-part' relations. These elements are also defined textually in order to provide a better understanding of the main concept. Concepts are further defined in terms of their *associations* with other concepts in the same abstraction (column), as well as *integrations* with concepts from other abstractions (columns).

7.6.8.3 Extended business definition model

This section discusses the characteristics of the models that define business concepts in the second row at a high level of detail. One version of this model is created for every column of the artefact in section 8.5. While no example model is provided in this section, Figure 8.12 can be used as a reference.

The extended business definition model defines business concepts visually with a high level of detail within an abstraction (column). The model adds new information for each concept, while retaining all the information from the basic business definition model (low level of detail) and business concept definition models (intermediate level of detail).

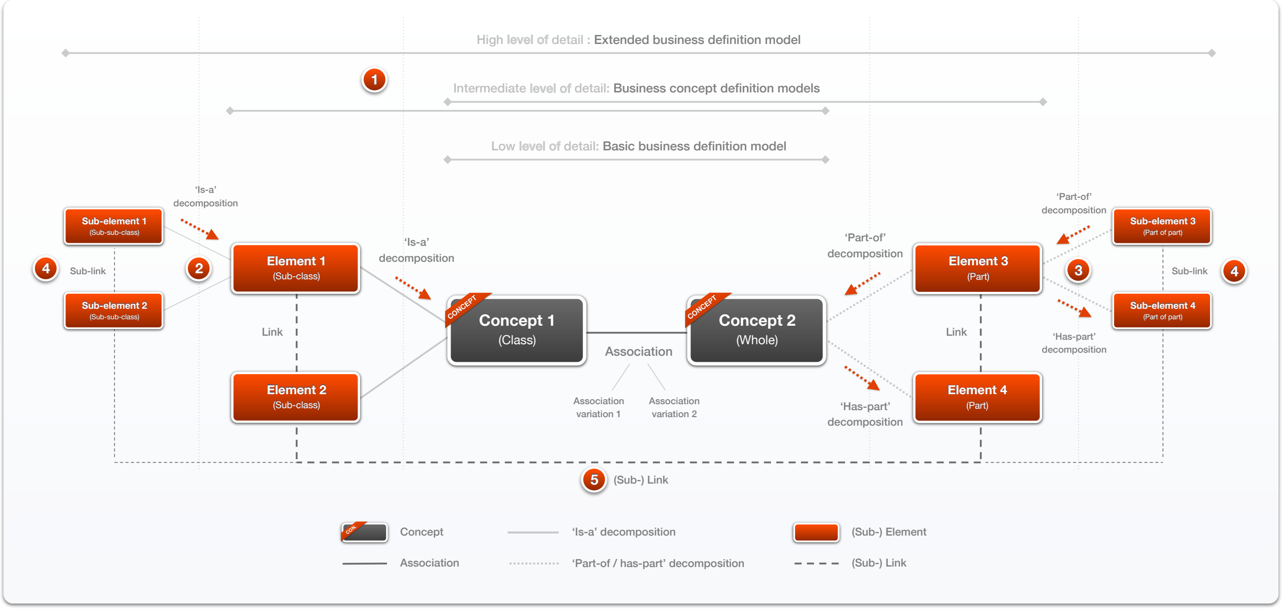


Figure 7.16: Meta-model for extended business definition models

The meta-model for extended business definition models builds on the meta-models for basic business definition models and business concept definition models, as shown in the scope indication (refer to Figure 7.16) (1). The extended business definition model, however, shows the deconstruction of all concepts in a given abstraction and does not only focus on a single concept.

In order to assist in defining concepts and also their constituent elements, elements may be deconstructed into constituent *sub-elements* via either 'is-a' (2) or 'part-of/has-part' (3) relationships. These sub-elements represent sub-entities, sub-transforms, sub-locations, sub-roles, sub-moments or sub-ends, depending on the abstraction that the concept definition model belongs to.

These sub-elements relate to each other through sub-links (4) that could be in the form of sub-relationships, sub-inputs/outputs, sub-connections, sub-work products, sub-intervals or sub-means, depending on the abstraction that the extended business definition model belongs to. Since extended business definition models do not only focus on a single concept, sub-elements could also have sub-links to sub-elements that do not originate from the same parent element or concept (5).

7.6.9 Compilation of artefact development roadmap

This section collates all the dimensional values that have been specified for the required implementation of the Zachman framework to form an artefact development roadmap.

By collating the various values that have been specified in the relevant architecture framework dimensions, it is possible to express the characteristics of an implementation of the Zachman framework that will produce a reference architecture. These values form the artefact development roadmap depicted in Table 7.1.

Table 7.1: Artefact development roadmap

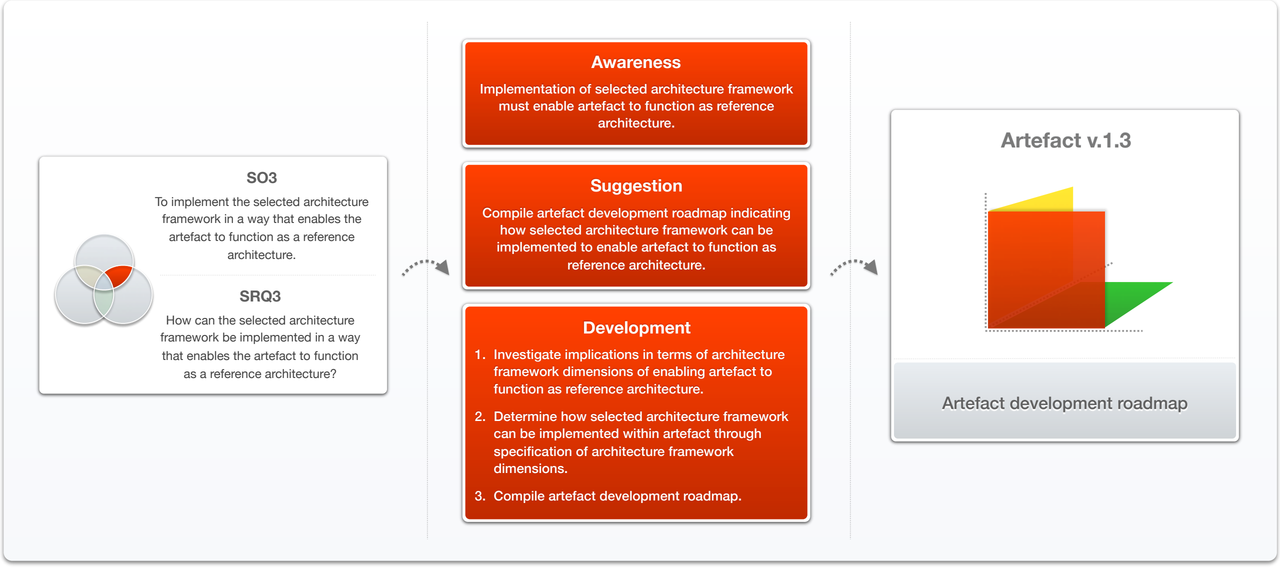
Dimension	**Value**	**Zachman framework implementation**
Initialisation phase		
1. Meta-level	Reference architecture	Basis for particular architecture descriptions
2. Nature	Reference models	Primitive reference models
3. Scope	Enterprise	Complete set (six columns)
4. Stakeholders	Primary (business) Secondary (technical)	Primary (rows 1 and 2) Secondary (rows 3 and 4)
5. Type	Business	Rows 1 and 2
6. Transformations	Versioned (current, future)	Multiple relevancies, transformation-agnostic
Population phase		
7. Detail	Varying levels	Multiple models in cells
8. Representation	Informal Semi-formal	Lists (row 1) Meta-models with semantic relationships (row 2)

For pragmatic development purposes, the values are rearranged and separated into two development phases, namely initialisation and population. In the initialisation phase an artefact skeleton is constructed in six steps with the specified characteristics in terms of meta-level (1), nature (2), scope (3), stakeholders (4), type of information (5) and transformations (6). Once this skeleton is initialised, the artefact is populated in two additional steps with content of varying levels of detail (7) that employ the specified representation formats (8).

In chapter 8 this artefact development roadmap is followed in the context of the engineering of IKNs to construct the reference architecture for IKNs.

7.7 Conclusion

This chapter presented the third cycle in the incremental design and development process of the reference architecture for IKNs. The cycle considered the compound requirement perspective that combines the technical and functional requirements (refer to the red intersection in Figure 7.17).



SO3

To implement the selected architecture framework in a way that enables the artefact to function as a reference architecture.

SRQ3

How can the selected architecture framework be implemented in a way that enables the artefact to function as a reference architecture?

Awareness

Implementation of selected architecture framework must enable artefact to function as reference architecture.

Suggestion

Compile artefact development roadmap indicating how selected architecture framework can be implemented to enable artefact to function as reference architecture.

Development

1. Investigate implications in terms of architecture framework dimensions of enabling artefact to function as reference architecture.
2. Determine how selected architecture framework can be implemented within artefact through specification of architecture framework dimensions.
3. Compile artefact development roadmap.

Figure 7.17: Conclusion of third design cycle

The design cycle addressed the third solution objective (SO3), and the investigation in this chapter was directed by the third secondary research question (SRQ3). This question asked how the selected architecture framework can be implemented to enable the artefact to function as a reference architecture.

The *awareness* of the problem echoed the solution objective and secondary research question above by stating that the implementation of the Zachman framework must enable the artefact to function as a reference architecture. The *suggestion* was subsequently made to compile an artefact development roadmap that documents how this required implementation of the Zachman framework can be achieved. The *development* of the artefact development roadmap was done through the assignment of appropriate values to relevant architecture framework dimensions (refer to Table 7.1). These values combine to articulate the required implementation of the Zachman framework from different points of view.

The roadmap is divided into two phases, namely an initialisation phase in which an artefact skeleton is constructed, and a population phase during which this skeleton is populated with content.

The first value to be assigned was to the meta-level dimension as this determined the objective of the implementation, i.e. to enable the artefact to function as a *reference architecture*. Other values were assigned to align with this meta-level value, the first being the nature of the architectural information in the implementation, which is specified as *primitive reference models.*

An architectural scope viewing the *network* as an *enterprise* is specified for the required implementation, and therefore the complete set of abstractions (six columns) provided by the Zachman framework are included in the roadmap.

A mapping of the perspectives (rows) of the Zachman framework to the IKN life cycle indicated which perspectives were prominent in the phases where the reference architecture is active. This led to primary and secondary audience values being allocated to the stakeholder dimension; effectively dividing the implementation into *business*-oriented (rows 1 and 2) and *technically* oriented (rows 3 and 4) layers. The type of architectural information was specified to be *business* information, thus delineating the research scope to the *top two rows* of the Zachman framework.

These models are to be *transformation-agnostic* and therefore serve as the basis for the development of versioned particular models that describe either *current* or *future* transformations. Multiple versions of these reference models with *varying levels of detail* are to be created for each cell in the top two rows of the framework. Reference models are expressed as *lists* (row 1) or within the representation format defined by the column *meta-models* (row 2), enriched with semantic relationships.

The search for an implementation of the Zachman framework that enables the artefact to function as a reference architecture represented the investigation of the *relationships between the views axis of the modelling framework and the axis indicating the meta-relationships between the artefact and its environment.* This is illustrated with the red plane of the modelling framework to the right of Figure 7.17.

The artefact development roadmap contributed by this chapter answers SQ3 and achieves SO3 by describing *an implementation of the Zachman framework that enables the artefact to function as a reference architecture*, and the second cycle is therefore circumscribed. The artefact development roadmap represents a third version of the reference architecture for IKNs (v.1.3).

The generated knowledge about the final artefact, gained through design decisions and packaged as an artefact development roadmap can now be used to update the requirement frame of reference and leads to the initiation of a next design cycle. The development of the artefact is concluded with a fourth design cycle in chapter 8 which develops the final version of the reference architecture for IKNs by following the artefact development roadmap compiled in this chapter.

8. Reference architecture for IKNs

8.1 Introduction

This chapter presents the fourth cycle in the incremental design and development process of the reference architecture for IKNs. The cycle considers the compound requirement perspective that combines technical and functional requirements, as well as scenarios (refer to the blue intersection in Figure 8.1).

The design cycle addresses the fourth solution objective (SO4) by indicating how an artefact can function as a reference architecture in the engineering of IKNs through the implementation of a suitable architecture framework. The search process in this chapter is directed by the fourth secondary research question (SRQ4), which asks how such an artefact can be developed.

The chapter therefore builds on the design of the artefact in chapters 5 (artefact behaviour specification), 6 (architecture framework selection) and 7 (artefact development roadmap). Its contribution is delivered in the form of a *reference architecture for IKNs as the final artefact*, and represents the investigation into the relationships between all three axes of the modelling framework (refer to section 5.6.2). This artefact achieves the research objective (RO) and therefore circumscribes (restricts) the activities of the design cycle, which leads to the evaluation of the artefact in Part 4 of the document.

The primary objective and associated primary research question that drives the design cycle is discussed in section 8.2. The design reasoning follows thereafter, with section 8.3 discussing the awareness that the need exists for a reference architecture in engineering of IKNs. However, the awareness that the desired behaviour, a suitable architecture framework and a development roadmap have been identified for such an artefact previously, is also expressed.

Section 8.4 makes the suggestion to develop a reference architecture for IKNs according to the artefact development roadmap by implementing the selected architecture framework to construct an artefact that adheres to the artefact behaviour specification.

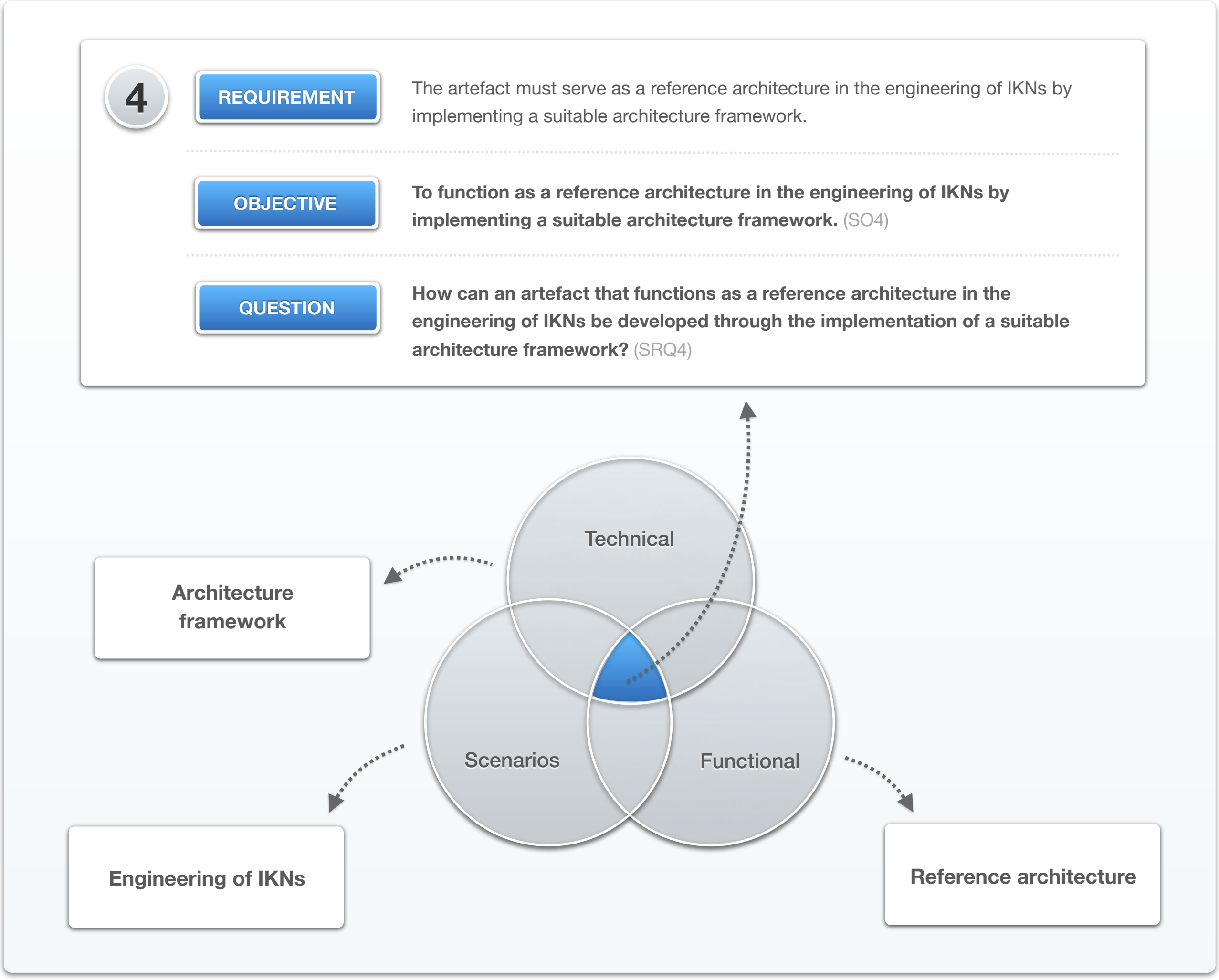
Section 8.5 documents the development of the next and final version of the reference architecture for IKNs, with both the initialisation and the population of the artefact with reference models being discussed. Section 8.6 concludes this chapter.

8.2 Design cycle drivers

This section presents the solution objective and secondary research question that drive this design cycle. This is done in order to position the design cycle (and chapter) in the context of the overall design and development work performed in Part 3 of the research design.

The design cycle is driven by SO4 and SRQ4 as derived by considering the blue requirements intersection of technical and functional requirements, as well as scenarios (refer to Figure 8.1).

Compound requirement:	The artefact must function as a reference architecture in the engineering of IKNs through the implementation of a suitable architecture framework.
Solution objective:	To function as a reference architecture in the engineering of IKNs through the implementation of a suitable architecture framework. (SO4)
Research question:	How can an artefact that functions as a reference architecture in the engineering of IKNs be developed through the implementation of a suitable architecture framework? (SRQ4)



The artefact must serve as a reference architecture in the engineering of IKNs by implementing a suitable architecture framework.

To function as a reference architecture in the engineering of IKNs by implementing a suitable architecture framework. (SO4)

How can an artefact that functions as a reference architecture in the engineering of IKNs be developed through the implementation of a suitable architecture framework? (SRQ4)

Figure 8.1: Drivers of fourth design cycle

8.3 Awareness

This section presents the awareness phase of the design reasoning in the fourth (blue) design cycle, and ties the design work done in the first three cycles into the fourth cycle.

Given SO4 and SRQ4 (refer to section 8.2), the awareness of the problem that is considered in this design cycle is stated as follows (refer to Figure 8.2):

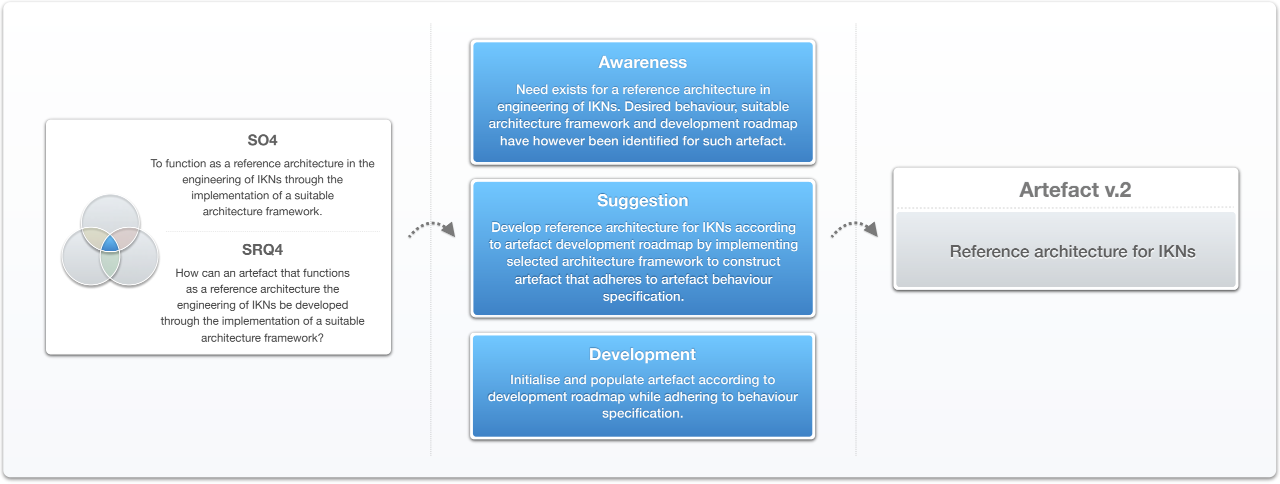


Figure 8.2: Reasoning in fourth design cycle

> **The need exists for a reference architecture in the engineering of IKNs. The desired behaviour, a suitable architecture framework and a development roadmap, however, have been identified previously for such an artefact.**

The problem awareness above articulates that the problem which is considered in the study can be solved through the development of a reference architecture for IKNs. This artefact will be developed in this fourth design cycle, building on the results of the previous three design cycles.

The desired behaviour of the artefact when interacting with its environment was specified in the first design cycle (green, refer to chapter 5). An architecture framework that suits this environment was selected in the second design cycle (yellow, refer to chapter 6). In the third design cycle (red, refer to chapter 7) an artefact development roadmap was compiled which shows how the selected architecture framework can be implemented to enable the artefact to function as a reference architecture.

By incorporating these previously designed components, this fourth design cycle (blue, refer to Figure 8.1) constructs an artefact that simultaneously addresses the entire requirement frame of reference (functional, technical and scenarios), and represents the convergent phase of the artefact design (refer to section 4.2.2).

8.4 Suggestion

This section presents the suggestion phase of the design reasoning in the fourth (blue) design cycle. It constitutes the artefact behaviour specification from the first (green) cycle and the development roadmap from the third (red) cycle as the guidelines for the development work in this cycle. The architecture framework selection from the second (yellow) cycle is also implicitly incorporated, as the roadmap is partly expressed in terms of the Zachman framework.

In light of the awareness of the problem presented in the previous section, the following suggestion for a possible solution is made (refer to Figure 8.2):

> **Develop a reference architecture for IKNs according to the artefact development roadmap by implementing the selected architecture framework to construct an artefact that adheres to the artefact behaviour specification.**

The artefact development roadmap already includes significant elements of the selected architecture framework, i.e. the Zachman framework (refer to Table 8.1). The objective of the development process is therefore to construct the reference architecture for IKNs by following the roadmap while ensuring that the artefact behaviour specification is adhered to.

Table 8.1: Artefact development roadmap

Dimension	Value	Zachman framework implementation
Initialisation phase		
1. Meta-level	Reference architecture	Basis for particular architecture descriptions
2. Nature	Reference models	Primitive reference models
3. Scope	Enterprise, network	Complete set (six columns)
4. Stakeholders	Primary (business), secondary (technical)	Primary (rows 1 and 2), secondary (rows 3 and 4)
5. Type	Business	Rows 1 and 2
6. Transformations	Versioned (current, future)	Multiple relevancies, transformation-agnostic
Population phase		
7. Detail	Varying levels	Multiple models in cells
8. Representation	Informal, semi-formal	Lists (row 1), meta-models with semantic relationships (row 2)

This development process starts with the initialisation phase of the artefact development roadmap during which the artefact skeleton is constructed. In this phase the artefact is configured in six steps according to the correct meta-level (step 1 in Table 8.1), i.e. as a reference architecture containing primitive reference models (step 2). The scope of this reference architecture is specified to describe an IKN as an enterprise (step 3). The reference architecture is aimed at its primary business audience and secondary technical audience (step 4). It is constrained to include only business information (step 5), while having an agnostic approach to current and future transformations (step 6). Throughout this initialisation phase, the artefact skeleton is set up in accordance with the relevant elements of the artefact behaviour specification, as it provides functional guidelines in terms of reference architecture usage, context and users.

In the population phase of the artefact development roadmap, the artefact skeleton is populated in two further steps with primitive reference models featuring varying levels of detail (step 7) and appropriate representation formats (step 8).

This development process produces an artefact that simultaneously addresses the entire requirement frame of reference, i.e. the engineering of IKNs, reference architecture functionality, as well as architecture framework implementation.

8.5 Development

This section presents the development phase of the design reasoning in the fourth (blue) design cycle and therefore documents the construction of the reference architecture for IKNs based on the artefact development roadmap and behaviour specification.

8.5.1 Artefact initialisation

The initialisation phase of the artefact development roadmap constructs an artefact skeleton in six steps (refer to Figure 8.3) by setting the artefact (1) meta-level, (2) nature, (3) scope, (4) type of information, (5) stakeholders and (6) transformations. Each of these steps is discussed in the following sections.

8.5.1.1 Meta-level

Step 1 of the artefact development roadmap: This section specifies that the artefact skeleton is set up to produce a reference architecture once it is populated. This is done according to the first step of the development roadmap (refer to section 7.6.1), while taking into account the context for reference architecture functions from the behaviour specification (refer to section 5.6.5.7).

The meta-level of the artefact is set to *reference architecture for IKNs*, i.e. as the basis for particular architecture descriptions, as illustrated in Figure 8.3 (1). The artefact therefore does not describe the architecture of any specific IKN, but describes generalised architectural patterns that are common to

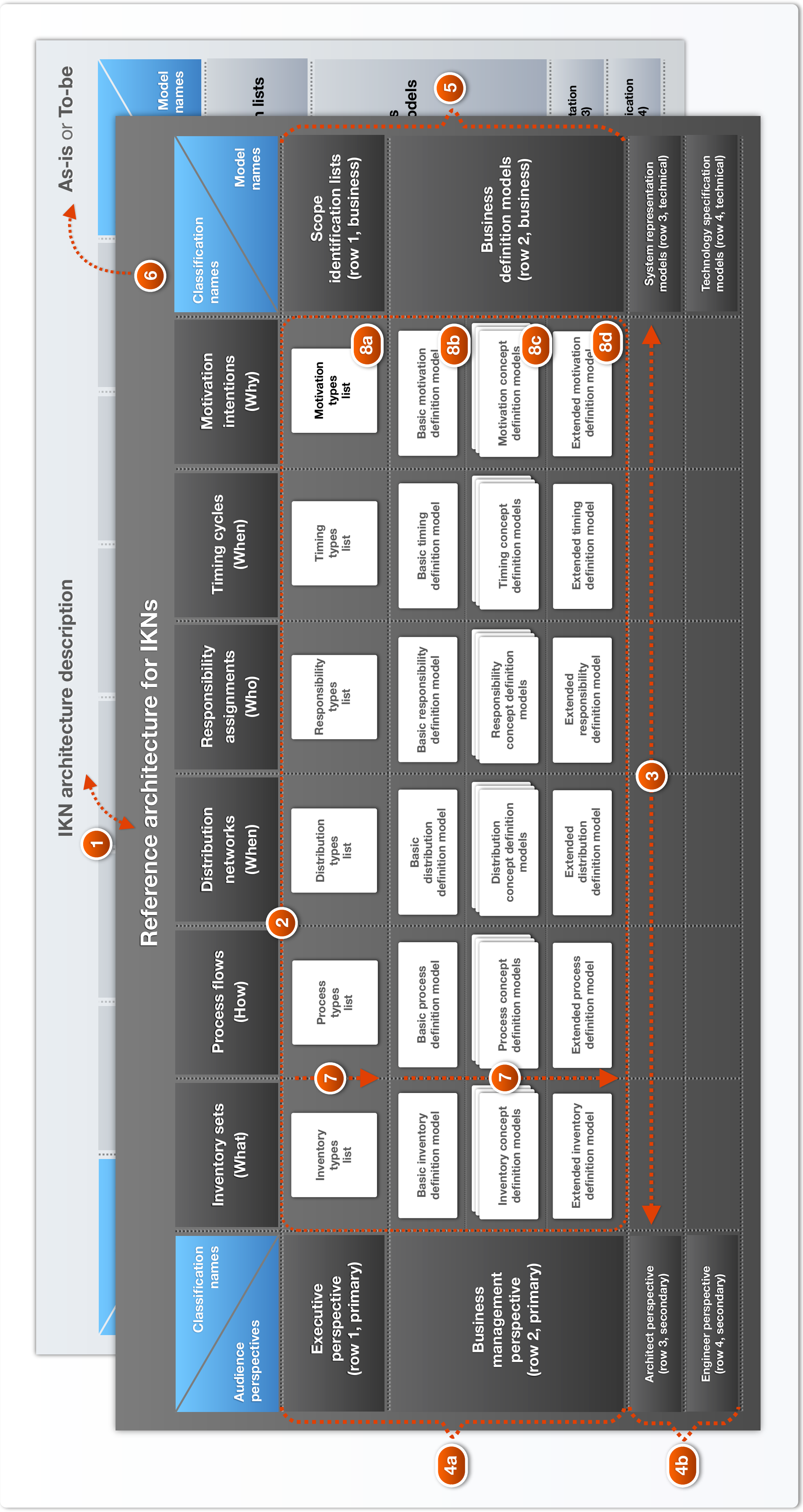


Figure 8.3: Artefact skeleton of reference architecture for IKNs

architectures of this class of enterprises. The genericity of the artefact, however, is limited to IKNs and the intention therefore is not to describe the architecture of CNOs in general. This freedom from implementation bias towards a particular IKN adheres to the artefact behaviour specification.

The artefact behaviour specification specifies that the artefact is to function as a reference architecture in the engineering of IKNs in the *design*, *refinement* and *phase-out* phases of their life cycle (refer to section 5.6.5.7). The artefact behaviour specification furthermore specifies five different contexts wherein this functionality is required:

1. Provide inputs for the design of a to-be architecture description when no as-is architecture description or instantiation is present.
2. Provide inputs for the design of an operational as-is architecture description when only an as-is instance is present.
3. Assist in assessing the performance of an operational as-is architecture description by serving as a benchmarking tool.
4. Provide inputs for the design of a refined to-be architecture description when a diagnosed, operational as-is architecture description is present.
5. Provide inputs for the design of a phased-out to-be architecture description when an operational as-is architecture description is present.

Amongst these contexts, the artefact behaviour specification identifies two *functional modes* for the reference architecture (refer to Figure 8.3) (1). The first mode involves the provision of inputs for the design of particular architecture descriptions, and the second mode is to assist in assessing an existing architecture description by serving as a benchmarking tool. These functional modes are achieved by including reference models that are agnostic of the artefact’s functional mode in a certain scenario, while ensuring that neither mode is prohibited by their content either. The functional modes of the artefact are therefore brought about through its interaction with its users in the specific scenario.

8.5.1.2 Nature

Step 2 of the artefact development roadmap: This section specifies the nature, and subsequently also the sources, of the architectural information that populates the artefact skeleton to construct the reference architecture for IKNs. It is based on the corresponding section in the development roadmap (refer to section 7.6.2) and aligns with the behaviour specification (refer to section 5.6.5.7).

8.5.1.2.1 Reference models

Given the meta-level of the reference architecture for IKNs (refer to section 8.5.1.1) and the desired behaviour of the artefact (refer to section 5.6.5.7), the artefact is populated with *reference models* (refer to Figure 8.3) (2). The objective of these models is to capture generalised patterns that exist in the architecture descriptions of IKNs and to represent them in such a way that they can be reused as the basis for the development of particular architectural models (Noran 2003; Cloutier et al. 2010). The intention of these models is therefore not to describe every possible combination of relationships that can feasibly be constructed in more specialised models.

These reference models represent generalised patterns and are seen as the starting point for the development of particular models. They are therefore not seen as limiting to the design of the particular models for specific IKNs. While the reference models are seen as a complete set at their specific level of abstraction, it is possible to extend the content of particular models beyond the content of the reference models to more accurately describe the architecture of a specific IKN.

The development and sources of the primitive reference models are discussed as part of the population of the reference architecture for IKNs in section 8.5.2.

8.5.1.3 Scope

Step 3 of the artefact development roadmap: This section specifies that the artefact skeleton includes all six columns of the Zachman framework classification schema in order to describe IKNs as enterprises. It is based on the corresponding section in the development roadmap (refer to section 7.6.3).

In order to produce a comprehensive architectural description of an IKN as an *enterprise*, the reference architecture includes the complete set of Zachman framework abstractions. The artefact skeleton therefore includes six columns that answer the six basic interrogatives regarding these *networks*, namely “What?”, “How?”, “Where?”, “Who?”, “When?” and “Why?” (refer to Figure 8.3) (3).

These abstractions produce descriptive representations labelled as follows:

- Inventory sets,
- Process flows,
- Distribution networks,
- Responsibility assignments,
- Timing cycles, and
- Motivation intentions.

Including the complete set of primitive (single-variable) abstractions in the reference architecture for IKNs allows for the construction of composite (multi-variable) models through integrative relationships. These composite models represent concepts in the way they would be encountered in the real world, i.e. as multi-variable compounds in the functioning *enterprise*.

8.5.1.4 Stakeholders

Step 4 of the artefact development roadmap: This section specifies a primary business-oriented audience and secondary technically-oriented audience for the artefact skeleton. It is based on the corresponding section in the development roadmap (refer to section 7.6.4) and aligns with the behaviour specification (refer to section 5.6.5.7).

The stakeholders engaging with the reference architecture for IKNs are designated as a *primary* audience that directly engages with the *business* layer of the artefact in the design, refinement and phase-out phases of the IKN life cycle. This primary audience equates to the stakeholders addressed by the top two perspectives or rows of the artefact (refer to Figure 8.3) (4a). These stakeholders are provided with reference models that serve as the basis for the particular architectural models they are aiming to develop for their particular IKNs.

The *secondary* audience of the artefact are stakeholders that engage with the business layer of the artefact through the alignment of their *technical* efforts. This secondary audience equates to the stakeholders addressed by the third and fourth rows of the artefact (refer to Figure 8.3) (4a). Unlike the primary audience of the artefact, these stakeholders are not provided with reference models. They are, however, directed to retain the reification process of the Zachman framework and produce particular technical models for their particular IKNs by transforming the business models developed by the primary audience.

This stakeholder specification for the reference architecture for IKNs is in line with the view on the users of the reference architecture provided by the artefact behaviour specification (refer to section 5.6.5.6).

8.5.1.5 Type of information

Step 5 of the artefact development roadmap: This section specifies that the artefact skeleton contains business information, organised into the top two rows of the Zachman framework classification schema. It is based on the corresponding section in the development roadmap (refer to section 7.6.5).

The type of information that is contained in the artefact is limited to *business* information. The reference architecture for IKNs is therefore limited to the *top two rows* of the Zachman framework, as illustrated by the presence of reference models in the top two rows of the artefact skeleton (refer to Figure 8.3) (5).

The models in the top row of the artefact identify the boundaries of the common architectural patterns that are found in the engineering of IKNs, and indicate the types of inventories, processes, distribution, responsibilities, timing and motivations that exist in the reference architecture. The models in the second row of the artefact define the business concepts that fall within these boundaries of the engineering of IKNs, as identified in the first-row models.

A content-based transformation therefore occurs between these rows – from the identified boundaries of the common architectural patterns, in the top row, to the definitions of concepts that constitute the common architectural patterns that fall within these boundaries, in the second row.

It should be noted that the models in the second row *define concepts in the context of the enterprise that is being modelled,* which in this case is IKNs. The definitions that are provided by these models are therefore working definitions for concepts, as they are understood from the generalised patterns found in the engineering of IKNs.

8.5.1.6 Transformations

Step 6 of the artefact development roadmap: This section specifies that the models that populate the artefact skeleton can produce both as-is and to-be versions of particular architecture descriptions for IKNs. It is based on the corresponding section in the development roadmap (refer to section 7.6.6) and it aligns with the behaviour specification (refer to section 5.6.5.7).

The artefact behaviour specification specifies five different time-based transformational environments in the life history of an IKN in which the reference architecture for IKNs is required to function (refer to section 8.5.1.1). It should be noted that these are time-based transformations, and should not be confused with the "transformations" that occur between the perspectives (rows) of the Zachman framework.

Although the exact functional mode of the reference architecture for IKNs differs in these environments, they can all be described as interaction with either *current* (as-is) or *future* (to-be) versions of architectural artefacts (refer to Figure 8.3) (6). The artefact skeleton is therefore configured to have *multiple relevancies*, i.e. providing inputs for, or assessing, a current (as-is) environment, as well as providing inputs for a future (to-be) environment, by including content that is *agnostic of time-based transformations* (refer to section 8.5.2).

8.5.2 Artefact population

The population phase of the artefact development roadmap constructs the reference architecture for IKNs in two concurrent steps (refer to Table 8.1) by populating the artefact skeleton with primitive reference models. These *primitive reference models represent the results of the qualitative systematic review* discussed in section 8.5.2.1 (refer to Appendix B for more detail and an example of this method). The primitive reference models exhibit varying levels of detail (refer to Figure 8.3) (7). The models are normalised and correspond to a specific abstraction (column), perspective (row) and detail level (position in cell) of the artefact skeleton and implement the corresponding representation format (refer to Figure 8.3) (8).

8.5.2.1 Development and sources of reference models

The results of a qualitative systematic review (refer to section 3.5.5.1) form the inputs for the development of the reference models that populate the artefact skeleton. These reference models are *primitive* (single-variable) models according to the Zachman framework classification schema, and can be combined to form *composite* (multi-variable) models. The implication of this, however, is that when data obtained from the reviewed sources provided architectural information that is of a composite nature, as is the case in most modelling environments, this information needed to be deconstructed into primitive elements.

Each source was therefore reviewed according to the six abstractions of the Zachman framework, i.e. *"What?"*, *"How?"*, *"Where?"*, *"Who?"*, *"When?"* and *"Why?"*. The studies listed in Table 8.2 represent the primary literary sources for this data collection method. These sources represent significant contributions to the body of knowledge on IKNs, with the full list of studies included in the review listed in Appendix A. An example of the review method, as well as the synthesis of the collected evidence from the example, is provided in Appendix B.

The evidence data from the review is graded by reference to the strengths and weaknesses of the evidence from the studies reviewed. Therefore, in Table 8.2, where a source briefly mentions a generalisable IKN construction component (i.e. a top row identification) that can be interpreted in terms of a particular abstraction, that abstraction is shaded in light blue (▨). This data is regarded as *identification data*. Where a source goes beyond a mere identification, and makes a significant contribution to the understanding of a generalisable IKN component (i.e. a second row definition) that can be interpreted in terms of a particular abstraction, that abstraction is shaded in darker blue (■). This data is regarded as *definition data*.

Table 8.2: Primary sources for reference models

Source	What	How	Where	Who	When	Why
Anklam (2007) Net Work	▨	■	▨	▨	■	■
Back et al. (2005) Putting Knowledge Networks Into Action	■	■	▨	▨	■	■
Berasategi et al. (2011) A comprehensive framework for collaborative networked innovation	▨	■	■	■	■	■
Bogers & West (2012) Managing Distributed Innovation: Strategic Utilization of Open and User Innovation	■	■	▨	■	▨	■
Bullinger et al. (2004) Managing innovation networks in the knowledge-driven economy	■	▨	■	■	▨	▨
Camarinha-Matos & Afsarmanesh (2008a) Collaborative Networks: Reference Modeling	■	■	■	■	■	■

Source	What	How	Where	Who	When	Why
Chesbrough & Prencipe (2008) Networks of innovation and modularity: a dynamic perspective	■	■		■	■	■
Chesbrough et al. (2006b) Open Innovation: Researching a New Paradigm	■	■	▨	■	▨	■
Cowan et al. (2007) Bilateral Collaboration and the Emergence of Innovation Networks	■	■	▨	▨	■	■
Dooley & O'Sullivan (2007) Managing within distributed innovation networks	■	■	▨	■	▨	▨
Douthwaite (2006) Enabling innovation: Technology-and system-level approaches that capitalize on complexity	■	▨		■	■	▨
Geels (2004) From sectoral systems of innovation to socio-technical systems	■			■	▨	■
Kruss et al. (2006) Creating knowledge networks	■	■	■	■	■	■
McPhee et al. eds. (2012) Living Labs	▨	■	▨	■		■
Nonaka et al. (2000) SECI, Ba and Leadership: a Unified Model of Dynamic Knowledge Creation	■	■		▨		■
Pittaway et al. (2004) Networking and innovation: a systematic review of the evidence	▨	■		■	▨	■
Du Preez & Louw (2008) A framework for managing the innovation process	▨	■	▨	■	■	■
Du Preez et al. (2008) A knowledge network approach supporting the value chain	■	■	▨	■	■	■
Schutte (2010) Executing innovation projects using the collaborative nature of integrated knowledge networks	■	■	▨	■	■	■
Swan et al. (1999) Knowledge management and innovation: networks and networking	■	■	▨	▨		■
Tidd & Bessant (2011) Managing Innovation	■	■	▨	▨	▨	■

The evidence data from the review was normalised according to the review questions, i.e. the abstractions of the Zachman framework, to make the results from different studies comparable. This enables the unification of concepts that are expressed in diverse ways in different studies. The complete set of evidence data from the review was then interpreted to identify normalised, generalisable architecture patterns and components that are significant to IKNs. These patterns and components form the basis of the identification models that populate the first row of the reference architecture for IKNs. The first row models, as well as definitions for the identified architecture components as provided by the data, were used as inputs for the development of the business concept definition models that populate the second row of the reference architecture for IKNs.

Amongst the sources in Table 8.2 are contributions, e.g. Camarinha-Matos and Afsarmanesh (2008a), which contain further generalised reference models that are specialised to match the range of the reference architecture for IKNs. Other sources, e.g. Kruss (2006), contain descriptions of IKN case studies, from which particular, i.e. more specialised, architecture descriptions are developed and generalised patterns identified. For the creation of the primitive reference models, data was employed as was available for each abstraction (column). For some abstractions, e.g. process flows (“How?”), more complete data were available than for others, e.g. distribution networks (“Where?”), as can be seen in the coverage of shading in Table 8.2.

8.5.2.2 Detail

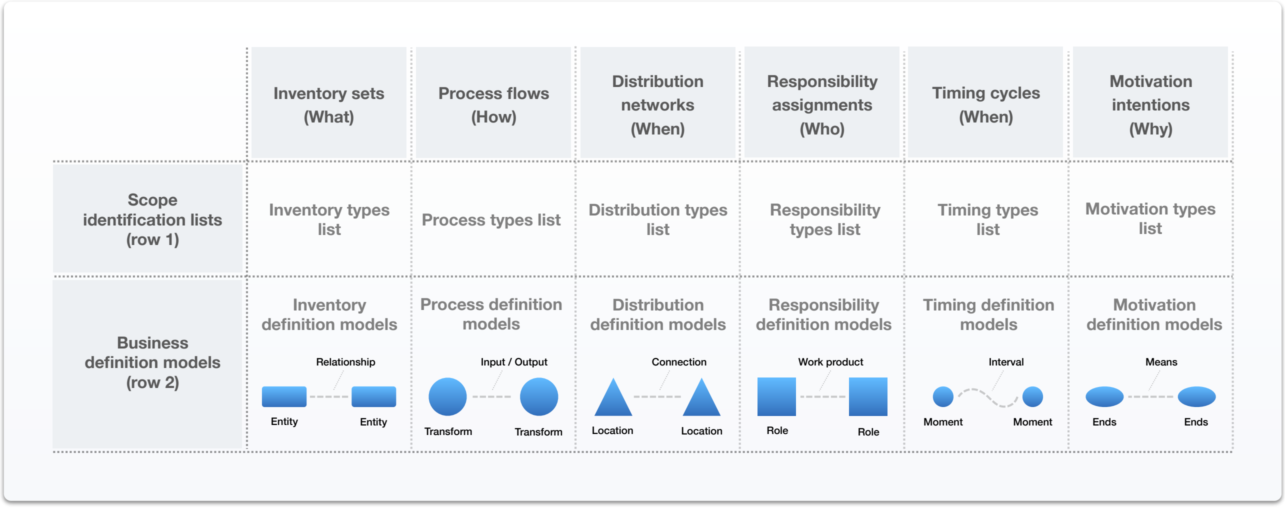
Step 7 of the artefact development roadmap: This section specifies that the artefact skeleton includes varying levels of detail in each cell. It is based on the corresponding section in the development roadmap (refer to section 7.6.7).

The reference architecture for IKNs is set up to include reference models with *varying levels of detail* in each cell (refer to Figure 8.3) (7). When moving between low, intermediate and high levels of detail, all the information of the level with less detail is retained as additional information is added. The intention is therefore to comprehensively model the generalised patterns that form the reference models while maintaining usability of the artefact by omitting details in certain versions of models.

8.5.2.3 Representation

Step 8 of the artefact development roadmap: This section specifies representation formats for the reference models that populate the artefact skeleton to construct the reference architecture for IKNs. It is based on the corresponding section in the development roadmap (refer to section 7.6.8), with emphasis on the second-row definition models.

The identification models in the first row of the reference architecture for IKNs (refer to Figure 8.3) (8a) employ structured natural language as an *informal* representation format to produce primitive (single-variable) models in the form of lists (refer to Figure 8.4). Hierarchical structures and indentations are used to introduce varying levels of detail in the lists (refer to section 8.5.2.2) thereby enhancing the resolution of the boundaries the models give to their particular abstraction of the reference architecture. The lists include



	Inventory sets (What)	Process flows (How)	Distribution networks (When)	Responsibility assignments (Who)	Timing cycles (When)	Motivation intentions (Why)
Scope identification lists (row 1)	Inventory types list	Process types list	Distribution types list	Responsibility types list	Timing types list	Motivation types list
Business definition models (row 2)	Inventory definition models	Process definition models	Distribution definition models	Responsibility definition models	Timing definition models	Motivation definition models

Figure 8.4: Representation in the reference architecture for IKNs

identification codes that allow for the tracing of identified items in the ensuing discussion of the model itself, as well as in second-row concept definition models.

A semi-formal representation format is employed in the second-row models (refer to Figure 8.3) (8b) through the implementation of meta-models that provide a representational language for each of the cells (refer to Figure 8.4). These meta-models follow the general structure of “*element-link-element*”. The reference architecture provides an “*entity-relationship-entity*” meta-model to define inventory concepts, ”*transform-input/output-transform*” for processes, ”*location-connection-location*” for distributions, “*role–work product-role*” for responsibilities, “*moment-interval-moment*” for timing, and “*ends-means-ends*” for the definition of motivation concepts. The ”business” prefix is appended to every element of these meta-models, given the type of information being described (refer to section 8.5.1.5).

In section 8.5.2.2, multiple versions of business definition models with varying levels of detail were specified for each cell in the second row. These versions of the business definition models follow a structured increase in detail in each of the cells. This increase can be indicated as follows:

1. *Basic business definition model* (low level of detail)
2. *Business concept definition models* (intermediate level of detail)
3. *Extended business definition model* (high level of detail)

The default meta-models provided by the Zachman framework are extended in a number of ways to assist in achieving the required variance in detail between these models. Some characteristics of each of these models are now reviewed before the models themselves are presented from section 8.5.3 onwards. Some practical notes on the presentation and discussion of the models are also provided where applicable.

8.5.2.3.1 Basic business definition model

This section reviews some of the characteristics of the models that define business concepts at a low level of detail. The derivation of these model characteristics is discussed in full in section 7.6.8.1 as part of the artefact development roadmap. One version of this model will be created for every column of the artefact skeleton.

The basic business definition model (refer to Figure 8.3) (8b) defines its abstraction's *types*, as identified in the corresponding identification model (first-row list), as *concepts* in the context of IKNs. These concepts may be inventory sets, process flows, distribution networks, responsibility assignments, timing cycles or motivation intentions, depending on the abstraction (column) in focus. This model has a low level of detail and represents concepts visually according to the part of the combined meta-model for definition models presented in Figure 8.5 which is indicated as falling within the scope of the basic business definition model.

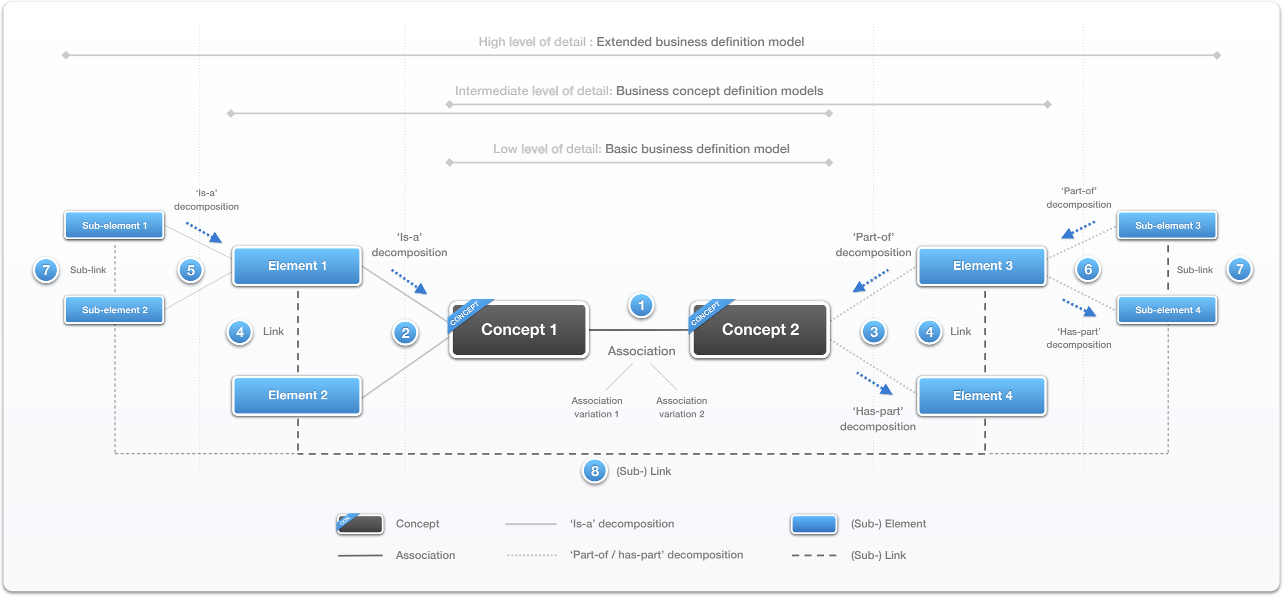


Figure 8.5: Meta-model for business definition models in reference architecture for IKNs

The concepts from the meta-model are labelled according to the various identified concepts, with the associations, including their variations, between concepts also labelled appropriately (refer to Figure 8.5) (1). While associations between concepts are viewed as bi-directional, association labels are mono-directional, and a single association between two components is therefore labelled differently from each point of view.

This low-detail model can be seen as an aggregation of the more detailed elements and links that are defined in the business concept definition models (intermediate level of detail) and extended business definition model (high level of detail). All of the information in this model is therefore retained in the models with a higher level of detail.

8.5.2.3.2 Business concept definition models

This section presents the characteristics of the models that define business concepts at an intermediate level of detail. The derivation of some of these characteristics is discussed in section 7.6.8.2 as part of the artefact development roadmap. Given that there are multiple concepts to define in each abstraction, multiple versions of these models will be created for each column of the artefact skeleton, each focussing on a different concept.

Business concept definition models each defines a single business concept in a given abstraction (column) at an intermediate level of detail from different points of view, through a combination of textual and visual representations. The models add new information for each concept, while retaining all the information from the less detailed basic business definition model.

The various points of view employed to define concepts include a textual *definition* of the concept. These definitions are not referenced, since they represent definitions that were developed for primitive concepts, while definitions found in literature commonly refer to composite concepts. Along with these textual definitions, descriptions of the concept's *deconstruction* into elements and sub-elements through either "is-a" or "has-part" relationships are provided. These elements are also defined textually in order to provide a better understanding of the main concept. Concepts are further defined in terms of their *associations* with other concepts in the same abstraction. *Integrations* with concepts in other abstractions are also presented to indicate common patterns in how composite (multi-variable) models may be assembled from primitive (single-variable) elements, as considering these integrations assist in defining concepts. This, however, does not imply that the business concept definition models themselves are composites, as they remain strictly primitive in their nature.

These textual definitions are accompanied by visual representations of the concept definitions that indicate their deconstruction into elements, as well as their intra-abstraction associations and links. These visual models adhere to the part of the combined meta-model for definition models presented in Figure 8.5 which is indicated as falling within the scope of business concept definition models.

The meta-model reiterates that business *concepts* may be *associated* with each other at the lowest level of detail, and that more than one variation of such associations between concepts may exist (refer to Figure 8.5) (1). Therefore, each business concept definition model focuses on a single concept and its deconstruction, while keeping the concept's associations with other concepts in mind.

In order to assist in defining it, the concept in focus is deconstructed into its constituent *elements* via either "is-a" (refer to Figure 8.5) (2) or "part-of/has-part" (3) relationships. These elements represent entities, transforms, locations, roles, moments or ends, depending on the abstraction to which the business concept definition model belongs to. Elements relate to each other through links (4) that could be in the form of relationships, inputs/outputs, connections, work products, intervals or means, depending on the abstraction that the concept definition model belongs to.

Emphasis is placed on the outbound components of the bi-directional associations or links between concepts or elements, as these are seen as most beneficial to the definition of the concept or element. The visual versions of the models are to use the line styles for associations and links that are indicated in the legend accompanying Figure 8.5.

Note that in the discussion of some models, particularly those in the process and responsibility abstractions, the links (i.e. inputs/outputs or work products) that link elements (i.e. transforms or roles) may resemble elements from other abstractions, e.g. inventory entities. The intention with the use of these links that stem from other abstractions to link elements in the current abstraction, is to refer to the potential use of these elements, and not to the elements themselves, as that would constitute a composite model.

For instance, references to contextualised explicit and tacit knowledge as inputs/outputs between the constituent transforms of the knowledge creation and transfer process flow (refer to section 8.5.4.4.2) do not refer to the explicit and tacit knowledge entities identified and defined in sections 8.5.3.1 and 8.5.3.4.1. That would imply a composite model, which is not the objective, and references to explicit and tacit knowledge in those cases are therefore intentionally not printed in italics.

Where the discussion of a transform, however, can be illuminated by a reference to a potential formal integration with an inventory entity, such a reference, however, will be printed in italics in accordance with the convention used in other sections of this chapter.

8.5.2.3.3 Extended business definition model

This section reviews some of the characteristics of the models that define business concepts at a high level of detail. The derivation of these model characteristics is discussed in full in section 7.6.8.3 as part of the artefact development roadmap. One version of this model will be created for every column of the artefact skeleton.

The extended business definition model (refer to Figure 8.3) (8d) defines business concepts in the context of the engineering of IKNs with a high level of visual detail and represents concepts visually according to full meta-model for definition models presented Figure 8.5. The model therefore adds new information to each concept, while retaining all the information from the basic business definition model (low level of detail) and business concept definition models (intermediate level of detail). The extended business definition model, however, shows the deconstruction of all concepts in a given abstraction, and does not only focus on a single concept.

In order to assist in defining concepts and also their constituent elements, elements may be deconstructed into constituent *sub-elements* via either “is-a” (refer to Figure 8.5) (5) or “part-of/has-part” (6) relationships. These sub-elements represent sub-entities, sub-transforms, sub-locations, sub-roles, sub-moments or sub-ends, depending on the abstraction that the extended business definition model belongs to.

These sub-elements relate to each other through sub-links (refer to Figure 8.5) (7) that could be in the form of sub-relationships, sub-inputs/outputs, sub-connections, sub-work products, sub-intervals or sub-means, depending on the abstraction that the extended business definition model belongs to. Since extended business definition models do not only focus on a single concept, sub-elements could also have sub-links to sub-elements that do not originate from the same parent element or concept (8).

Where the high level of detail of the extended business definition model allows for more accurate indication of association variations than was possible in the models with a lower level of detail, these detailed associations are indicated. In all cases, the models are to use the line styles for relationships that are indicated in the legend accompanying Figure 8.5.

However, it should be noted that, although it is possible to create even more detailed definition models than this extended version, the model *only indicates elements, sub-elements, links and sub-links that are essential to the definition of the concepts*. In even more detailed models, further relationships between concepts could be identified and indicated without transforming the model to a different perspective, i.e. a third-row system representation model.

The decision to only include links that are essential to the definition of concepts is in line with the nature of the architectural information in the artefact skeleton, which was specified to be *reference models* in section 8.5.1.2. The intention of these models is to describe common patterns amongst concepts, and not to describe every feasible combination of links.

The deconstruction of elements into sub-elements (refer to Figure 8.5) (5, 6) is furthermore only done for models where a generalised reference model is possible at that level of detail. In cases where this is not done, the extended business definition model, however, does still provide an overview of the associations, deconstruction, and links of all concepts in the particular abstraction (column).

8.5.2.3.4 Visual cues and model numbering

The visual cues shown in Figure 8.6 will be provided uncaptioned with every reference model presented from section 8.5.3 onwards to clarify the model’s role and placement in the artefact skeleton.

A number is allocated to every model represents a concatenation of the model’s placement in a certain column, row and detail level. In the case of business concept definition models (intermediate detail level in the second row, refer to Figure 8.3 (8c)), the number of the concept which is identified is also added. The

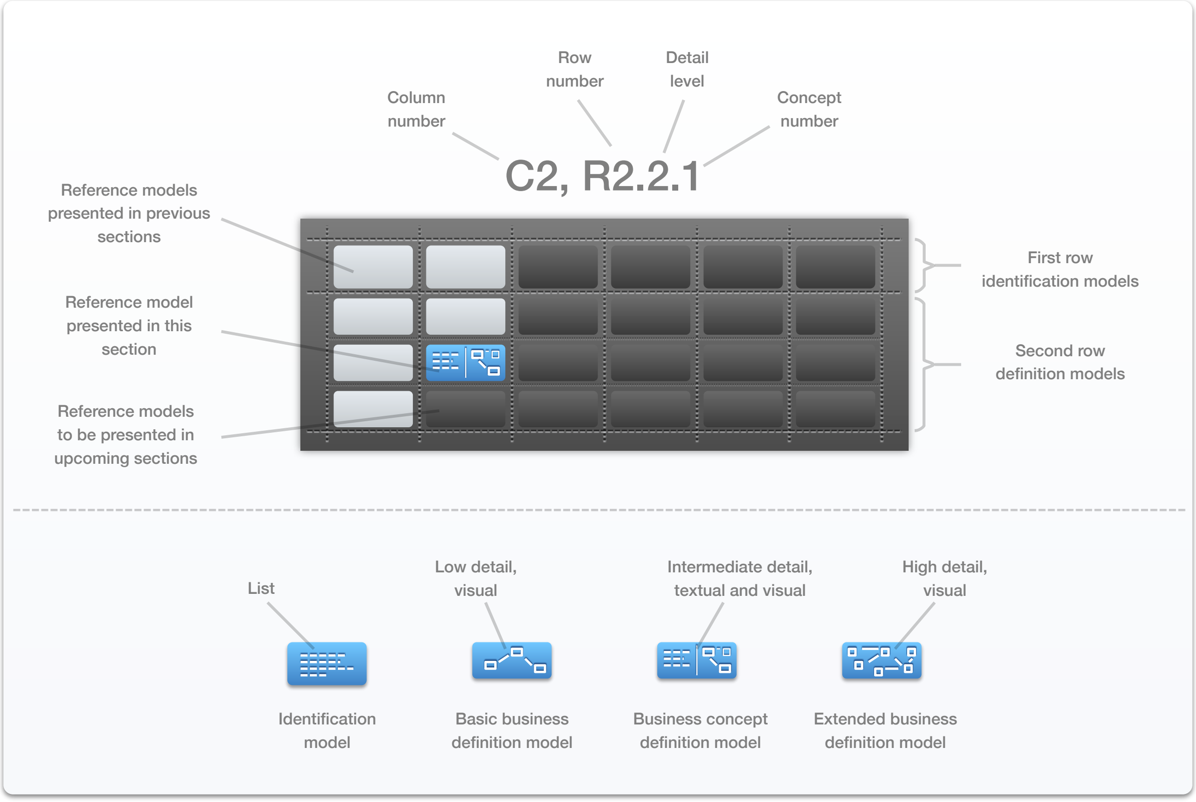


Figure 8.6: Visual cues for reference models in the reference architecture for IKNs

model's placement in either the first or second row (with its various detail levels) is shown, as well as the model's placement in the artefact with respect to previously discussed and upcoming models. An indication of the detail level and representation format that can be expected for the particular model is also provided.

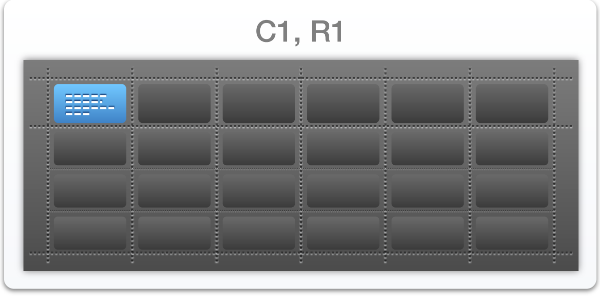
The primitive reference models that populate the reference architecture for IKNs (based on the sources and analysis method discussed in section 8.5.2.1) are now presented per abstraction (column). The specific representation format or meta-model implemented in the models for the respective abstractions is also discussed.

8.5.3 Inventory set models

This section discusses the inventory abstraction (first column) of the artefact, and the models presented therefore answers the interrogative "What?" in the engineering of IKNs. The top-row reference model is presented and motivated first. Thereafter the meta-model for the second row of the first column is presented, followed by the various second-row reference models according to increasing levels of detail. Where the description of a model includes a reference to another formalised concept in the reference architecture, the name of the concept is printed in italics.

8.5.3.1 Inventory identification model

The inventory identification model (C1, R1; refer to Table 8.3) lists inventory types that are significant and common to the engineering of IKNs. This model is based on the interpretation of the *identification data* from the qualitative systematic review described in section 8.5.2.1, with regard to the "*What?*" review question.



At the lowest level of detail, three main inventory types are identified, namely *knowledge*, *innovation artefacts* and *resources*. Constituent inventory sub-types are also identified that classify these types at a higher level of detail.

Table 8.3: Inventory identification model (C1, R1)

Code	Inventory type	Code	Constituent inventory sub-types
Inv-1	**Knowledge**	Inv-1.1	Information artefacts
		Inv-1.2	Explicit knowledge
		Inv-1.3	Tacit knowledge
Inv-2	**Innovation artefacts**	Inv-2.1	Ideas
		Inv-2.2	Concepts
		Inv-2.3	Prototypes
		Inv-2.4	Projects
		Inv-2.5	Products
		Inv-2.6	Services
Inv-3	**Resources**	Inv-3.1	Information systems

Code	Inventory type	Code	Constituent inventory sub-types
		Inv-3.2	Workspaces
		Inv-3.3	Equipment
		Inv-3.4	Funding

Knowledge is a key inventory type for IKNs, given their knowledge-sharing purpose (refer to section 2.5.4). There also exist deep links between the *knowledge creation and transfer* process where these *knowledge* entities are created and manipulated, and the *innovation* process. The *knowledge* inventory type is deconstructed into inventory sub-types of *information artefacts*, *explicit knowledge* and *tacit knowledge*.

Innovation artefacts are the productive outputs of IKNs, and their generation is closely linked to the *innovation intention* and purpose of IKNs. *Innovation artefacts* are produced and transformed during the *innovation* process, which is one of the distinguishing features of IKNs. The *innovation artefacts* type is broken down into inventory sub-types of *ideas*, *concepts*, *prototypes*, *projects*, *products* and *services*.

Resources are required for the successful operation of IKNs, especially during the *innovation* process, and the sharing of *resources* amongst *network stakeholders* is a key component of the *innovation strategy* of IKNs. The *resources* type is further classified into *information systems*, *workspaces, equipment* and *funding* as inventory sub-types.

Definitions for the inventory types and sub-types identified in the inventory identification model are provided in the second-row inventory definition models presented in sections 8.5.3.3, 8.5.3.4 and 8.5.3.5.

8.5.3.2 Meta-model for definition models in inventory abstraction

The primitive meta-model presented in Figure 8.7 serves as the semi-formal representational language for the various second-row definition models in the inventory abstraction, or "*what*" column, of the reference architecture for IKNs.

This meta-model assists in defining *inventory sets* as concepts and follows the single-variable structure of '*entity-relationship-entity*' amongst its elements, links, sub-elements and sub-links. The meta-model therefore deconstructs and represents inventory sets into entities and sub-entities, along with their respective relationships and sub-relationships.

This representation is done at various levels of detail in the basic inventory definition model (refer to section 8.5.3.3), inventory set definition models (refer to section 8.5.3.4) and extended inventory definition model (refer to section 8.5.3.5). For all of these models, the meta-model scope indication shown at the top of Figure 8.7 applies.

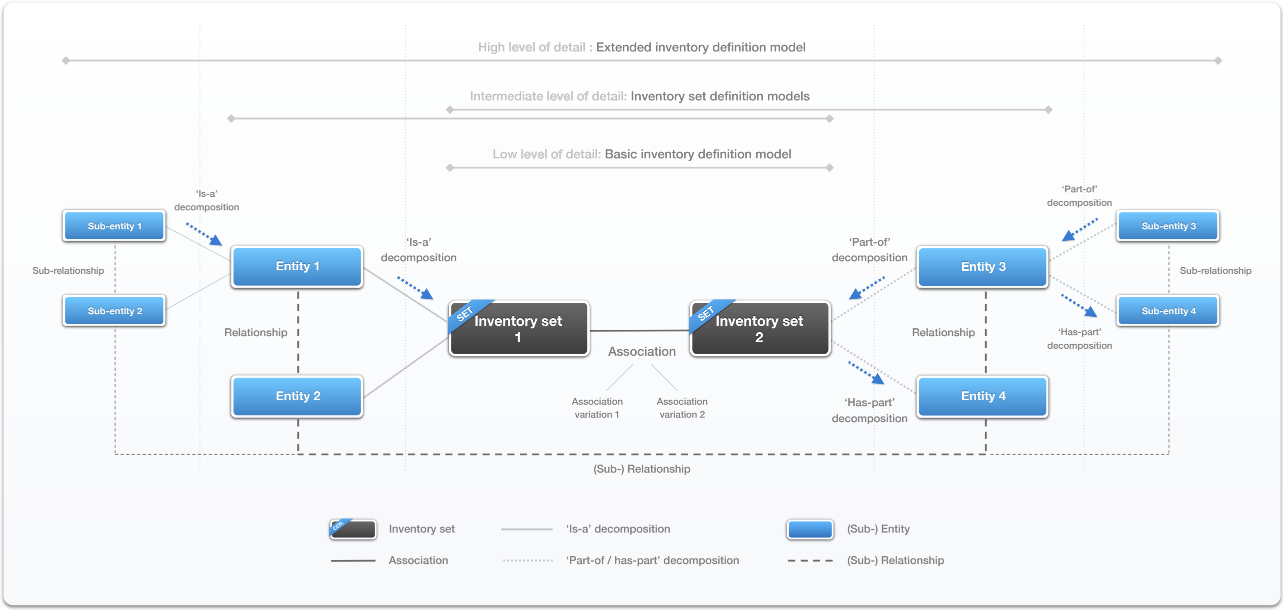
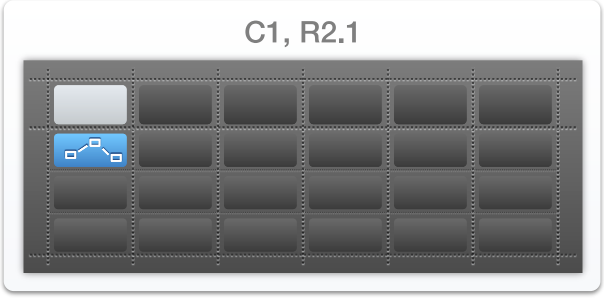


Figure 8.7: Meta-model for definition models in inventory abstraction

8.5.3.3 Basic inventory definition model



The basic inventory definition model (C1, R2.1) defines the inventory sets that were identified as inventory types in the inventory identification model (C1, R1; refer to Table 8.3) at a low level of detail by representing them visually according to the abstraction meta-model (refer to Figure 8.7). This model is based on the interpretation of the *definition data* from the qualitative systematic review described in section 8.5.2.1, with regard to the "*What?*" review question.

The model depicts the various identified inventory sets, as well as the associations between them (refer to Figure 8.8). The model indicates that *knowledge* plays a role in the production and documentation of *innovation artefacts*. The model further indicates that *knowledge* associates with *resources* through references to specific resources in the content of knowledge entities. *Innovation artefacts* associate with *knowledge* through references to existing *knowledge* entities, as well as the contribution of new *knowledge* entities, through their content. *Innovation artefacts* associate with *resources* by utilising them. *Resources* associate with *knowledge* and *innovation artefacts* through various kinds of support for their development and management.

These inventory sets are now defined in more detail in the applicable inventory set definition models in section 8.5.3.4.

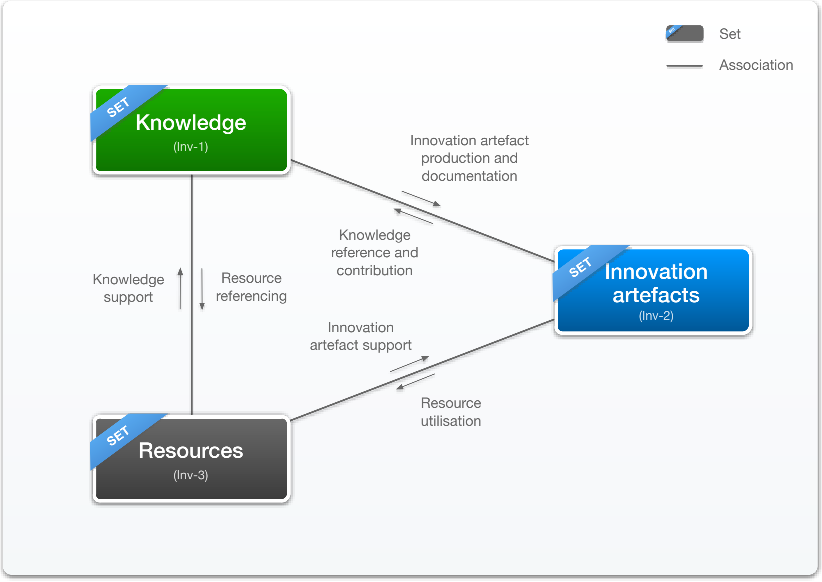


Figure 8.8: Basic inventory definition model (C1, R2.1)

8.5.3.4 Inventory set definition models

The inventory set definition models presented in this section each defines an identified inventory set at an intermediate level of detail. The models therefore add new information for each set, while retaining all the relevant information from the basic inventory definition model (refer to Figure 8.8). Both textual and visual versions are provided to assist in defining the sets. Deconstructions, relationships and integrations between concepts are expressed in terms of identification codes in side-notes for traceability purposes.

The models presented in this section are based on the interpretation of the *definition data* for each identified inventory set, as obtained from the qualitative systematic review described in section 8.5.2.1.

8.5.3.4.1 Knowledge

This inventory set definition model (C1, R2.2.1) defines *knowledge* (Inv-1) as an inventory set at an intermediate level of the detail. The textual version of this model is presented in Table 8.4 and the visual version in Figure 8.9.

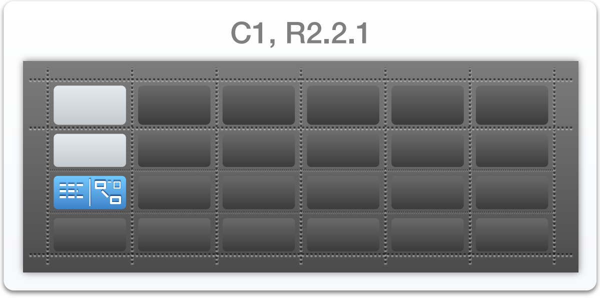


Table 8.4: Inventory set definition model – Knowledge (C1, R2.2.1)

	Code	Name / definition
A. Concept	**Inv-1**	**Knowledge**
B. Definition		Facts, information, understanding, skills, values and beliefs produced and acquired through experience or education.
C. Deconstruction	Inv-1.1	Information artefacts
	Inv-1.2	Explicit knowledge
	Inv-1.3	Tacit knowledge
D. Associations	Inv-2	Innovation artefacts
	Inv-3	Resources
E. Integrations	Proc-2	Knowledge creation and transfer
	Dist-2.1	Network stakeholder locations
	Resp-3	Knowledge owners
	Tim-3	Knowledge life cycle
	Mot-3.2	Network domain and purpose

The *knowledge* (Inv-1) inventory set is **defined** (refer to Table 8.4) (B) as:

> *Facts, information, understanding, skills, values and beliefs produced and acquired through experience or education.*

The *knowledge* inventory set is **deconstructed** (refer to Table 8.4 (C), Figure 8.9 (1)) into three constituent entities, namely *information artefacts*, *explicit knowledge* and *tacit knowledge*, via "is-a" relationships.

Inv-1 →
Inv-1.1, Inv-1.2,
Inv-1.3

Information artefacts (Inv-1.1) are:

> *Physical containers of either hard or soft copy that contain information, e.g. a printout of a budget report as well as its digital version.*

Information artefacts are the physical containers for the representation of codified *explicit knowledge* entities, and are the most concrete entity in the *knowledge* inventory set.

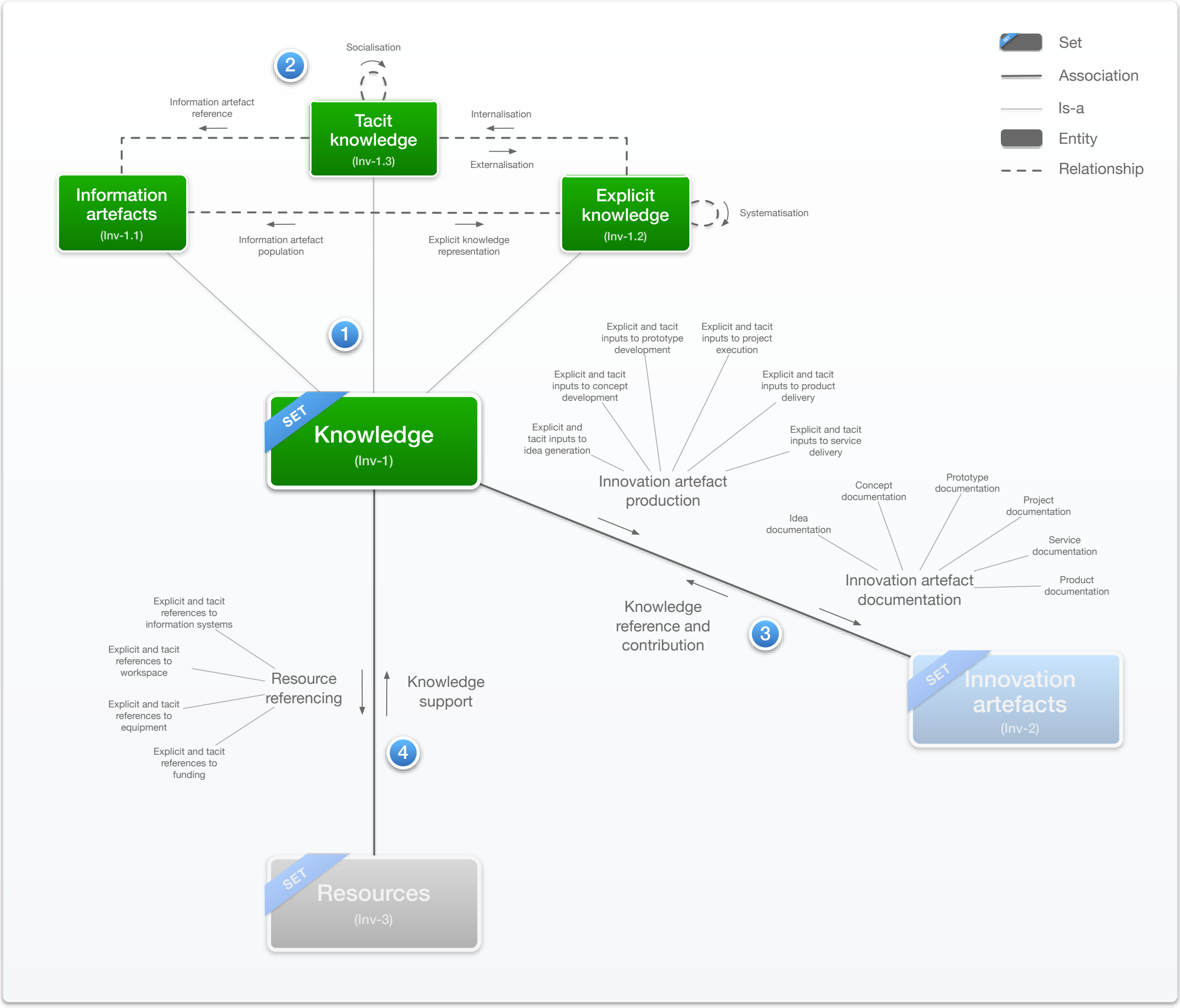


Figure 8.9: Inventory set definition model – Knowledge (C1, R2.2.1)

Explicit knowledge (Inv-1.2) is:

> *Knowledge that can be expressed in formal and systematic language and shared in the form of data, scientific formulae, specifications, manuals and such like. It can be processed, transmitted and stored relatively easily.*

Explicit knowledge can be separated from its origin (knower) and be used to populate information artefacts with comparative ease. Explicit knowledge is transformed into further explicit knowledge through the systematisation process and into tacit knowledge through internalisation. Explicit knowledge provides inputs to the production of innovation artefacts and also documents these inventory entities. The content of explicit knowledge entities may also contain references to various resources.

Tacit knowledge (Inv-1.3) is:

> *Knowledge that is highly personal and hard to formalise. Subjective insights, intuitions and hunches fall into this category of knowledge. Tacit knowledge is deeply rooted in action, procedures, routines, commitment, ideals, values and emotions. It 'indwells' in a comprehensive cognisance of the human mind and body. It is difficult to communicate tacit knowledge to others, since it is an analogue process that requires a kind of 'simultaneous processing'.*

Tacit knowledge is the most abstract entity of the *knowledge* inventory set. *Tacit knowledge* is transformed into further *tacit knowledge* through the *socialisation* process and into *explicit knowledge* through *externalisation*. *Tacit knowledge* provides inputs for the production of *innovation artefacts*, and the content of *tacit knowledge* entities includes references to *information artefacts* and various *resources*.

Based on the preceding discussion, the constituent entities of the *knowledge* (Inv-1) inventory set share the following relationships (refer to Figure 8.9) (2):

• The interaction of *explicit* and *tacit knowledge* through externalisation and internalisation.	Inv-1.2 ↔ Inv-1.3
• The interaction of *tacit knowledge* entities through socialisation.	Inv-1.3 → Inv-1.3
• The interaction of *explicit knowledge* entities through systematisation.	Inv-1.2 → Inv-1.2
• References to *information artefacts* in the content of *tacit knowledge* entities.	Inv-1.3 → Inv-1.1
• The population of *information artefacts* with *explicit knowledge*.	Inv-1.2 → Inv-1.1
• The capture and representation of *explicit knowledge* in *information artefacts*.	Inv-1.1 → Inv-1.2

Knowledge (Inv-1) and its constituent entities also show **associations** (refer to Table 8.4) (D) with the following sets in the inventory abstraction (column):

- *Innovation artefacts* (Inv-2; refer to Figure 8.9 (3)):

• The provision of both *tacit* and *explicit knowledge* entities that serve as inputs for the production of *innovation artefacts*, e.g. *ideas*, *concepts*, *prototypes, projects, products* and *services.*	Inv-1.2, Inv-1.3 → Inv-2.1, Inv-2.2, Inv-2.3, Inv-2.4, Inv-2.5, Inv-2.6
• The documentation of *innovation artefacts* as *explicit knowledge*, e.g. *idea* documentation, *concept* documentation, *prototype* documentation, *project* documentation, *product* documentation and *service* documentation.	Inv-1.2 → Inv-2.1, Inv-2.2, Inv-2.3, Inv-2.4, Inv-2.5, Inv-2.6
• References and contributions to *explicit* and *tacit knowledge* entities in the content of *innovation artefacts* (for more detail refer to section 8.5.3.4.2 on *innovation artefacts*).	Inv-2 → Inv-1.2, Inv-1.3

- *Resources* (Inv-3; refer to Figure 8.9 (4)):
    - References to specific resources in the content of both *explicit* and *tacit knowledge* entities, e.g. *information systems, workspace, equipment* and *funding*.

      Inv-1.2, Inv-1.3 → Inv-3.1, Inv-3.2, Inv-3.3, Inv-3.4
    - Support for *information artefacts* in *resources* (for more detail refer to section 8.5.3.4.3 on *resources*).

      Inv-3 → Inv-1.1

Knowledge (Inv-1) and its constituent entities furthermore show **integrations** (refer to Table 8.4) (E) with the following concepts in other abstractions (columns) of the reference architecture for IKNs:

- *Knowledge creation and transfer*; the process in which new *knowledge* entities are created and existing *knowledge* entities are transferred between types.

  Inv-1 ↔ Proc-2
- *Network stakeholder locations*; through the hosting of *information artefacts* and *explicit knowledge* at these locations.

  Inv-1.1, Inv-1.2 ↔ Dist-2.1
- *Knowledge owners*; the responsibility assignment that possesses *tacit knowledge*.

  Inv-1.3 ↔ Resp-3
- *Network stakeholders*; *knowledge*, especially *tacit knowledge*, is linked to *network stakeholders,* as they are *knowledge owners*.

  Inv-1.3 (→ Resp-3) ↔ Resp-1
- *Knowledge life cycle*; which describes the timing of *knowledge* operations.

  Inv-1 ↔ Tim-3
- *Network domain and purpose*; as the creation and transfer of knowledge amongst IKN members is one of the fundamental motivations for the existence of the network.

  Inv-1 ↔ Mot-3.2

8.5.3.4.2 Innovation artefacts

This inventory set definition model (C1, R2.2.2) defines *innovation artefacts* (Inv-2) as an inventory set at an intermediate level of the detail. The textual version of this model is presented in Table 8.5, and the visual version in Figure 8.10.

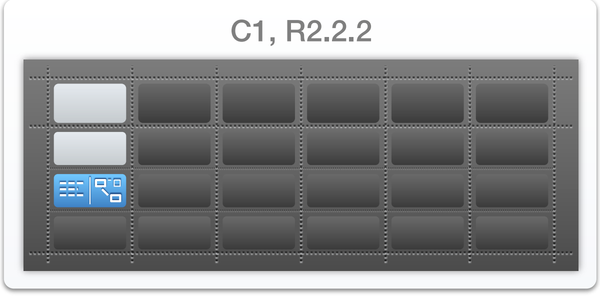


Table 8.5: Inventory set definition model - Innovation artefacts (C1, R2.2.2)

	Code	**Name / definition**
A. Concept	**Inv-2**	**Innovation artefacts**
B. Definition		Objects that are created during, or as a result of the innovation process. These objects can be both of a tangible or intangible nature.
C. Deconstruction	Inv-2.1	Ideas
	Inv-2.2	Concepts
	Inv-2.3	Prototypes
	Inv-2.4	Projects
	Inv-2.5	Products
	Inv-2.6	Services
D. Associations	Inv-1	Knowledge
	Inv-3	Resources
E. Integrations	Proc-3	Innovation
	Dist-2.1	Network stakeholder locations
	Dist-2.2	Operational locations
	Resp-2	Innovators
	Resp-1	Network stakeholders
	Resp-3	Knowledge owners
	Tim-1	Technology life cycle
	Tim-4	Innovation life cycle
	Mot-3.2	Network domain and purpose

The *innovation artefacts* (Inv-2) inventory set is **defined** (refer to Table 8.5) (B) as follows:

> *Innovation artefacts are objects that are created during, or as a result of the innovation process. These objects can be either tangible or intangible.*

The *innovation artefacts* inventory set is **deconstructed** (refer to Table 8.5 (C), Figure 8.10 (1)) into six constituent entities, namely *ideas*, *concepts*, *prototypes*, *projects*, *products* and *services*, via ”is-a” relationships.

Inv-2 →
Inv-2.1, Inv-2.2,
Inv-2.3, Inv-2.4,
Inv-2.5, Inv-2.6

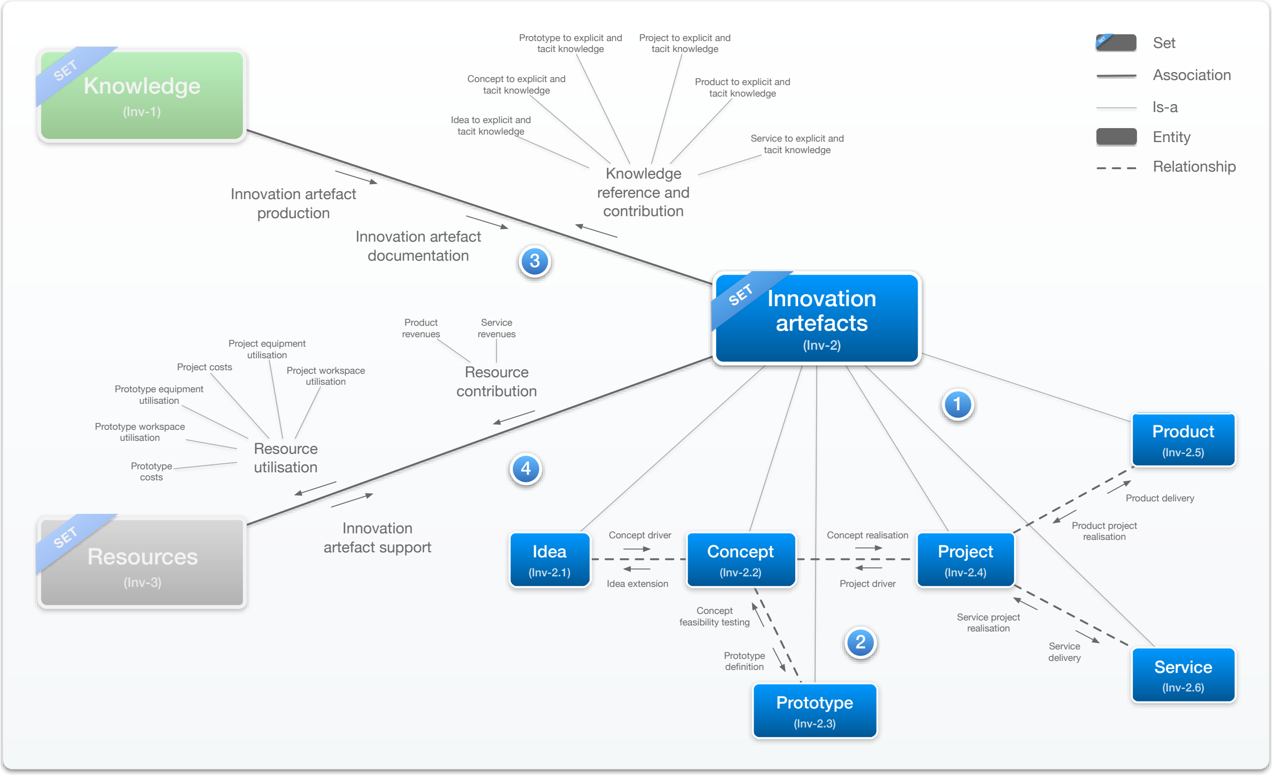


Figure 8.10: Inventory set definition model – Innovation artefacts (C1, R2.2.2)

An *idea* (Inv-2.1) is:

> *A thought or suggestion as to a possible course of innovative action.*

Ideas are the first entities in the chain of *innovation artefacts* produced in the *innovation* process, where they are *generated*, *collected*, *documented* and *filtered*. *Ideas* can be spur-of-the-moment thoughts, or can be purposefully generated in workshops or brainstorming sessions. *Ideas* may reference the content of both *explicit* and *tacit knowledge* entities, and contribute to *tacit knowledge* through their generation and to *explicit knowledge* through their documentation. Once *documented* and *filtered*, *ideas* are the drivers for the *development* of *concepts*.

A *concept* (Inv-2.2) is:

> *A mature and viable idea or collection of related ideas. The purpose of a concept is to serve as a vehicle for testing the feasibility of an innovation.*

Concepts are the second entities in the chain of *innovation artefacts* produced in the *innovation* process, and are extensions of *ideas*. During the *innovation* process, *concepts* are *developed*, *incubated and refined*, *filtered*, *tested*, *refined*, *approved* and *funded*. Concepts may reference the content of both *explicit* and *tacit knowledge* entities, and contribute to *tacit knowledge* through their generation and to *explicit knowledge* through their documentation. *Concepts* define which *prototypes* are to be *built*. *Approved* and *funded concepts* are the drivers of innovation *projects*.

A *prototype* (Inv-2.3) is:

> *A first or preliminary version of an innovation, designed to test the feasibility of a concept, from which further formalised versions are developed.*

Prototypes are the third entities in the chain of *innovation artefacts* produced in the *innovation* process, and are *built* to test the feasibility of *concepts*. *Prototypes* may reference the content of both *explicit* and *tacit knowledge* entities, and contribute to *tacit knowledge* through their generation and to *explicit knowledge* through their documentation. *Prototypes* utilise *workspaces* and *equipment* during their *building*, with their costs paid for from network *funding*.

A project (Inv-2.4) is:

> *A collaborative endeavour that is carefully planned and executed to deliver an innovation.*

Projects are the fourth entities in the chain of *innovation artefacts* produced in the *innovation* process, and realise *concepts* by delivering *products* or *services*. During the *innovation* process, *projects* are *approved for funding*, *managed in a portfolio*, *prepared for launch*, *approved for launch*, *planned*, *detail designed and tested*, *approved for implementation*, *implemented*, *operated, refined and formalised*, and *selected for exploitation*. *Projects* may reference the content of both *explicit* and *tacit knowledge* entities, contribute to *tacit knowledge* through their planning and execution, and contribute to *explicit knowledge* through their documentation. *Projects* utilise *workspaces* and *equipment* during their planning and execution, and their costs are paid for from network *funding*.

A product (Inv-2.5) is:

> *An article or substance that is manufactured or refined to create value, and the result of the innovation process.*

Products are the fifth entities in the chain of *innovation artefacts* produced in the *innovation* process, and are produced and delivered by *projects*. As such, *products* are a possible realisation of innovation *projects*. During the *innovation* process, *products* form part of business models that may be *selected for exploitation*. *Products* may reference the content of both *explicit* and *tacit knowledge* entities, and contribute to *tacit knowledge* through their production and delivery, and to *explicit knowledge* through their documentation. *Products* may *create value* in the form of generated revenues that contribute to network *funding*.

A service (Inv-2.6) is:

> *An action of helping or work done for someone, including assistance or advice given to customers during and after the sale of goods.*

Services are the sixth entities in the chain of *innovation artefacts* produced in the *innovation* process, and are produced and delivered by *projects*. As such, *services* are a possible realisation of innovation *projects*. During the *innovation* process, *services* form part of business models that may be *selected for exploitation. Services* may reference the content of both *explicit* and *tacit knowledge* entities, and contribute to *tacit knowledge* through their production and delivery, and to *explicit knowledge* through their documentation. *Services* may *create value* in the form of generated revenues that contribute to network *funding*.

Based on the preceding discussion, the constituent entities of the *innovation artefacts* (Inv-2) inventory set share the following relationships (refer to Figure 8.10) (2):

- *Ideas* are the drivers for the development of *concepts*. Inv-2.1 → Inv-2.2
- *Concepts* extend *ideas* to reach a next level of maturity. Inv-2.2 → Inv-2.1
- *Concepts* define which *prototypes* are to be built. Inv-2.2 → Inv-2.3
- *Prototypes* test the feasibility of *concepts*. Inv-2.3 → Inv-2.2
- *Concepts* are the drivers for innovation *projects.* Inv-2.2 → Inv-2.4
- *Projects* are the realisation of feasible *concepts.* Inv-2.4 → Inv-2.2
- *Products* are developed and delivered by innovation *projects*, and are the realisation of these *projects*. Inv-2.4 ↔ Inv-2.5
- *Services* that are developed and delivered by innovation *projects*, and are the realisation of these *projects*. Inv-2.4 ↔ Inv-2.6

Innovation artefacts (Inv-2) and its constituent entities also show **associations** (refer to Table 8.5) (D) with the following sets in the inventory abstraction (column):

- *Knowledge* (Inv-1; refer to Figure 8.10 (3)):
  - References and contributions to both *explicit knowledge* and *tacit knowledge* in the content of *ideas*, *concepts*, *prototypes*, *projects*, *products* and *services*. Inv-2.1, Inv-2.2, Inv-2.3, Inv-2.4, Inv-2.5, Inv-2.6 → Inv-1.2, Inv-1.3
  - The provision of *knowledge* as input to the production of *innovation artefacts*, e.g. *ideas*, *concepts*, *prototypes*, *projects*, *products* and *services* (for more detail refer to section 8.5.3.4.1 on *knowledge*). Inv-1 → Inv-2.1, Inv-2.2, Inv-2.3, Inv-2.4, Inv-2.5, Inv-2.6
  - The documentation of *ideas*, *concepts*, *prototypes*, *projects*, *products* and *services* as *knowledge* (for more detail refer to section 8.5.3.4.1 on *knowledge*). Inv-1 → Inv-2.1, Inv-2.2, Inv-2.3, Inv-2.4, Inv-2.5, Inv-2.6

- *Resources* (Inv-3; refer to Figure 8.10 (4)):
    - The utilisation of *resources,* e.g. the utilisation of *workspaces* and *equipment*, as well as the *funding*, of *prototypes* and *projects*.

      Inv-2.3, Inv-2.4 → Inv-3.2, Inv-3.3, Inv-3.4
    - The contribution to *resources*, e.g. *product* and *service* revenues to network *funding*.

      Inv-2.5, Inv-2.6 → Inv-3.4
    - The support of *prototypes*, *projects*, *products* and *services* through *resources* (for more detail refer to section 8.5.3.4.3 on *resources*).

      Inv-3 → Inv-2.3, Inv-2.4, Inv-2.5, Inv-2.6

Innovation artefacts (Inv-2) and its constituent entities furthermore show **integrations** (refer to Table 8.5) (E) with the following concepts in other abstractions (columns) of the reference architecture for IKNs:

- *Innovation*; the process during which *innovation artefacts* are produced and transformed.

  Inv-2 ↔ Proc-3
- *Network stakeholder locations*; the locations where *prototypes* and *projects* are hosted.

  Inv-2 ↔ Dist-2.1
- *Operational locations*; locations where *innovation artefacts* are found, including locations that do not overlap with *network stakeholder locations*.

  Inv-2 ↔ Dist-2.2
- *Innovators*; the responsibility assignment of functionally producing and transforming *innovation artefacts*.

  Inv-2 ↔ Resp-2
- *Network stakeholders*; the responsibility assignment that is tasked with organisationally producing and transforming the various *innovation artefacts* through their association with *innovators*.

  Inv-2 ↔ (Resp-2 →) Resp-1
- *Knowledge owners*; the responsibility assignment which is relied upon for *tacit knowledge* inputs to the production and transformation of *innovation artefacts* and to which *innovation artefacts* contribute *tacit knowledge.*

  Inv-2 ↔ Resp-3
- *Technology life cycle*; the timing cycle that enables or instigates the production of *innovation artefacts*.

  Inv-2 ↔ Tim-1
- *Innovation life cycle*; the timing cycle that coordinates the production and transformation of *innovation artefacts* in the *innovation* process.

  Inv-2 ↔ Tim-4
- *Network domain and purpose*; the motivational end for the production and transformation of *innovation artefacts*, as well as their interrelations.

  Inv-2 ↔ Mot-3.2

8.5.3.4.3 Resources

This inventory set definition model (C1, R2.2.3) defines *resources* (Inv-3) as an inventory set at an intermediate level of the detail. The textual version of this model is presented in Table 8.6, and the visual version in Figure 8.11.

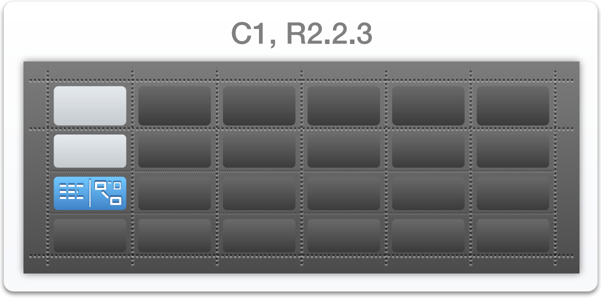


Table 8.6: Inventory set definition model – Resources (C1, R2.2.3)

	Code	Name / definition
A. Concept	**Inv-3**	**Resources**
B. Definition		Assets that can be drawn on by an organisation in order to function effectively.
C. Deconstruction	Inv-3.1	Information systems
	Inv-3.2	Workspaces
	Inv-3.3	Equipment
	Inv-3.4	Funding
D. Associations	Inv-1	Knowledge
	Inv-2	Innovation artefacts
E. Integrations	Proc-3	Innovation
	Dist-2.1	Network stakeholder locations
	Dist-2.2	Operational locations
	Resp-1	Network stakeholders
	Tim-1	Technology life cycle
	Tim-4	Innovation life cycle
	Mot-3.9	Operational requirement rules

The *resources* (Inv-3) inventory set is **defined** (refer to Table 8.6) (B) as:

> *Assets that can be drawn on by an IKN in order to function effectively.*

The *resources* inventory set is **deconstructed** (refer to Table 8.6 (C), Figure 8.11 (1)) into four constituent entities, namely *information systems*, *workspaces, equipment* and *funding*, via "is-a" relationships.

Inv-3 →
Inv-3.1, Inv-3.2,
Inv-3.3, Inv-3.4

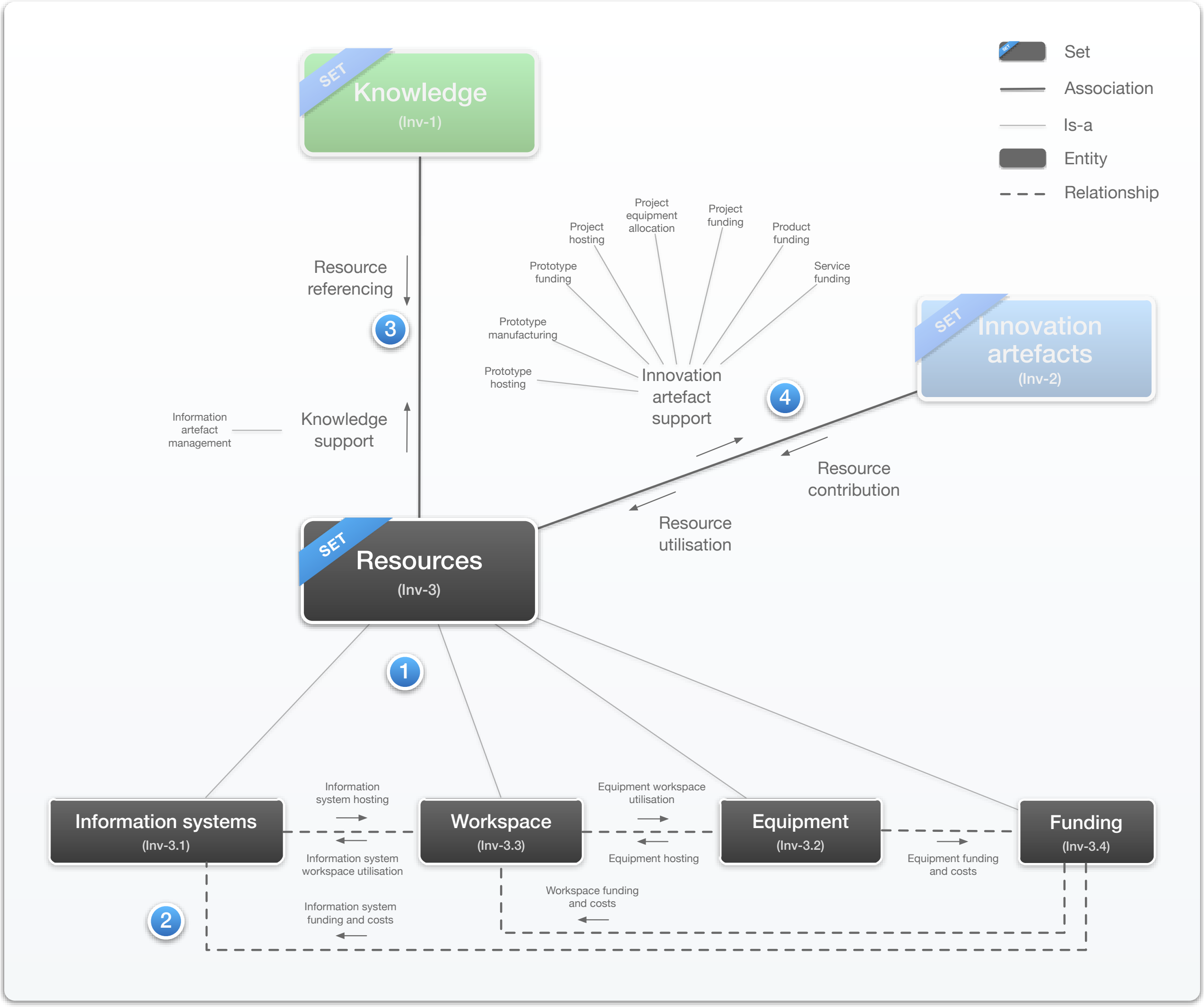


Figure 8.11: Inventory set definition model – Resources (C1, R2.2.3)

Information systems (Inv-3.1) are:

> *Sets of complementary networks of hardware and software that IKNs and their constituent members use to collect, filter, process, create, and distribute data.*

The objective of *information systems* is to manage *information artefacts*, and *network stakeholders* are the users of such systems. The cost of *information systems* may be paid for from *network* funding, or the cost could be contributed by *network stakeholders*. In such cases *information systems* could be located at *network stakeholder locations* or other *operational locations*.

Workspaces (Inv-3.2) are:

> *Spaces in which the work of an IKN is done, e.g. a laboratory, office space, factory space.*

Workspaces host *information systems*, *equipment* and the production of *prototypes*. Innovation *projects* may also be hosted in *workspaces*. The cost of *workspaces* may be paid for from *network* funding, or it could be contributed by *network stakeholders*. In such cases *workspaces* could be located at *network stakeholder locations* or other *operational locations*.

Equipment (Inv-3.3) is:

> *The necessary items for a particular purpose in an IKN, e.g. manufacturing equipment, research equipment, and office equipment.*

Equipment is used for the manufacturing of *prototypes* and may be assigned to *projects* for the production of *products* and/or *services*. *Equipment* utilise *workspaces*, and its cost may be paid for from network *funding*, or it could be contributions from *network stakeholders*. In such cases *equipment* could be located at *network stakeholder locations* or other *operational locations*.

Funding (Inv-3.4) is:

> *Money provided by a member or members of an IKN for a particular purpose.*

The costs of *prototypes*, *projects, products*, *services*, *information systems*, *workspaces* and *equipment* are paid for from network *funding*. The funds are contributed by one or more *network stakeholders*.

Based on the preceding discussion, the constituent entities of the *resources* (Inv-3) inventory set share the following relationships (refer to Figure 8.11) (2):

- The hosting of *information systems* and *equipment* as *workspace* utilisation.

  Inv-3.2 ↔ Inv-3.1, Inv-3.3

- The costs of *information systems, equipment* and *workspaces* that are paid for from network *funding*.

  Inv-3.1, Inv-3.2, Inv-3.3 ↔ Inv-3.4

Resources (Inv-3) and its constituent entities also show **associations** (refer to Table 8.6) (D) with the following sets in the inventory abstraction (column):

- *Knowledge* (Inv-1; refer to Figure 8.11 (3)):
  - The management of *information artefacts* in *information systems*.

    Inv-3.1 → Inv-1.1

  - References to particular *resources*, e.g. *information systems*, *workspaces*, *equipment* and *funding*, in the content of *knowledge* entities (for more detail refer to section 8.5.3.4.1 on *knowledge*).

    Inv-1 → Inv-3.1, Inv-3.2, Inv-3.3, Inv-3.4

- *Innovation artefacts* (Inv-2; refer to Figure 8.11 (4)):

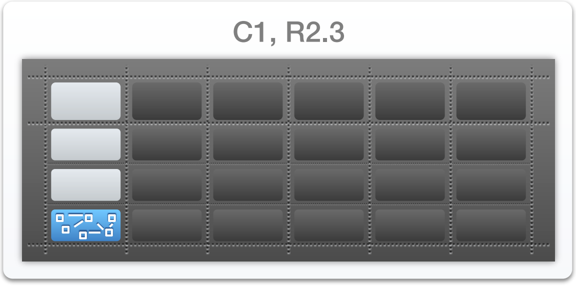
- The hosting of *prototypes* and *projects* in *workspaces*. Inv-3.2 → Inv-2.3, Inv-2.4
- The manufacturing of *prototypes* with *equipment*. Inv-3.3 → Inv-2.3
- The allocation of *equipment* for use in *projects*. Inv-3.3 → Inv-2.4
- The *funding* of *prototypes, projects, products* and *services*. Inv-3.4 → Inv-2.3, Inv-2.4, Inv-2.5, Inv-2.6
- The contribution and utilisation of *resources* by *innovation artefacts* (refer to section 8.5.3.4.2 for more detail of *innovation artefacts*). Inv-2 → Inv-3

Resources (Inv-3) and its constituent entities furthermore show **integrations** (refer to Table 8.6) (E) with the following concepts in other abstractions (columns) of the reference architecture for IKNs:

- *Innovation* – the process in which *innovation artefacts,* which relate to various *resources,* are produced and transformed. Inv-3 ↔ Proc-3
- *Network stakeholder locations* – *information systems, workspaces* and *equipment* may be located here. Inv-3 ↔ Dist-2.1
- *Operational locations* – *information systems, workspaces* and equipment may be located here. Inv-3 ↔ Dist-2.2
- *Network stakeholders* – may contribute *information systems*, *workspaces*, *equipment* and *funding*, and are also the users of *information systems*. Inv-3 ↔ Resp-1
- *Technology life cycle* – this timing cycle dictates when certain *resources* will be available for the production and transformation of *innovation artefacts*. Inv-3 ↔ Tim-1
- *Innovation life cycle* – this timing cycle dictates when certain *resources* are required to support the production and transformation of *innovation artefacts*. Inv-3 ↔ Tim-4
- *Operational requirement rules* – this motivation influences the network's approach to sourcing and sharing of *resources*. Inv-3 ↔ Mot-3.9

8.5.3.5 Extended inventory definition model

The extended inventory definition model (C1, R2.3) defines the inventory sets identified in the inventory identification model (C1, R1; refer to Table 8.3) at a high level of detail by representing them visually according to the abstraction meta-model (refer to Figure 8.7). This model is based on the interpretation of the *definition data* from the qualitative systematic review described in section 8.5.2.1, with regard to the "*What?*" review question.



The various constituent entities are indicated and organised into swim lanes according to their overarching high-level inventory sets (refer to Figure 8.12). The detailed relationships between them are also labelled appropriately. This model therefore retains all the defining deconstructions and associations from the inventory set definition models in the preceding sections.

No deconstruction of entities into sub-entities is done for the extended inventory definition model. To ensure that the model remains a primitive (single-variable) model, as well as for readability reasons, integrations with concepts in other abstractions (columns) are not indicated in the model.

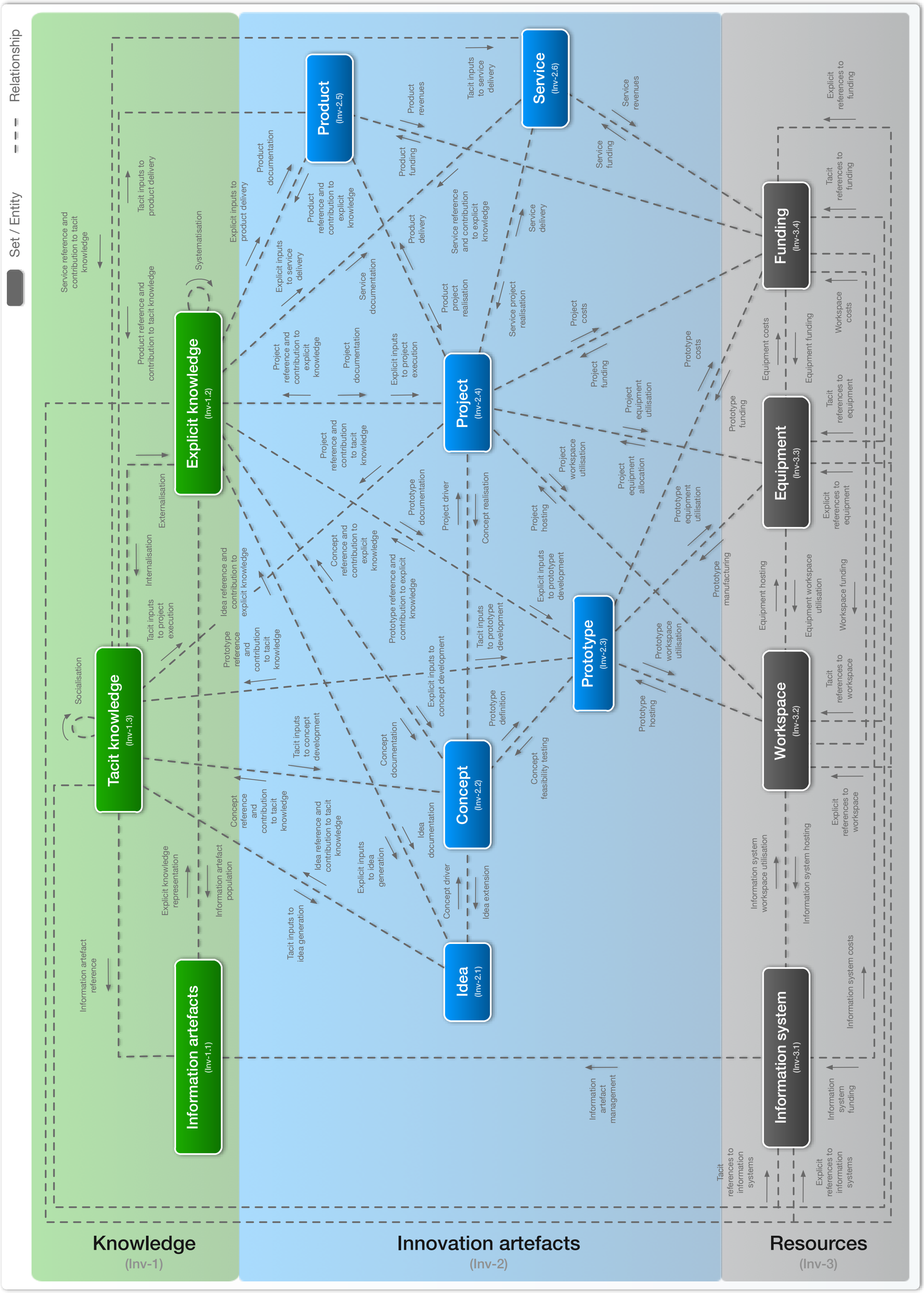
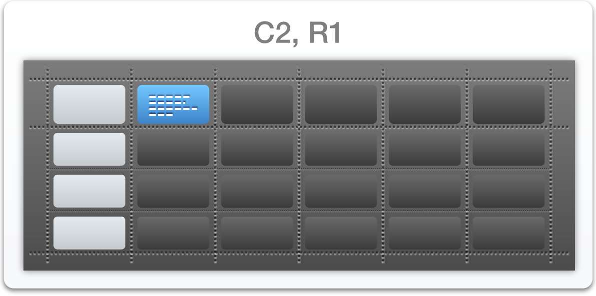


Figure 8.12: Extended inventory definition model

8.5.4 Process flow models

This section discusses the process abstraction (second column) of the artefact, and the models presented therefore answer the interrogative of "How?" in the engineering of IKNs. The top row reference model is presented and motivated first. Thereafter the meta-model for the second row of the second column is presented, followed by the various second row reference models according to increasing levels of detail. Where the description of a model includes a reference to another formalised concept in the reference architecture, the name of the concept is printed in italics.

8.5.4.1 Process identification model



The process identification model (C2, R1; refer to Table 8.7) lists process types that are significant and common to the engineering of IKNs. This model is based on the interpretation of the *identification data* from the qualitative systematic review described in section 8.5.2.1, with regard to the "*How?*" review question.

At the lowest level of detail, three main process types are identified, namely *network building*, *knowledge creation and transfer,* and *innovation*. Constituent process subtypes are also identified that classify these types at a higher level of detail.

Table 8.7: Process identification model

Code	**Process type**	**Code**	**Constituent process subtypes**
Proc-1	**Network building**	Proc-1.1	Network design
		Proc-1.2	Network implementation
		Proc-1.3	Network operation
		Proc-1.4	Network refinement
		Proc-1.5	Network phase-out
Proc-2	**Knowledge creation and transfer**	Proc-2.1	Socialisation
		Proc-2.2	Externalisation
		Proc-2.3	Systematisation
		Proc-2.4	Internalisation
Proc-3	**Innovation**	Proc-3.1	Identification and evaluation

Code	Process type	Code	Constituent process subtypes
		Proc-3.2	Portfolio management
		Proc-3.3	Deployment and exploitation

Network building is a key process type for IKNs, as it is the process that constructs the network as an enterprise. This process accounts for the intentional design and implementation of an IKN, as well as its refinement and adaptation to the innovative landscape in which it operates. The *network building* process type is deconstructed into process subtypes of *network design*, *network implementation*, *network operation*, *network refinement* and *network phase-out*.

The identification of *knowledge creation and transfer* as a key process in the engineering of IKNs is in line with the knowledge-sharing purpose of these networks (refer to section 2.5.4). There exists deep links between this process and the *innovation* process, which also characterises IKNs. *Knowledge creation and transfer* is deconstructed into process subtypes of *socialisation*, *externalisation*, *systematisation* and *internalisation*.

As with *knowledge creation and transfer*, the identification of *innovation* as a key process in the engineering of IKNs is in line with the purpose of IKNs, in this case particularly the innovative purpose of these networks (refer to section 2.5.4). The *innovation* process type is deconstructed into process subtypes of *identification and evaluation*, *portfolio management,* and *deployment and exploitation*.

The *knowledge creation and transfer* and *innovation* process flows are two of the characteristic elements of IKNs that distinguish them from other kinds of collaborative networked organisations (CNOs).

Definitions for the process types and subtypes identified in the process identification model are provided in the second-row process definition models presented in sections 8.5.4.3, 8.5.4.4 and 8.5.4.5.

8.5.4.2 Meta-model for definition models in process abstraction

The primitive meta-model presented in Figure 8.13 serves as the semi-formal representational language for the various second-row definition models in the process abstraction, or ‘*how*’ column, of the reference architecture for IKNs. This meta-model assists in defining *process flows* as concepts and follows the single-variable structure of “*transform–input/output–transform*” amongst its elements, links, sub-elements and sub-links. The meta-model therefore deconstructs and represents process flows into transforms and sub-transforms, along with their respective inputs/outputs and sub-inputs/outputs.

This representation is done at various levels of detail in the basic process definition model (refer to section 8.5.4.3), process flow definition models (refer to section 8.5.4.4) and extended process definition model

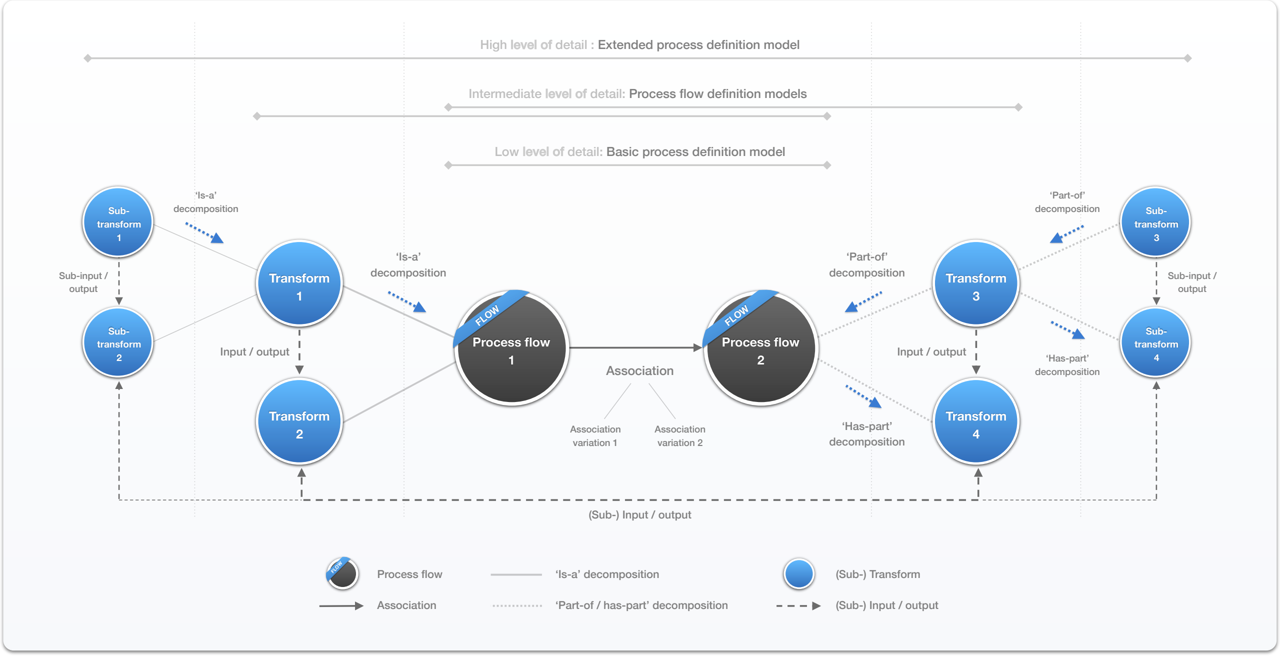
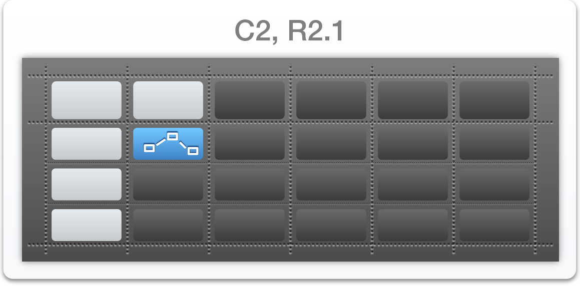


Figure 8.13: Meta-model for definition models in process abstraction

(refer to section 8.5.4.5). For all of these models, the meta-model scope indication shown at the top of Figure 8.13 applies.

8.5.4.3 Basic process definition model

The basic process definition model (C2, R2.1) defines the process flows that were identified as process types in the process identification model (C2, R1; refer to Table 8.7) at a low level of detail by representing them visually according to the abstraction meta-model (refer to Figure 8.13). This model is based on the interpretation of the *definition data* from the qualitative systematic review described in section 8.5.2.1, with regard to the "*How?*" review question.



The model depicts the various identified process flows, as well as the associations between them (refer to Figure 8.14). The model shows that the *network building* process flow associates with both *knowledge creation and transfer* and *innovation* through the design, implementation, operation and refinement of an IKN that is intentionally constructed to facilitate these two key process flows. The second process flow, *knowledge creation and transfer*, supports *innovation* by providing and transforming the required *knowledge* (both *tacit* and *explicit*) to execute this complex process and produce *innovation artefacts* (refer to section 8.5.3.4.1 on *knowledge,* and section 8.5.3.4.2 on *innovation artefacts*) on the one hand. Execution of the

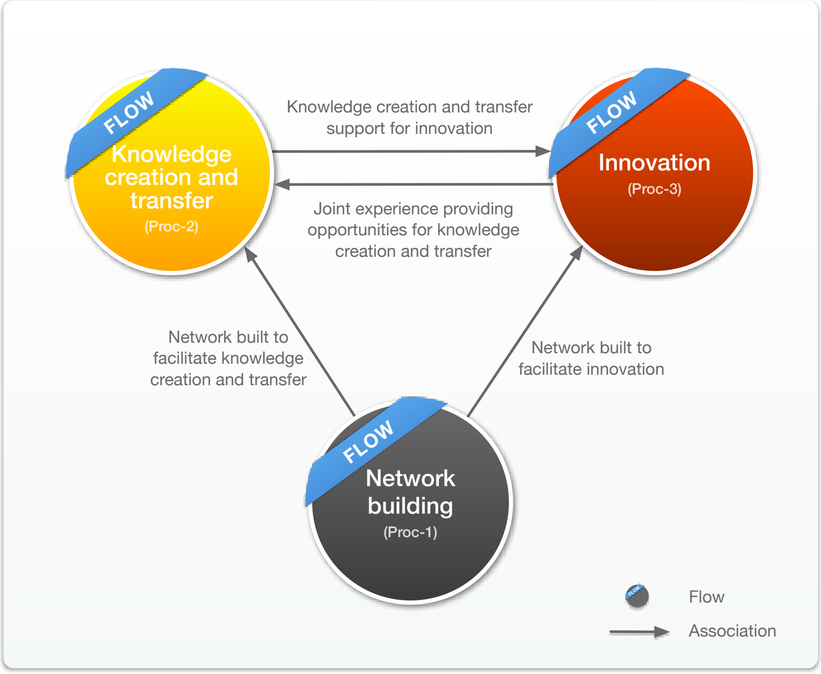


Figure 8.14: Basic process definition model

innovation process flow in an IKN, on the other hand, leads to joint experience that provides opportunities for *knowledge creation and transfer*.

These process flows are now defined in more detail in the applicable process flow definition models in section 8.5.4.4.

8.5.4.4 Process flow definition models

The process flow definition models presented in this section each define an identified process flow at an intermediate level of detail. The models therefore add new information to each flow, while retaining all the information from the basic process definition model (refer to Figure 8.14). Both textual and visual versions are provided to assist in defining the flows. Deconstructions, relationships and integrations between concepts are expressed in terms of identification codes in side-notes for traceability purposes.

The models presented in this section are based on the interpretation of the *definition data* for each identified process flow, as obtained from the qualitative systematic review described in section 8.5.2.1.

8.5.4.4.1 Network building

This process flow definition model (C1, R2.2.1) defines *network building* (Proc-1) as a process flow at an intermediate level of the detail. The textual version of this model is initially presented in Table 8.8 and the visual version in Figure 8.15.

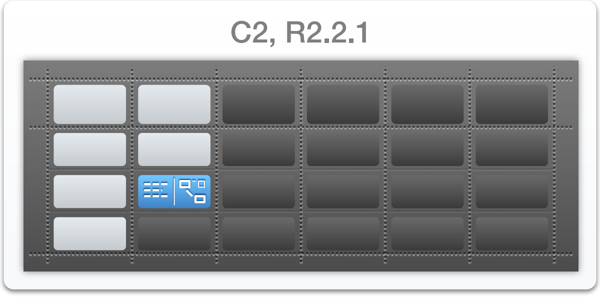


Table 8.8: Process flow definition model - Network building (C2, R2.2.1)

	Code	Name / definition
A. Concept	**Proc-1**	**Network building**
B. Definition		The process of designing, developing, improving and phasing out the integrated knowledge network.
C. Deconstruction	Proc-1.1	Network design
	Proc-1.2	Network implementation
	Proc-1.3	Network operation
	Proc-1.4	Network refinement
	Proc-1.5	Network phase-out
	Proc-1.6	Network legacy
D. Associations	Proc-2	Knowledge creation and transfer
	Proc-3	Innovation
E. Integrations	Dist-1	Strategic network
	Dist-2	Operational network
	Resp-1	Network stakeholders
	Tim-2	Network life cycle
	Mot-3.4	Network building rules

The *network building* process flow is **defined** (refer to Table 8.8) (B) as:

> *The process of designing, developing, improving and phasing out the integrated knowledge network.*

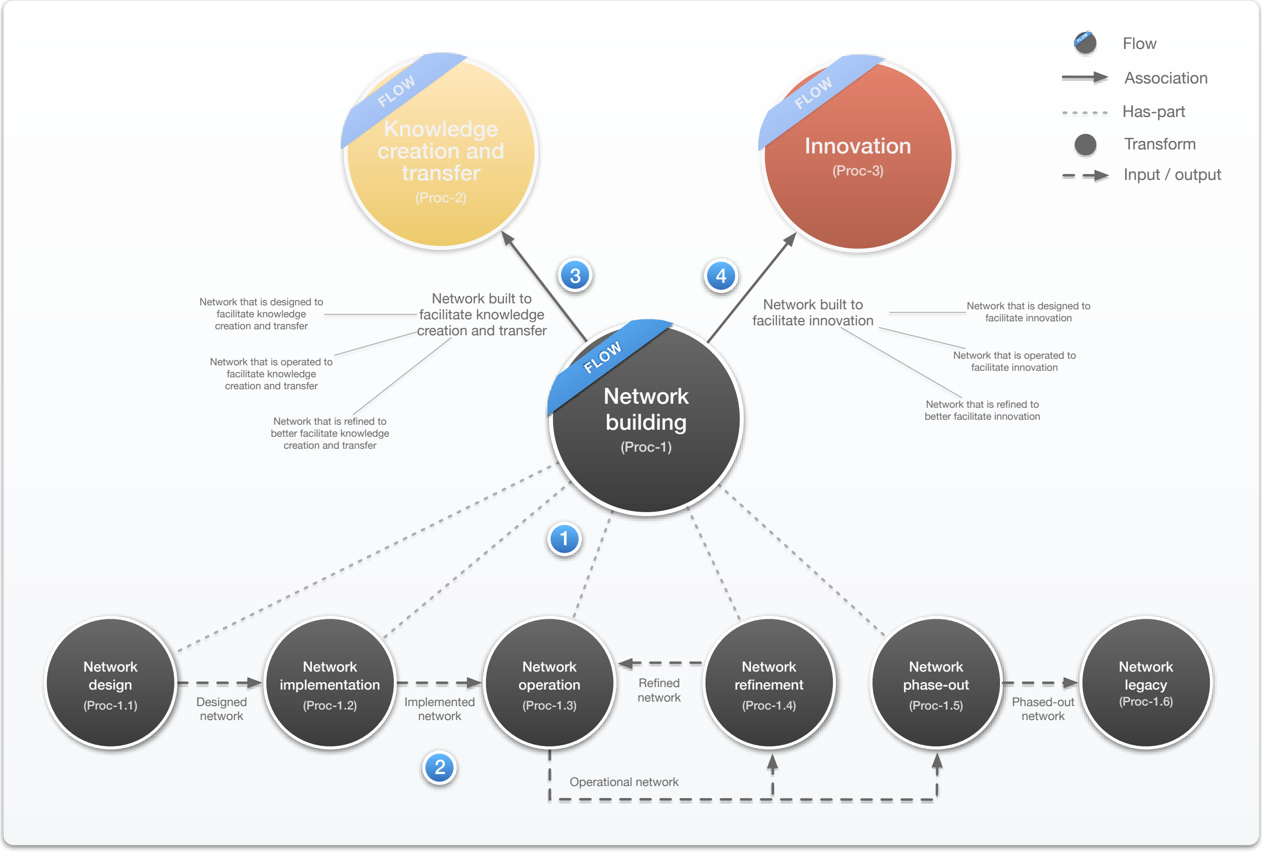


Figure 8.15: Process flow definition model - Network building (C2, R2.2.1)

Network building supports both *knowledge creation and transfer* and *innovation* through the design, implementation, operation and refinement of an IKN that is intentionally constructed to facilitate these two key process flows.

The *network building* process flow is **deconstructed** (refer to Table 8.8 (C), Figure 8.15 (1)) into six constituent transforms, namely *network design*, *network implementation*, *network operation*, *network refinement*, *network phase-out* and *network legacy*, via “part-of/has-part” relationships.

Proc-1 →
Proc-1.1, Proc-1.2,
Proc-1.3, Proc-1.4,
Proc-1.5, Proc-1.6

Network design (Proc-1.1) is:

> *The planning of an IKN with the purpose of arriving at an organisational configuration that is implementable.*

The objective of the *network design* transform is to generate a plan and configuration of an IKN that will support and enable both the *knowledge creation and transfer* and *innovation* process flows. The *network design* transform provides a designed network to the *network implementation* transform.

Network implementation (Proc-1.2) is:

> *The action of putting the design of an IKN into effect, thereby creating a network instance based on prior planning.*

The *network implementation* transform provides an implemented IKN to the *network operation* transform. It is crucial that this implemented IKN resembles the designed network from the *network design* transform.

Network operation (Proc-1.3) is:

> *The management of an active, functioning IKN that operates in line with its purpose.*

The objective of the *network operation* transform is to sustain the IKN in order for the *knowledge creation and transfer* and *innovation* process flows to be executed. The *network operation* transform provides an operational network to the *network refinement* and *network phase-out* transforms.

Network refinement (Proc-1.4) is:

> *Augmenting elements of an IKN to ensure increased network performance, or to pursue a new strategic direction altogether.*

The *network refinement* transform aims to measure and improve the operational IKN in order to improve the performance of the *knowledge creation and transfer* and *innovation* process flows. It is an important transform that ensures that the IKN adapts to a changing environment and the requirements of the other process flows it supports. *Network refinement* provides a refined network to the *network operation* transform.

Network phase-out (Proc-1.5) is:

> *The discontinuation an IKN and the archival its generated outputs.*

The managed discontinuation of an IKN that has either served its purpose or has failed is important in order to ensure sustained availability of the outputs that were generated during its operation. The *network phase-out* transform provides a phased-out network to the *network legacy* transform.

Network legacy (Proc-1.6) is:

> *The existence of the legacy of a phased-out IKN.*

The *network legacy* transform ensures the sustained availability of the archived outputs of a phased-out IKN.

Based on the preceding discussion, the constituent transforms of the *network building* (Proc-1) process flow share the following inputs/outputs (refer to Figure 8.15) (2), namely:

- *Network design* passing a designed network to *network implementation*. Proc-1.1 → Proc-1.2
- *Network implementation* passing an implemented network to *network operation*. Proc-1.2 → Proc-1.3
- *Network operation* passing an operational network to *network refinement* and *network phase-out*. Proc-1.3 → Proc-1.4, Proc-1.5
- *Network refinement* passing a refined network to *network operation*. Proc-1.4 → Proc-1.3
- *Network phase-out* passing a phased-out network to the *network legacy*. Proc-1.5→ Proc-1.6

Network building (Proc-1) and its constituent transforms also show **associations** (refer to Table 8.8) (D) with the following flows in the process abstraction (column):

- *Knowledge creation and transfer* (Proc-2; refer to Figure 8.15 (8)):
    - The *design*, *operation* and *refinement* of an IKN that is intentionally constructed to facilitate *knowledge creation and transfer*. Proc-1.1, Proc-1.3, Proc-1.4 → Proc-2
- *Innovation* (Proc-3; refer to Figure 8.15 (9)):
    - The *design*, *operation* and *refinement* of an IKN that is intentionally constructed to facilitate *innovation*. Proc-1.1, Proc-1.3, Proc-1.4 → Proc-3

Network building (Proc-1) and its constituent transforms furthermore show **integrations** (refer to Table 8.8) (E) with the following concepts in other abstractions (columns) of the reference architecture for IKNs:

- *Strategic network* – locations impact *network building* decisions on a strategic level, and vice versa. Proc-1 ↔ Dist-1
- *Operational network* – locations impact *network building* decisions on an operational level, and vice versa. Proc-1 ↔ Dist-2
- *Network stakeholders* – *network building* decisions impact the different types of *network stakeholders* that are involved in the IKN. Proc-1 ↔ Resp-1
- *Network life cycle* – this timing cycle cues and synchronises *network building* activities with other timing cycles that are significant to IKNs. Proc-1 ↔ Tim-2
- *Network building rules* – impacts the way in which *network building* is approached and governed in an IKN. Proc-1 ↔ Mot-3.4

8.5.4.4.2 Knowledge creation and transfer

This process flow definition model (C2, R2.2.2) defines *knowledge creation and transfer* (Proc-2) as a process flow at an intermediate level of the detail. The textual version of this model is presented in Table 8.9, and the visual version in Figure 8.16.

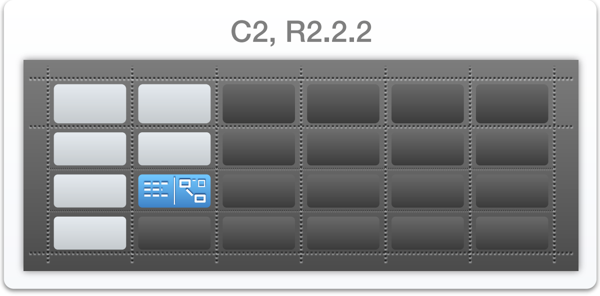


Table 8.9: Process flow definition model – Knowledge creation and transfer (C2, R2.2.2)

	Code	**Name / definition**
A. Concept	**Proc-2**	**Knowledge creation and transfer**
B. Definition		The process of transferring knowledge between individuals and knowledge stores, while converting between tacit and explicit formats, thereby creating new knowledge.
C. Deconstruction	Proc-2.1	Socialisation
	Proc-2.2	Externalisation
	Proc-2.3	Systematisation
	Proc-2.4	Internalisation
D. Associations	Proc-1	Network building
	Proc-3	Innovation
E. Integrations	Inv-1.1	Information artefacts
	Inv-1.2	Explicit knowledge
	Inv-1.3	Tacit knowledge
	Inv-3.1	Information systems
	Dist-1	Strategic network
	Dist-2.1	Network stakeholder locations
	Resp-3	Knowledge owners
	Resp-1	Network stakeholders
	Tim-3	Knowledge life cycle
	Mot-3.5	Knowledge creation and transfer rules

The *knowledge creation and transfer* (Proc-2) process flow is **defined** (refer to Table 8.9) (B) as:

> *The process of transferring knowledge between individuals and knowledge stores, while converting between tacit and explicit formats, thereby creating new knowledge.*

The *knowledge creation and transfer* process flow is **deconstructed** (refer to Table 8.9 (C), Figure 8.16 (1)) into four constituent transforms, namely *socialisation*, *externalisation*, *systematisation* and *internalisation*, via “part-of/has-part” relationships.

Proc-2 →
Proc-2.1, Proc-2.2,
Proc-2.3, Proc-2.4

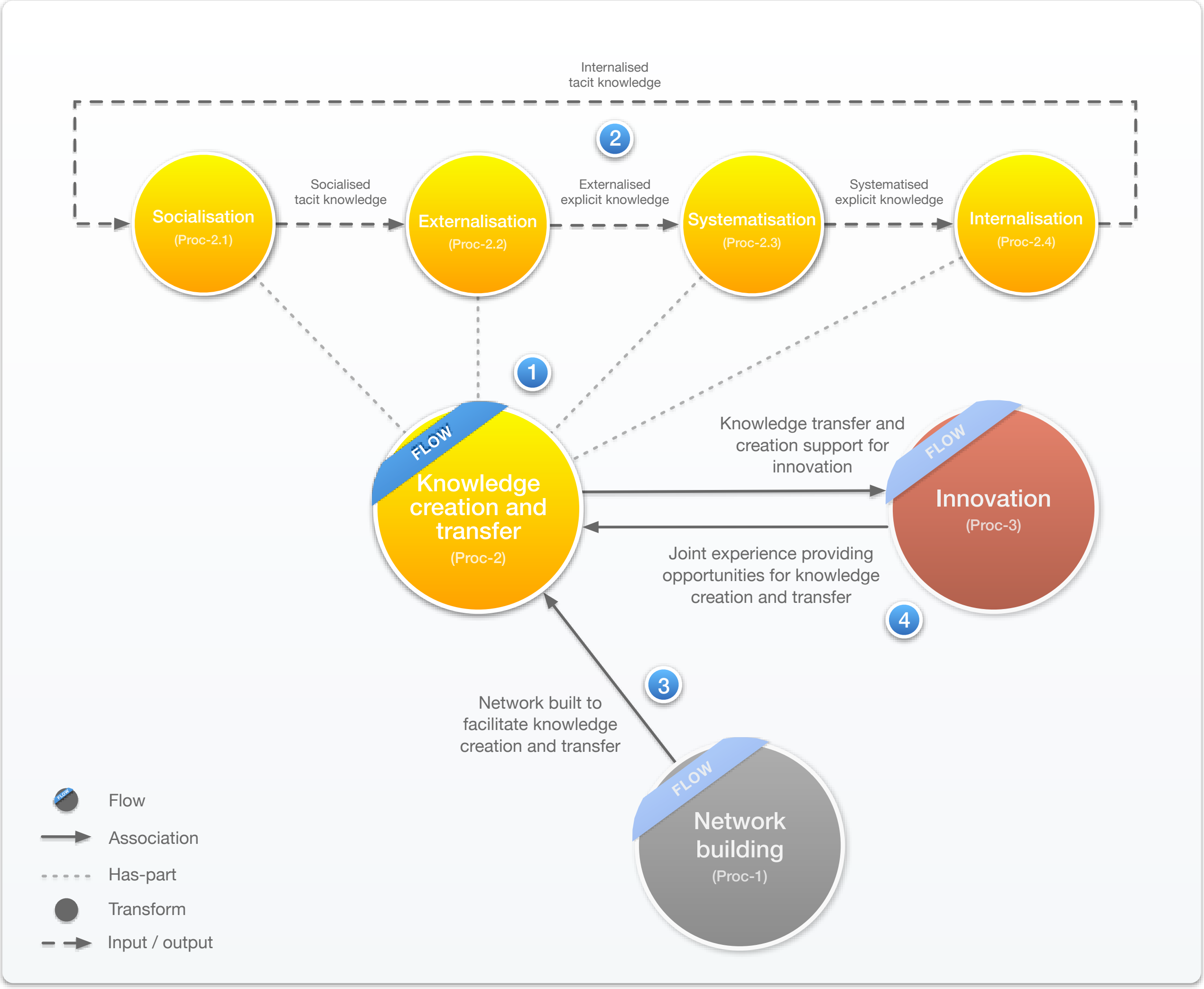


Figure 8.16: Process flow definition model – Knowledge creation and transfer (C2, R2.2.2)

Socialisation (Proc-2.1) is:

> *The exchange of tacit knowledge between knowledge owners in the network in order to convey personal knowledge and experience. Joint experience results in new, shared tacit knowledge such as common values or technical skills.*

Collaborative innovation projects as executed in operational IKNs leads to the creation and transfer of *tacit knowledge* through *socialisation* amongst network stakeholders. To enable the *knowledge creation and transfer* transform in an IKN that might be geographically dispersed, however, is a challenge and suitable mechanisms to overcome this issue need to be included during *network design*. The *socialisation* transform provides socialised tacit knowledge to the *externalisation* transform.

Externalisation (Proc-2.2) is:

> *The conversion of the tacit knowledge of knowledge owners into explicit knowledge. Since tacit knowledge is difficult to express, the conversion process is often supported by the use of metaphors, analogies, stories, and language rich in imagery, as well as visualisation aids, such as models, diagrams, or prototypes.*

The *externalisation* of *tacit knowledge* in a network context is enabled by shared values and trust amongst *network stakeholders*, which a facilitating condition that has to be managed during *network operation*. The *externalisation* transform provides externalised explicit knowledge to the *systematisation* transform.

Systematisation (Proc-2.3) is:

> *The transformation of explicit knowledge into more complex and more systematised explicit knowledge. It is necessary to combine different fields of explicit knowledge with each other and make new knowledge available on a network-wide basis. The systematisation and refinement increases the practical value of existing knowledge and increases its transferability to all network stakeholders.*

As the *systematisation* transform often involves *information artefacts* and the use of *information systems*, mechanisms to enable this transform in an inter-organisational context have to be included during *network design*, with the focus on the interoperability of systems. The *systematisation* transform provides systematised *explicit knowledge* to the *internalisation* transform.

Internalisation (Proc-2.4) is:

> *The conversion of network-wide explicit knowledge into the tacit knowledge of knowledge owners. This requires that the individual is able to recognise personally relevant knowledge within the organisation. Continuous learning and the gathering of one's own experience through 'learning-by-doing' supports employees in these internalisation processes.*

Internalised tacit knowledge and the experience gained on an individual level by *knowledge owners* can again be shared through *socialisation* transforms between network stakeholders, so that the knowledge spiral may be set in motion once more. The *internalisation* transform therefore provides internalised tacit knowledge to the *socialisation* transform.

Based on the preceding discussion, the constituent transforms of the *knowledge creation and transfer* (Proc-2) process flow share the following inputs/outputs (refer to Figure 8.16) (2):

- *Socialisation* passing socialised tacit knowledge to *externalisation*. Proc-2.1 → Proc-2.2
- *Externalisation* passing externalised explicit knowledge to *systematisation*. Proc-2.2 → Proc-2.3
- *Systematisation* passing systematised explicit knowledge to *internalisation*. Proc-2.3 → Proc-2.4
- *Internalisation* passing internalised tacit knowledge to *socialisation.* Proc-2.4 → Proc-2.1

Knowledge creation and transfer (Proc-2) and its constituent transforms also show **associations** (refer to Table 8.9) (D) with the following flows in the process abstraction (column):

- *Network building* (Proc-1; refer to Figure 8.16 (3)):
    - *Network building* accounts for the design, implementation, operation and refinement of an IKN that is intentionally constructed to facilitate *knowledge creation and transfer* (for more detail refer to section 8.5.4.4.1 on *network building*). Proc-1 → Proc-2
- *Innovation* (Proc-3; refer to Figure 8.16 (4)):
    - *Knowledge creation and transfer* supports *innovation* by providing and transforming the required *knowledge* (both *tacit* and *explicit*) to execute the *innovation* process and thereby produce *innovation artefacts*. Proc-2 → Proc-3
    - Execution of the *innovation* process flow in an IKN leads to joint experience amongst *knowledge owners* that provides opportunities for *knowledge creation and transfer* (for more detail refer to section 8.5.4.4.3 on *innovation*). Proc-3 → Proc-2

Knowledge creation and transfer (Proc-2) and its constituent transforms furthermore show **integrations** (refer to Table 8.9) (E) with the following concepts in other abstractions (columns) of the reference architecture for IKNs:

- *Information artefacts* – these inventory entities contain the *explicit knowledge* that is involved in *internalisation*, *externalisation* and *systematisation*. Proc-2.2, Proc-2.3, Proc-2.4 ↔ Inv-1.1
- *Explicit knowledge* – *internalisation* has *explicit knowledge* as an input, *externalisation* has it as an output, and *systematisation* has it as both an input and an output. Proc-2.2, Proc-2.3, Proc-2.4 ↔ Inv-1.2

- *Tacit knowledge* – *externalisation* has *tacit knowledge* as an input, *internalisation* has it as an output, and *socialisation* has it as both an input and an output.
  Proc-2.1, Proc-2.2, Proc-2.4 ↔ Inv-1.3
- *Information systems* – this resource manages *information artefacts* (see above) and enables the mechanisms for knowledge creation and transfer that connects network stakeholder locations, and impacts the network's approach to the *knowledge creation and transfer* process flow.
  Proc-2.2, Proc-2.3, Proc-2.4 ↔ Inv-3.1
- *Strategic network* – locations impact *knowledge creation and transfer* at a strategic level.
  Proc-2 ↔ Dist-1
- *Network stakeholder locations* – the *operational network* of an IKN, especially its *network stakeholder locations* and the mechanisms for knowledge creation and transfer, that connects these locations, impacts the network's approach to the *knowledge creation and transfer* process flow.
  Proc-2 ↔ Dist-2.1
- *Knowledge owners* – this role is linked to the *tacit knowledge* that is involved in *externalisation*, *internalisation* and *socialisation*.
  Proc-2.1, Proc-2.2, Proc-2.4 ↔ Resp-3
- *Network stakeholders* – *knowledge creation and transfer* is a phenomenon that occurs between *network stakeholders*, via their association with the *knowledge owners* responsibility assignment.
  Proc-2 ↔ (Resp-3 →) Resp-1
- *Knowledge life cycle* – this timing cycle cues and synchronises *knowledge creation and transfer* activities with other timing cycles that are significant to IKNs.
  Proc-2 ↔ Tim-3
- *Knowledge creation and transfer rules* – impacts the way in which *knowledge creation and transfer* is approached and governed in an IKN.
  Proc-2 ↔ Mot-3.5

8.5.4.4.3 Innovation

This process flow definition model (C2, R2.2.3) defines *innovation* (Proc-3) as a process flow at an intermediate level of detail. The textual version of this model is presented in Table 8.10, and the visual version in Figure 8.17.

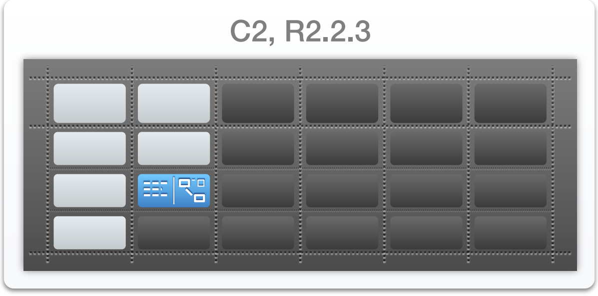


Table 8.10: Process flow definition model – Innovation (C2, R2.2.3)

	Code	Name / definition
A. Concept	**Proc-3**	**Innovation**

	Code	**Name / definition**
B. Definition		The successful generation, development and implementation of new and novel ideas, which introduce new or enhances current products, processes and/or strategies to a company, leading to commercial success and possible market leadership and creating value for stakeholders, driving economic growth and improving standards of living.
C. Deconstruction	Proc-3.1	Identification and evaluation
	Proc-3.2	Portfolio management
	Proc-3.3	Deployment and exploitation
D. Associations	Proc-1	Knowledge creation and transfer
	Proc-2.2	Network building
E. Integrations	Inv-1	Knowledge
	Inv-2	Innovation artefacts
	Inv-3	Resources
	Dist-1	Strategic network
	Dist-2	Operational network
	Resp-2	Innovators
	Resp-1	Network stakeholders
	Resp-3	Knowledge owners
	Tim-4	Innovation life cycle
	Mot-3	Innovation intention

The *innovation* (Proc-2) process flow is **defined** (refer to Table 8.10) (B) as:

> *The process of successful generation, development and implementation of new and novel ideas which introduce new or enhance current products, services, processes and/or strategies in a network, leading to commercial success and possible market leadership and creating value for stakeholders, driving economic growth and improving standards of living.*

The *innovation* process flow is **deconstructed** (refer to Table 8.10 (C), Figure 8.17 (1)) into three constituent transforms, namely *identification and evaluation*, *portfolio management,* and *deployment and exploitation*, via “part-of/has-part” relationships.

Proc-3 → Proc-3.1, Proc-3.2, Proc-3.3

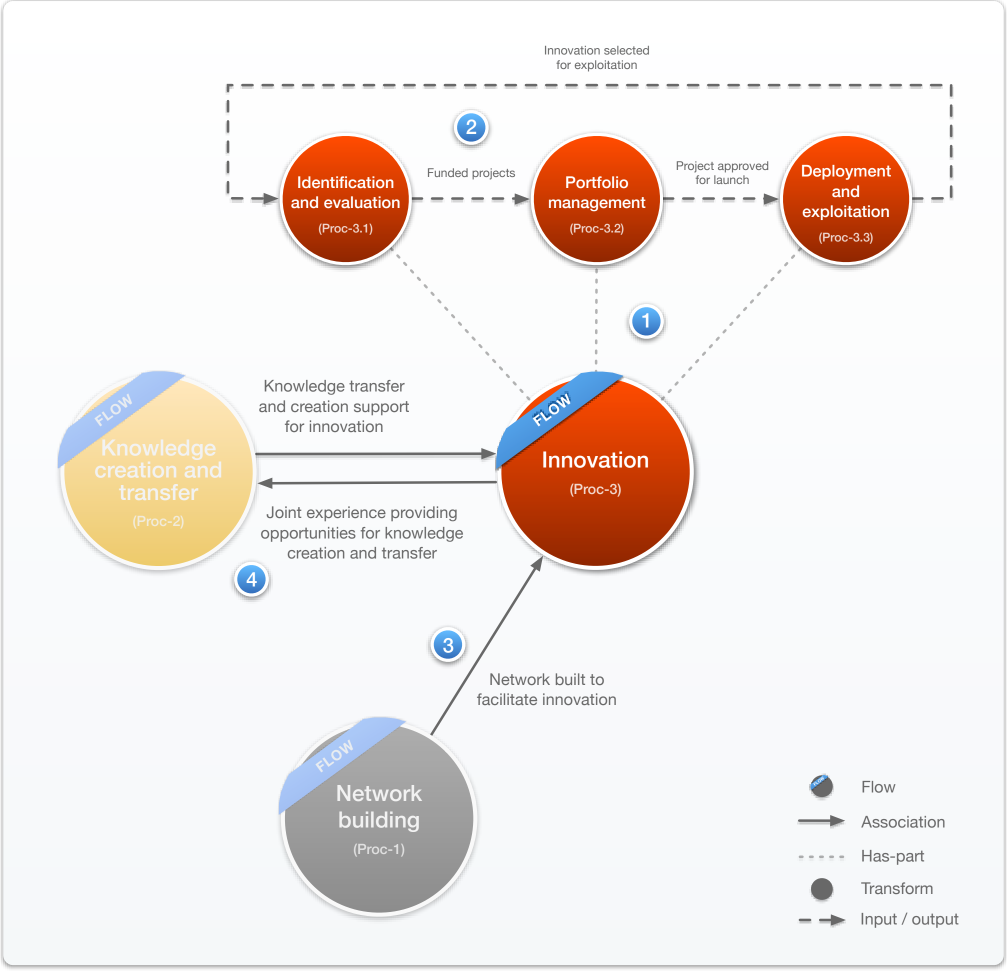


Figure 8.17: Process concept definition model – Innovation (C2, R2.2.3)

Identification and evaluation (Proc-3.1) is:

> *The convergent front-end or funnel of the innovation process flow, which includes activities related to idea generation and identification, concept definition, concept feasibility testing, and concept refinement.*

The collaborative execution of this transform in an IKN results in a strong integration with the *coordinator* role, with other roles that also form part of the *innovator* responsibility assignment also involved. These roles rely on *knowledge* and *resources* to create and manipulate various *ideas*, *concepts* and *prototypes* in this transform. The *identification and evaluation* transform provides funded projects to the *portfolio management* transform.

Portfolio management (Proc-3.2) is:

> *The holistic management of the network's innovation initiatives, which includes the prioritisation, scheduling and alignment of innovation projects. Portfolio management*

> *furthermore constitutes the allocation of resources, assignment of responsibility and the continuous monitoring of innovation initiatives, understanding the aggregate effect thereof, to ensure that the strategic objectives of the network are achieved.*

This transform integrates with the *coordinator* and *leader* roles that form part of the *innovator* responsibility assignment. These roles have to ensure that collaborative innovation *projects* are managed and coordinated in the IKN, and that these projects align with the network's *innovation intention*. The *portfolio management* transform provides innovation projects that are approved for launch to the *deployment and exploitation* transform.

Deployment and exploitation (Proc-3.3) is:

> *The divergent bugle of the innovation process flow, which includes activities related to the deployment, refinement, formalisation and exploitation of innovations.*

This transform also shows an integration with the *coordinator* role, given the challenges that come with the collaborative *deployment and execution* of an innovation *project* in an inter-organisational context. Other roles from the *innovator* responsibility assignment are also involved and rely on *knowledge* and *resources* to create and manipulate *projects, products* and *services*. The *deployment and exploitation* transform provides innovations that are selected for exploitation to the *identification and evaluation* transform.

Based on the preceding discussion, the constituent transforms of the *knowledge creation and transfer* (Proc-2) process flow share the following inputs/outputs (refer to Figure 8.17) (2):

- *Identification and evaluation* passing funded innovation projects to *portfolio management*.
- *Portfolio management* passing innovation projects that are approved for launch to *deployment and exploitation*.
- *Deployment and exploitation* passing innovations that are selected for exploitation to *identification and evaluation*.

Proc-3.1 → Proc-3.2

Proc-3.2 → Proc-3.3

Proc-3.3 → Proc-3.1

Innovation (Proc-3) and its constituent transforms also show **associations** (refer to Table 8.10) (D) with the following flows in the process abstraction (column):

- *Network building* (Proc-1; refer to Figure 8.17 (3)):
  - The provision of an IKN that is intentionally constructed to facilitate *innovation* (for more detail refer to section 8.5.4.4.1 on *network building*).

Proc-1 → Proc-3

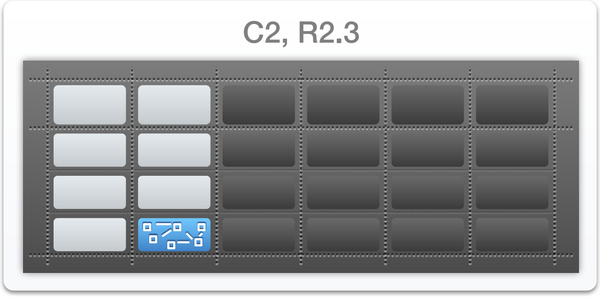
- *Knowledge creation and transfer* (Proc-2; refer to Figure 8.17 (4)):

- Execution of the *innovation* process flow in an IKN leads to joint experience amongst *knowledge owners* that provides opportunities for *knowledge creation and transfer*.
  Proc-3 → Proc-2
- *Knowledge creation and transfer* supports *innovation* by providing and transforming the required *knowledge* to execute the *innovation* process and thereby produce *innovation artefacts* (for more detail refer to section 8.5.4.4.2 on *knowledge creation and transfer*).
  Proc-2 → Proc-3

Innovation (Proc-3) and its constituent transforms furthermore show **integrations** (refer to Table 8.10) (E) with the following concepts in other abstractions (columns) of the reference architecture for IKNs:

- *Knowledge* – this inventory set provides the *information artefacts*, *explicit knowledge* and *tacit knowledge* required to execute the *innovation* process flow.
  Inv-1 ↔ Proc-3
- *Innovation artefacts* – this inventory set consists of the entities (ideas, *concepts*, *prototypes*, *projects*, *products* and *services*) that are created and manipulated during the *innovation* process flow.
  Proc-3 ↔ Inv-1
- *Resources* – this inventory set provides the *information systems*, *workspaces*, *equipment* and *funding* required to execute the *innovation* process flow.
  Inv-3 ↔ Proc-3
- *Strategic network* – locations impact *innovation* at a strategic level.
  Dist-1 ↔ Proc-3
- *Operational network* – locations impact *innovation* at an operational level.
  Dist-2 ↔ Proc-3
- *Innovators* – this responsibility assignment includes the *leader*, *networker*, *builder*, *coordinator* and *anthropologist* roles that execute the various transforms in the *innovation* process flow.
  Resp-2 ↔ Proc-3
- *Network stakeholders* – the *innovation* process flow is executed collaboratively in an IKN through the association between *network stakeholders* and *innovators.*
  Proc-3 ↔ (Resp-2 →) Resp-2
- *Knowledge owners* – this role is linked to the *tacit knowledge* that is required to execute the various transforms in the *innovation* process flow.
  Proc-3↔ (Inv-1.3 →) Resp-3
- *Innovation life cycle* – this timing cycle cues and synchronises *innovation* activities with other timing cycles that are significant to IKNs.
  Tim-4 ↔ Proc-3
- *Innovation intention* – this motivation intention includes the *network domain and purpose*, *innovation process rules* and *innovation role rules* that impact the way in which *innovation* is approached and governed in IKNs.
  Mot-3.5, Mot-3.6 ↔ Proc-3

8.5.4.5 Extended process definition model



The extended process definition model (C2, R2.3) defines the process flows identified in the process identification model (C2, R1; refer to Table 8.7) at a high level of detail by representing them visually according to the abstraction meta-model (refer to Figure 8.14). This model is based on the interpretation of the *definition data* from the qualitative systematic review described in section 8.5.2.1, with regard to the “*How?*” review question.

This model retains all the defining deconstructions and associations from the process flow definition models in the preceding sections. The various constituent transforms, however, are further deconstructed into sub-transforms and are organised into swim lanes according to their overarching process flows and transforms (refer to Figure 8.18). The detailed sub-inputs/outputs between these sub-transforms are also labelled appropriately.

In the *network building* (Proc-1) process flow, the following transforms are deconstructed into sub-transforms that are linked via sub-inputs/outputs (refer to Figure 8.18):

- *Network design* (Proc-1.1): A vision, strategy, domain and stakeholders are defined and passed onto a subsequent sub-transform that establishes the requirements for such a network. These requirements serve as the sub-input for the detail design and planning of an IKN.

- *Network implementation* (Proc-1.2): The implementation of the network is planned and prepared in an initial sub-transform, before this planned implementation is passed on as the input to a sub-transform that subsequently performs the actual network roll-out and implementation.

- *Network operation* (Proc-1.3): The management of stakeholders and facilitation of funding is performed in a first sub-transform, which passes this operational environment onto a second sub-transform that evolves, maintains and facilitates network activities. This sustained network is passed as an input/output to a third sub-transform that maintains the network architecture.

- *Network refinement* (Proc-1.4): The stakeholders and funding of the network is expanded in an initial sub-transform, which passes this expanded environment to a second sub-transform that measures the performance of the network and identifies key issues. This performance measurement serves as the input to another sub-transform, which enhances the network architecture based on the identified key issues.

- *Network phase-out* (Proc-1.5): The network phase-out is planned in a first sub-transform and passed to a second sub-transform that handles the closure and migration of the network architecture. This closed architecture is the input of a third sub-transform that documents the results and performance of the network, before a final sub-transform performs the contractual close-out of the IKN.

In the *knowledge creation and transfer* (Proc-2) process flow, the following transforms are deconstructed into sub-transforms that are linked via sub-inputs/outputs (refer to Figure 8.18):

- *Socialisation* (Proc-2.1): Tacit knowledge is captured through interaction in a first sub-transform. This captured tacit knowledge is the input to another transform where tacit knowledge is disseminated. Socialised tacit knowledge is then passed on as an input to the *externalisation* transform.

- *Externalisation* (Proc-2.2): Tacit knowledge is identified for externalisation in a first sub-transform, and serves as the input for a second sub-transform where this identified tacit knowledge is articulated. This articulated tacit knowledge is the input for a third sub-transform where it is translated into explicit knowledge, before becoming an input to the *systematisation* transform.

- *Systematisation* (Proc-2.3): Explicit knowledge is captured and integrated in a first sub-transform, and passed on as the input to a second sub-transform where it is disseminated. This disseminated explicit knowledge is the input for a third sub-transform where it is edited and processed, and then passed on as the input to the *internalisation* transform.

- *Internalisation* (Proc-2.4): Explicit knowledge is identified and understood in an initial sub-transform and passed on as processed explicit knowledge as an input to a subsequent sub-transform where it is embodied. Internalised tacit knowledge then becomes an input to the *socialisation* transform to restart the knowledge creation spiral.

In the *innovation* (Proc-3) process flow, the following transforms are deconstructed into sub-transforms that are linked via sub-inputs/outputs (refer to Figure 8.18):

- *Identification and exploitation* (Proc-3.1): Information is collected, categorised and presented in an initial sub-transform that passes this information to a second sub-transform where innovative ideas are generated and collected. These collected ideas are passed to a sub-transform that captures them, after which another sub-transform filters the captured ideas. Filtered ideas are developed into concepts in a subsequent sub-transform before these initial concepts are incubated and refined in another sub-transform. Refined concepts are filtered before their feasibility is determined through the development of prototypes in a sub-transform. These prototypes serve as the input to a sub-transform that further refines the concepts before ultimately being evaluated for the approval of funding and transformation into projects. Funded projects are then passed as an input to the *portfolio management* transform.

- *Portfolio management* (Proc-3.2): A portfolio of funded projects is developed and managed in a first sub-transform that passes this portfolio onto a next sub-transform that prepares innovation projects for launch. These prepared projects are passed to a final sub-transform where approval for a scheduled project launch is given, before these approved projects are passed as an input to the *deployment and exploitation* transform.

- *Deployment and exploitation* (Proc-3.3): Innovation projects are launched in a first sub-transform, after which launched projects are planned and also detail designed and tested in subsequent sub-transforms. These projects are approved for implementation in a next sub-transform, and then implemented in another sub-transform. Implemented projects are then operated, refined and formalised in a next sub-transform, after which certain operational innovation projects may be selected for exploitation. In such cases, a next sub-transform performs the exploitation of the business model, before the selected innovation is fed back as a possible input to the *identification and evaluation* transform.

To ensure that the model remains a primitive (single-variable) model, as well as for readability reasons, integrations with concepts in other abstractions (columns) are not indicated in the model.

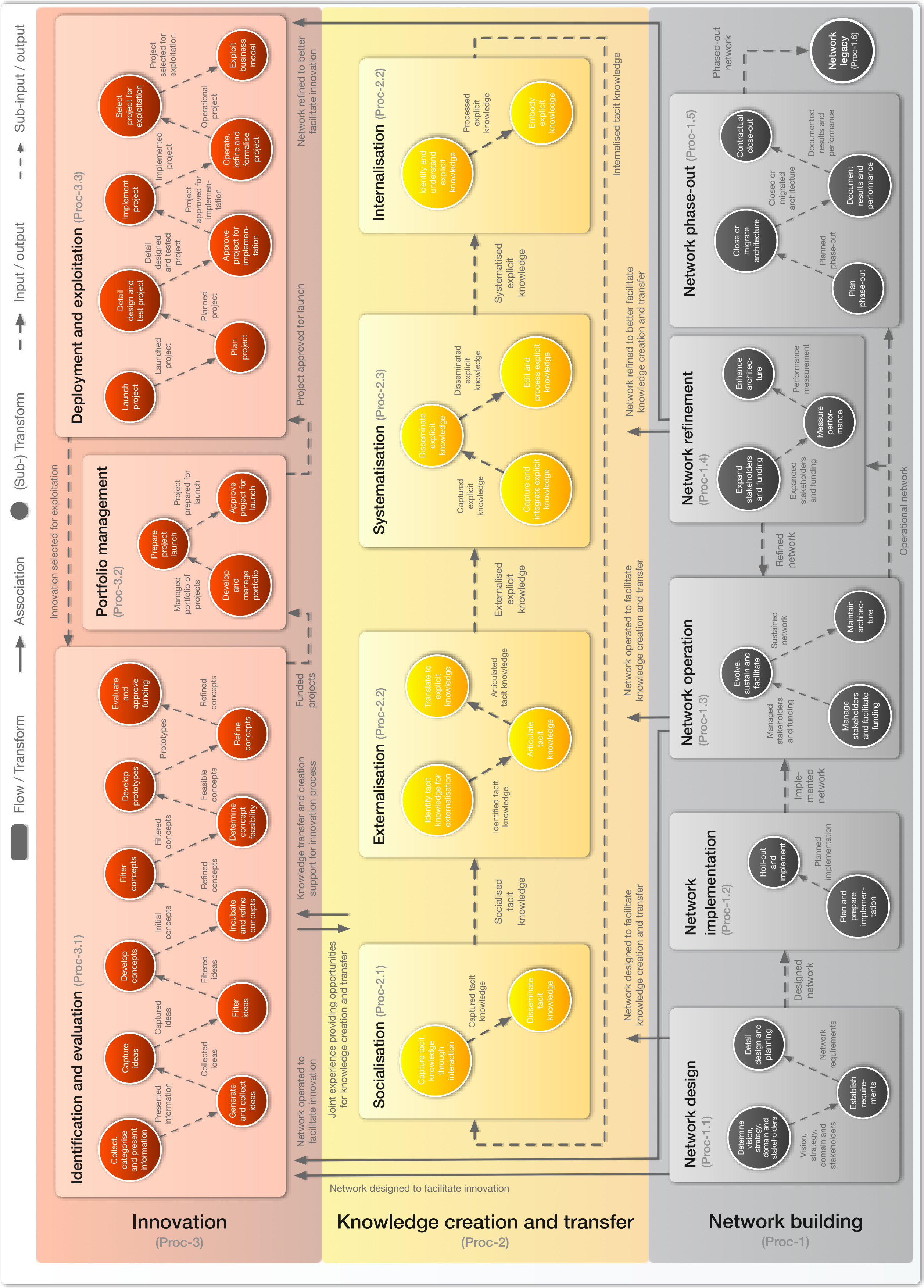
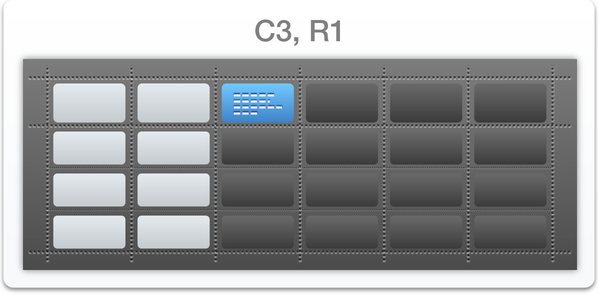


Figure 8.18: Extended process definition model

8.5.5 Distribution network models

This section discusses the distribution abstraction (third column) of the artefact. Therefore, the models presented answer the interrogative of "Where?" in the engineering of IKNs. The top-row reference model is presented and motivated first. The meta-model for the second row of the third column is then presented, followed by the various second-row reference models according to increasing levels of detail. Where the description of a model includes a reference to another formalised concept in the reference architecture, the name of the concept is printed in italics.

8.5.5.1 Distribution identification model



The distribution identification model (C3, R1; refer to Table 8.11) lists distribution types that are significant and common to the engineering of IKNs. This model is based on the interpretation of the *identification data* from the qualitative systematic review described in section 8.5.2.1, with regard to the "*Where?*" review question.

At the lowest level of detail, two main distribution types are identified, namely *strategic network* and *operational network*. Constituent distribution subtypes are also identified that classify these types at a higher level of detail.

Table 8.11: Distribution identification model (C3, R1)

Code	Distribution type	Code	Constituent distribution subtypes
Dist-1	**Strategic network**	Dist-1.1	International locations
		Dist-1.2	National locations
		Dist-1.3	Regional locations
		Dist-1.4	Local locations
Dist-2	**Operational network**	Dist-2.1	Network stakeholder locations
		Dist-2.2	Operational locations

The *strategic network* is a key distribution type, given that the geographical dispersion of the members and activities of an IKN from a strategic point of view has a large impact on its operation. The *strategic network* distribution type is deconstructed into distribution subtypes of *international locations*, *national locations*, *regional locations* and *local locations*.

The *operational network* is another key distribution type, as the operational geographical dispersion of the members and activities of an IKN also has an impact on its operation. The *operational network* distribution type is deconstructed into distribution subtypes of *network stakeholder locations* and *operational locations*.

Definitions for the distribution types and subtypes identified in the distribution identification model are provided in the second-row distribution definition models presented in sections 8.5.5.3, 8.5.5.4 and 8.5.5.5.

8.5.5.2 Meta-model for definition models in distribution abstraction

The primitive meta-model presented in Figure 8.19 serves as the semi-formal representational language for the various second-row definition models in the distribution abstraction, or '*where*' column, of the reference architecture for IKNs.

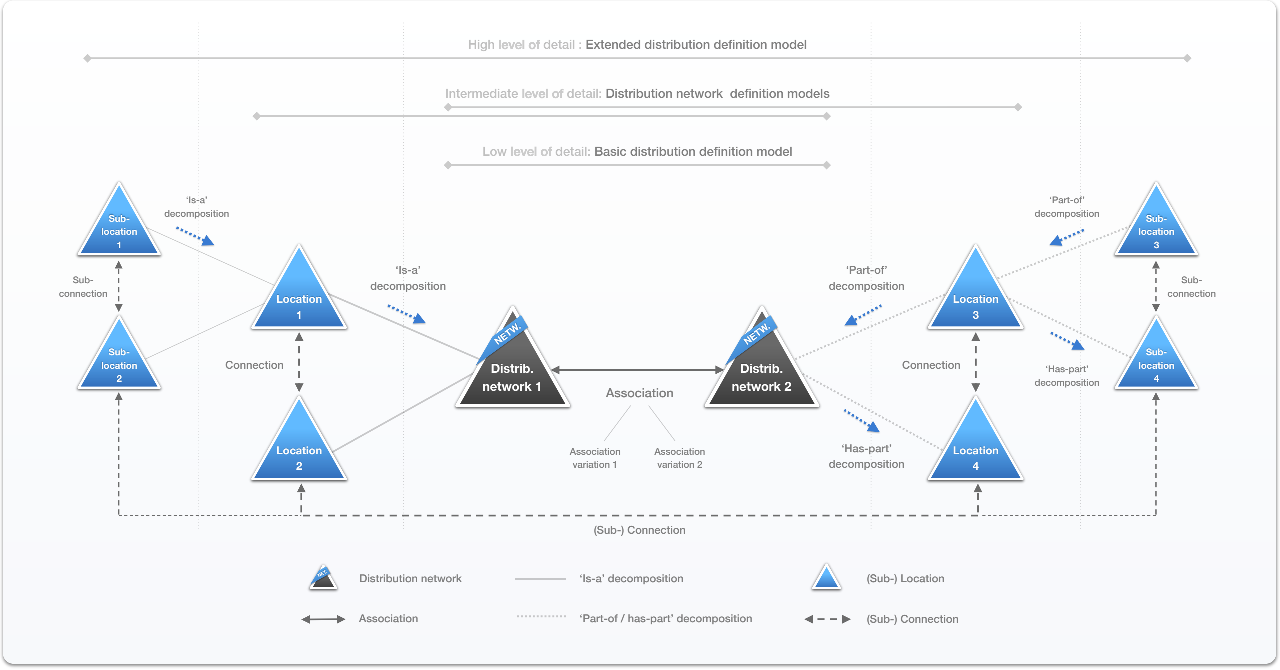
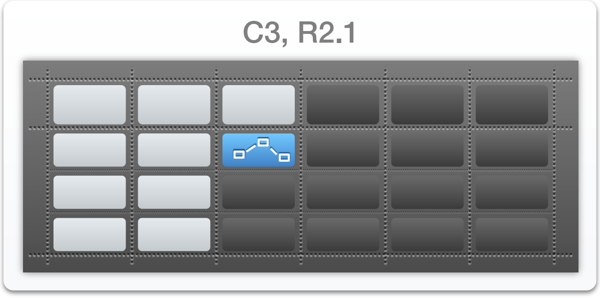


Figure 8.19: Meta-model for definition models in distribution abstraction

This meta-model assists in defining *distribution networks* as concepts and follows the single-variable structure of "*location-connection-location*" amongst its elements, links, sub-elements and sub-links. The meta-model therefore deconstructs and represents distribution networks into locations and sub-locations, along with their respective connections and sub-connections.

This representation is done at various levels of detail in the basic distribution definition model (refer to section 8.5.5.3), distribution network definition models (refer to section 8.5.5.4) and extended distribution definition model (refer to section 8.5.5.5). For all of these models, the meta-model scope indication shown at the top of Figure 8.19 applies.

8.5.5.3 Basic distribution definition model



The basic distribution definition model (C3, R2.1) defines the distribution concepts that were identified as distribution types in the distribution identification model (C3, R1; refer to Table 8.11) at a low level of detail by representing them visually according to the abstraction meta-model (refer to Figure 8.19). This model is based on the interpretation of the *definition data* from the qualitative systematic review described in section 8.5.2.1, with regard to the “*Where?*” review question.

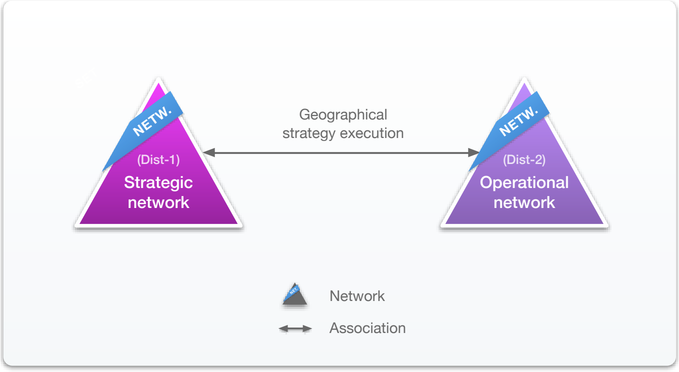


Figure 8.20: Basic distribution definition model (C3, R2.1)

The model depicts the identified distribution networks, as well as the association between them (refer to Figure 8.20). This model shows that a *strategic network* and an *operational network* are connected through the geographical strategy execution of an IKN. For this reason the way in which the *strategic network* of an IKN is configured impacts the way in which its *operational network* is configured. The *operational network* of an IKN is therefore a realisation of its *strategic network.*

These distribution networks are now defined in more detail in the relevant distribution network definition models in section 8.5.5.4.

8.5.5.4 Distribution network definition models

The distribution network definition models presented in this section each defines an identified distribution network at an intermediate level of detail. The models therefore add new information to each network, while retaining all the information from the basic distribution definition model (refer to Figure 8.20). Both textual and visual versions are provided to assist in defining the networks. Deconstructions, relationships and integrations between concepts are expressed in terms of identification codes in side-notes for traceability purposes.

The models presented in this section are based on the interpretation of the *definition data* for each identified distribution network, as obtained from the qualitative systematic review described in section 8.5.2.1.

8.5.5.4.1 Strategic network

This distribution network definition model (C3, R2.2.1) defines *strategic network* (Dist-1) as a distribution network at an intermediate level of the detail. The textual version of this model is initially presented in Table 8.12 and the visual version in Figure 8.21.

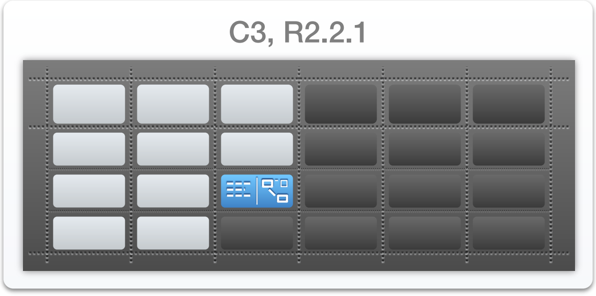


Table 8.12: Distribution network definition model – Strategic network (C3, R2.2.1)

	Code	**Name / definition**
A. Concept	**Dist-1**	**Strategic network**
B. Definition		A network of locations that are architecturally relevant to the network from a strategic perspective.
C. Deconstruction	Dist-1.1	International locations
	Dist-1.2	National locations
	Dist-1.3	Regional locations
	Dist-1.4	Local locations
D. Associations	Dist-2	Operational network
E. Integrations	Proc-1	Network building
	Proc-2	Knowledge creation and transfer
	Proc-3	Innovation
	Resp-1	Network stakeholders
	Mot-3.8	Geographical strategy

The *strategic network* distribution network is **defined** (refer to Table 8.12) (B) as:

> *A network of locations that are architecturally relevant to the network from a strategic perspective.*

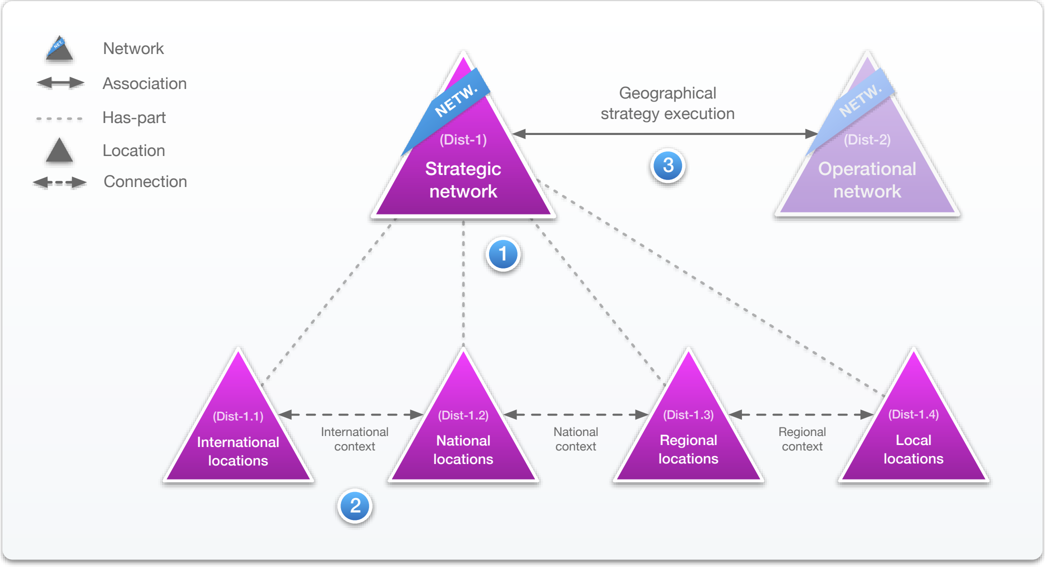


Figure 8.21: Distribution network definition model – Strategic network (C3, R2.2.1)

The *strategic network* distribution network is deconstructed (refer to Table 8.12 (C), Figure 8.21 (1)) into four constituent locations, namely *international locations*, *national locations*, *regional locations*, and *local locations*, via “part-of/has-part” relationships.

Dist-1 → Dist-1.1, Dist-1.2, Dist-1.3, Dist-1.4

International locations (Dist-1.1) are:

> *Locations that are characterised by their international position, i.e. on a global or worldwide scale.*

International locations are not bound by their locality in any given country and therefore use the entire globe as their point of reference. *International locations* provide international context to *national locations*.

National locations (Dist-1.2) are:

> *Locations that are characterised by their national position, i.e. on a countrywide scale.*

National locations are bound by their locality in a particular country and therefore use this country as a point of reference. *National locations* are connected to *international locations* through international context that relates this point of reference to a larger scope, while providing national context to *regional locations*.

Regional locations (Dist-1.3) are:

> *Locations that are characterised by their regional position, i.e. on a provincial or state-wide scale.*

Regional locations are bound by their locality in a particular region and therefore use this region as a point of reference. *Regional locations* are connected to *national locations* through national context that relates this point of reference to a larger scope, while providing regional context to *local locations*.

Local locations (Dist-1.4) are:

> *Locations that are characterised by their local position, i.e. on a citywide or even office wide scale.*

Local locations are bound by their locality in a particular city or even a particular building and therefore use this local area as a point of reference. *Local locations* are connected to *regional locations* through regional context that relates this point of reference to a larger scope.

Based on the preceding discussion, the constituent locations of the *strategic network* (Dist-1) distribution network share the following connections (refer to Figure 8.21) (2):

- *International locations* provide international context to *national locations.*

Dist-1.1 → Dist-1.2

- *National locations* provide national context to *regional locations.*

Dist-1.2 → Dist-1.3

- *Regional locations* provide regional context to *local locations.*

Dist-1.3 → Dist-1.4

The *strategic network* (Dist-1) and its constituent locations also show an **association** (refer to Table 8.12) (D) with a network in the distribution abstraction (column):

- *Operational network* (Dist-2; refer to Figure 8.21 (3)):
    - The *strategic network* and *operational network* of an IKN are connected through geographical strategy execution.

Dist-1 ↔ Dist-2

The *strategic network* (Dist-1) and its constituent locations furthermore show **integrations** (refer to Table 8.12) (E) with the following concepts in other abstractions (columns) of the reference architecture for IKNs:

- *Network building* – *international locations*, *national locations*, *regional locations* and *local locations* impact *network building* decisions on a strategic level, and vice versa.

Dist-1.1, Dist-1.2, Dist-1.3, Dist-1.4 ↔ Proc-1

- *Knowledge creation and transfer* – *international locations*, *national locations*, *regional locations* and *local locations* impact *knowledge creation and transfer* on a strategic level.

Dist-1.1, Dist-1.2, Dist-1.3, Dist-1.4 ↔ Proc-2

- *Innovation*; *international locations*, *national locations*, *regional locations* and *local locations* impact an IKNs approach to execution of the *innovation* process flow on a strategic level.

Dist-1.1, Dist-1.2, Dist-1.3, Dist-1.4 ↔ Proc-3

- *Network stakeholders*– the geographical distribution of *network stakeholders*

Dist-1 (→ Dist-2.1) ↔ Resp-1

via their *network stakeholder locations* in the *strategic network* impacts various IKN operations.

- *Geographical strategy* – this motivational end of an IKN determines the way in which its *strategic network* is managed and governed.

Dist-1 ↔ Mot-3.8

8.5.5.4.2 Operational network

This distribution network definition model (C3, R2.2.2) defines an *operational network* (Dist-2) as a distribution network at an intermediate level of the detail. The textual version of this model is presented in Table 8.13, and the visual version in Figure 8.22.

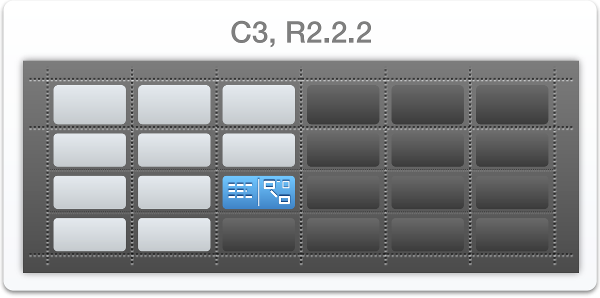


Table 8.13: Distribution network definition model – Operational network (C3, R2.2.2)

	Code	**Name / definition**
A. Concept	**Dist-2**	**Operational network**
B. Definition		A network of locations that are architecturally relevant to the network from an operational perspective.
C. Deconstruction	Dist-2.1	Network stakeholder locations
	Dist-2.2	Operational locations
D. Associations	Dist-1	Strategic network
E. Integrations	Inv-1	Knowledge
	Inv-2	Innovation artefacts
	Inv-3	Resources
	Proc-1	Network building
	Proc-2	Knowledge creation and transfer
	Proc-3	Innovation
	Resp-1	Network stakeholders
	Mot-3.8	Geographical strategy

The *operational network* (Dist-2) distribution network is **defined** (refer to Table 8.13) (B) as:

> *A network of locations that are architecturally relevant to the network from an operational perspective.*

The *operational network* distribution network is deconstructed (refer to Table 8.13 (C), Figure 8.16 (1)) into two constituent locations, namely *network stakeholder locations* and *operational locations*, via “part-of/has-part” relationships.

Dist-2 →
Dist-2.1, Dist-2.2

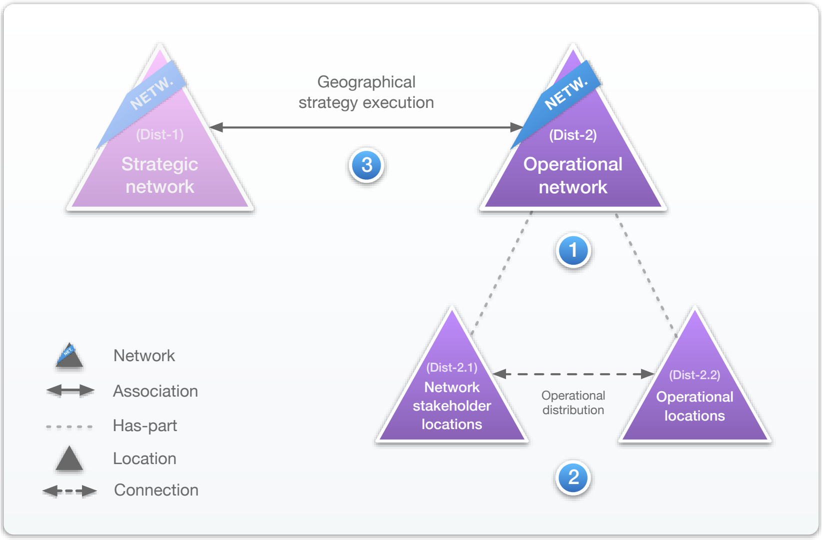


Figure 8.22: Distribution network definition model – Operational network (C3, R2.2.2)

Network stakeholder locations (Dist-2.1) are:

> *The geographical positions of individuals or organisations that are stakeholders in the IKN.*

The *network stakeholder locations* of an IKN form an important part of its *operational network*, as this is where the bulk of its operations will occur, e.g. the *knowledge creation and transfer* and *innovation* process flows.

Operational locations (Dist-2.2) are:

> *The geographical positions where activities that are related to the IKN are performed.*

The *operational locations* of an IKN are usually connected to its *network stakeholder locations* through its *operational distribution*, as most of the activities that are related to the IKN are performed at locations where its *network stakeholders* are situated. This, however, is not mandatory, and *operational locations* may exist that are significant to the IKN, but that are not co-located with a *network member location*. An example of this is a location where the *products* or *services* of an IKN are consumed, but where the corresponding member of the market that is performing this action, is not a *network member*.

Based on the preceding discussion, the constituent locations of the *operational network* (Dist-2) distribution network share the following connection (refer to Figure 8.22) (2):

- *Network stakeholder locations* and *operational locations* align according the operational distribution of an IKN.

Dist-2.1 ↔ Dist-2.2

The *operational network* (Dist-2) and its constituent locations show an **association** (refer to Table 8.9) (D) with a network in the distribution abstraction (column):

- *Strategic network* (Dist-1; refer to Figure 8.22 (4)):
  - The *operational network* and *strategic network* of an IKN are connected through geographical strategy execution.

Dist-2 ↔ Dist-1

The *operational network* (Dist-2) and its constituent locations furthermore show **integrations** (refer to Table 8.9) (E) with the following concepts in other abstractions (columns) of the reference architecture for IKNs:

- *Knowledge*; through the hosting of *information artefacts* and *explicit knowledge* at these locations.

Inv-1.1, Inv-1.2 ↔ Dist-2.1

- *Innovation artefacts*; *Prototypes* and *projects* may be hosted at *network stakeholder locations*, with other *innovation artefacts* potentially also being located at other *operational locations*.

Inv-2 ↔ Dist-2.1, Dist-2.2

- *Resources* – *information systems*, *workspaces* and *equipment* may be located at either *network stakeholder locations* or other *operational locations*.

Inv-3.1, Inv-3.2, Inv-3.2 ↔ Dist-2.1, Dist-2.2

- *Network building* – this process flow may lead to the addition of new *network stakeholders*, and therefore also to new *network stakeholder locations*.

Proc-3 ↔ (Resp-1 →) Dist-2.1

- *Knowledge creation and transfer* – this process flow is executed at *network stakeholder locations* and *operational locations*.

Dist-2.1, Dist-2.2 ↔ Proc-2

- *Innovation* – this process flow is executed at *network stakeholder locations* and *operational locations*.

Dist-2.1, Dist-2.2 ↔ Proc-3

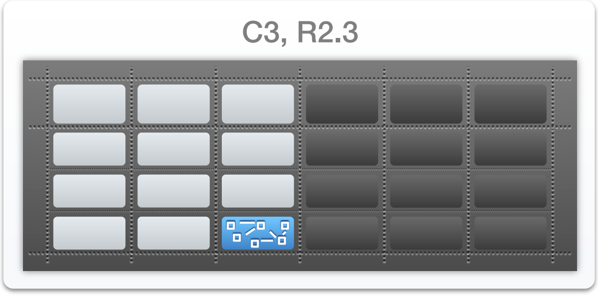
- *Network stakeholders* – individuals or organisations with this responsibility assignment are situated at *network stakeholder locations*.

Dist-2.1 ↔ Resp-1

- *Geographical strategy* – this motivational end of an IKN determines the way in which its *operational network* is managed and governed.

Dist-2 ↔ Mot-3.8

8.5.5.5 Extended distribution definition model



The extended distribution definition model (C3, R2.3) defines the distribution locations identified in the distribution identification model (C3, R1; refer to Table 8.11) at a high level of detail by representing them visually according to the abstraction meta-model (refer to Figure 8.19). This model is based on the interpretation of the *definition data* from the qualitative systematic review described in section 8.5.2.1, with regard to the “*Where?*” review question.

This model retains all the defining deconstructions and associations from the distribution network definition models in the preceding sections. Where applicable, however, the various constituent locations are further deconstructed into sub-locations and are organised into swim lanes according to their overarching network and location (refer to Figure 8.23). The detailed sub-connections between these sub-locations are also labelled appropriately.

In the *operational network* (Dist-2) process flow, the following locations are deconstructed into sub-locations that are linked via sub-connections (refer to Figure 8.23):

- *Network stakeholder locations* (Proc-2.1): These locations are deconstructed into government locations, research institution locations, intermediary locations, industry enterprise locations and market locations according to the type of *network stakeholder* that is situated at the particular location.

  Government locations are connected to intermediary locations with mechanisms for knowledge creation and transfer between these locations. These mechanisms could be in the form of an information system, as well as scheduled time for face-to-face contact to facilitate *socialisation*. Government locations are also connected to intermediary locations with regulations and incentives that specifically pertain to intermediaries due to their location within government jurisdiction. Government locations are connected to research institution locations with similar mechanisms for knowledge creation and transfer between these locations, as well as regulations and incentives that specifically pertain to research institutions due to their location within government jurisdiction. Similar regulations and incentives connect government locations to industry enterprise locations and market locations.

  Research institution locations are connected to intermediary locations and industry enterprise locations with mechanisms for knowledge creation and transfer between these locations. Research institution locations are also connected to each other with similar mechanisms. Intermediary

> locations are additionally connected to industry enterprise locations and market locations with similar mechanisms for knowledge creation and transfer.
>
> Industry enterprise locations are connected to each other with mechanisms for knowledge creation and transfer, as well as mechanisms for resource sharing. Industry enterprise locations are furthermore connected to market locations through product and/or service delivery, as well as mechanisms for knowledge creation and transfer.
>
> Government locations, research institution locations, intermediary locations, industry enterprise locations and market locations are all connected to *operational locations* through the way in which they impact the locations where the activities of the IKN occur.

To ensure that the model remains a primitive (single-variable) model, as well as for readability reasons, integrations with concepts in other abstractions (columns) are not indicated in the model.

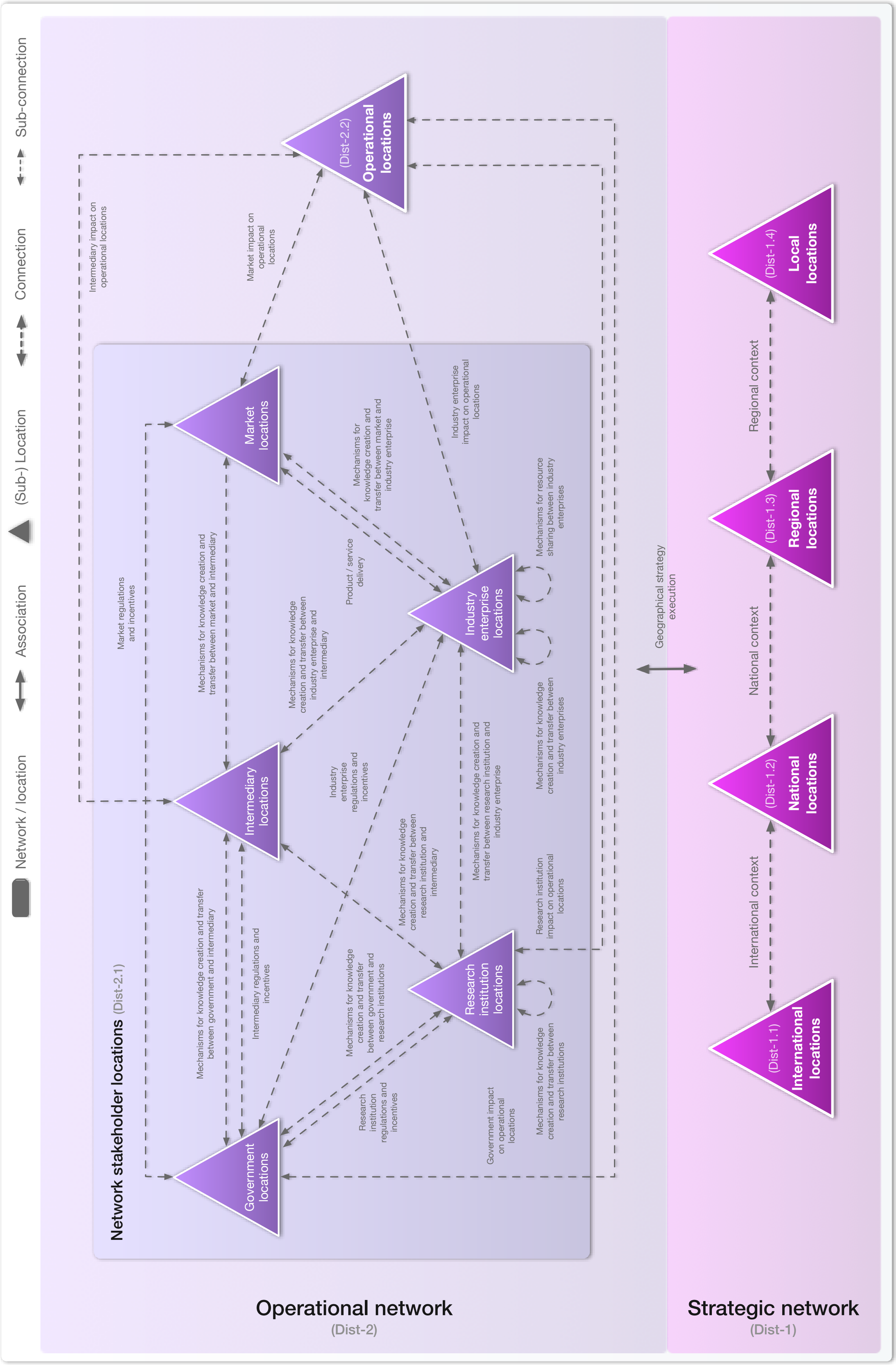


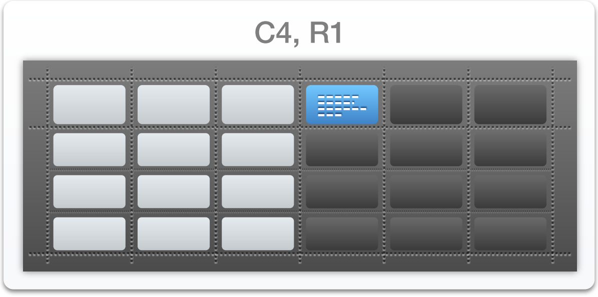
Figure 8.23: Extended distribution definition model (C3, R2.3)

8.5.6 Responsibility assignment models

This section discusses the responsibility abstraction (fourth column) of the artefact, and the models presented therefore answer the interrogative of "Who?" in the engineering of IKNs. The top-row reference model is presented and motivated first. Thereafter the meta-model for the second row of the fourth column is presented, followed by the various second-row reference models according to increasing levels of detail. Where the description of a model includes a reference to another formalised concept in the reference architecture, the name of the concept is printed in italics.

8.5.6.1 Responsibility identification model

The responsibility identification model (C4, R1; refer to Table 8.14) lists responsibility types that are significant and common to the engineering of IKNs. This model is based on the interpretation of the *identification data* from the qualitative systematic review described in section 8.5.2.1, with regard to the "*Who?*" review question.



At the lowest level of detail, three main responsibility types are identified, namely *network stakeholders*, *innovators* and *knowledge owners*. Constituent responsibility subtypes are also identified that classify these types at a higher level of detail.

Table 8.14: Responsibility identification model (C4, R1)

Code	**Responsibility type**	**Code**	**Constituent responsibility subtypes**
Resp-1	**Network stakeholders**	Resp-1.1	Industry enterprises
		Resp-1.2	Intermediaries
		Resp-1.3	Research institutions
		Resp-1.4	Market
		Resp-1.5	Government
Resp-2	**Innovators**	Resp-2.1	Leaders
		Resp-2.2	Networkers
		Resp-2.3	Builders
		Resp-2.4	Coordinators
		Resp-2.5	Anthropologists

Code	**Responsibility type**	**Code**	**Constituent responsibility subtypes**
Resp-3	**Knowledge owners**		

Network stakeholders are a key responsibility type for IKNs that describe the types of organisations and/or individuals that are involved in such networks. This responsibility assignment is therefore a fundamental characteristic of IKNs and helps to distinguish them from other kinds of collaborative networked organisations (CNOs). The *network stakeholders* responsibility type is deconstructed into responsibility subtypes of *industry enterprises*, *intermediaries*, *research institutions*, *market* and *government*.

The *innovators* responsibility type describes the types of responsibilities that are specifically connected to the *innovation* process (Proc-3, refer to section 8.5.4.4.3), the *innovation life cycle* (Tim-4, refer to section 8.5.7.4.4) and the enabling environment that surrounds them. The *innovators* responsibility type is deconstructed into responsibility subtypes of *leaders*, *networkers*, *builders*, *coordinators* and *anthropologists*. The relevance and involvement of these responsibility subtypes vary throughout the *innovation* process and *innovation life cycle*.

The *knowledge owners* responsibility type is connected to the *knowledge* (Inv-1) inventory set, and especially the *tacit knowledge* (Inv-1.3) inventory entity. *Knowledge owners* are individuals and/or organisations that possess certain *tacit knowledge* and are therefore crucial to the activities of an IKN, especially the *knowledge creation and transfer* process (Proc-2, refer to section 8.5.4.4.2). The *knowledge owners* responsibility type is not deconstructed into responsibility subtypes.

Definitions for the responsibility types and subtypes identified in the responsibility identification model are provided in the second-row responsibility definition models presented in sections 8.5.6.3, 8.5.6.4 and 8.5.6.5.

8.5.6.2 Meta-model for definition models in responsibility abstraction

The primitive meta-model presented in Figure 8.24 serves as the semi-formal representational language for the various second-row definition models in the responsibility abstraction, or '*who*' column, of the reference architecture for IKNs.

This meta-model assists in defining *responsibility assignments* as concepts and follows the single-variable structure of "*role-work/product-role*" amongst its elements, links, sub-elements and sub-links. The meta-model therefore deconstructs and represents responsibility assignments into roles and sub-roles, along with their respective work products and sub-work products.

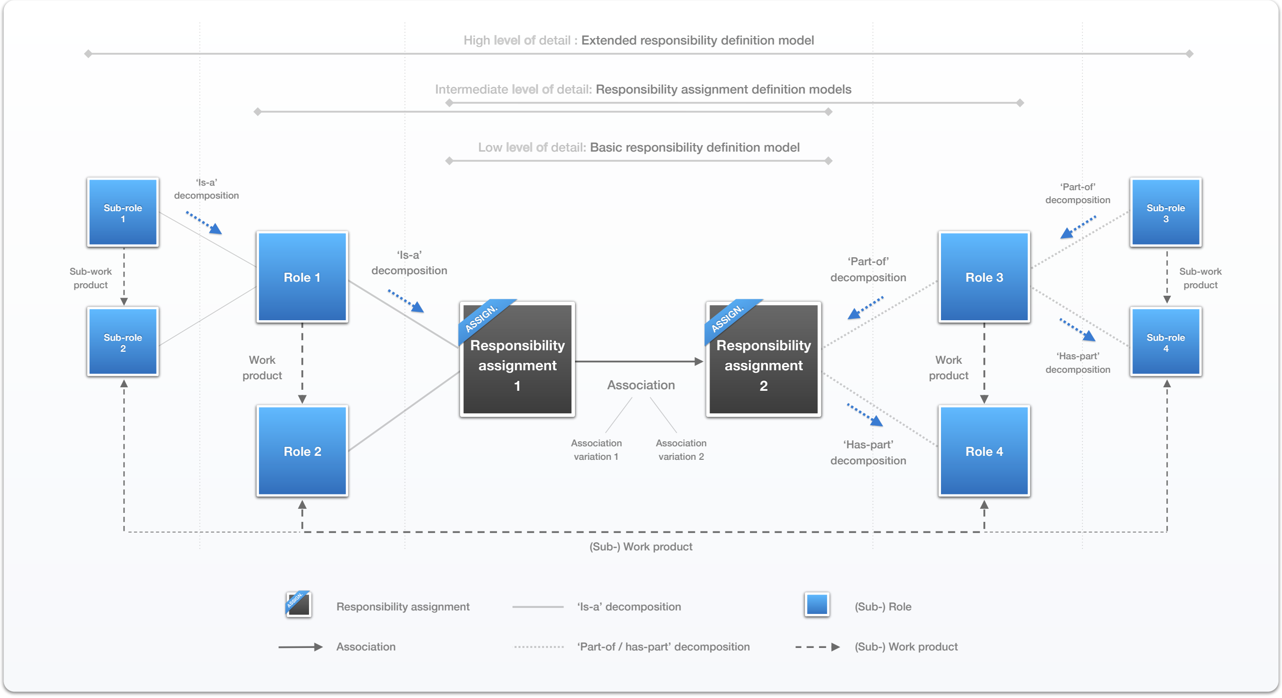
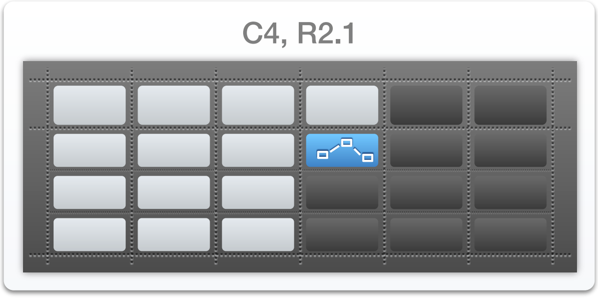


Figure 8.24: Meta-model for definition models in responsibility abstraction

This representation is done at various levels of detail in the basic responsibility definition model (refer to section 8.5.6.3), responsibility assignment definition models (refer to section 8.5.6.4) and extended responsibility definition model (refer to section 8.5.6.5). For all of these models, the meta-model scope indication shown at the top of Figure 8.24 applies.

8.5.6.3 Basic responsibility definition model



The basic responsibility definition model (C4, R2.1) defines the responsibility assignments that were identified as responsibility types in the responsibility identification model (C4, R1; refer to Table 8.14) at a low level of detail by representing them visually according to the abstraction meta-model (refer to Figure 8.24). This model is based on the interpretation of the *definition data* from the qualitative systematic review described in section 8.5.2.1, with regard to the "*Who?*" review question.

The model depicts the various identified responsibility assignments, as well as the associations between them (refer to Figure 8.25). This model shows that *network stakeholders* participate in the activities of an IKN by being assigned a responsibility as an *innovator.* Furthermore, organisations and/or individuals that

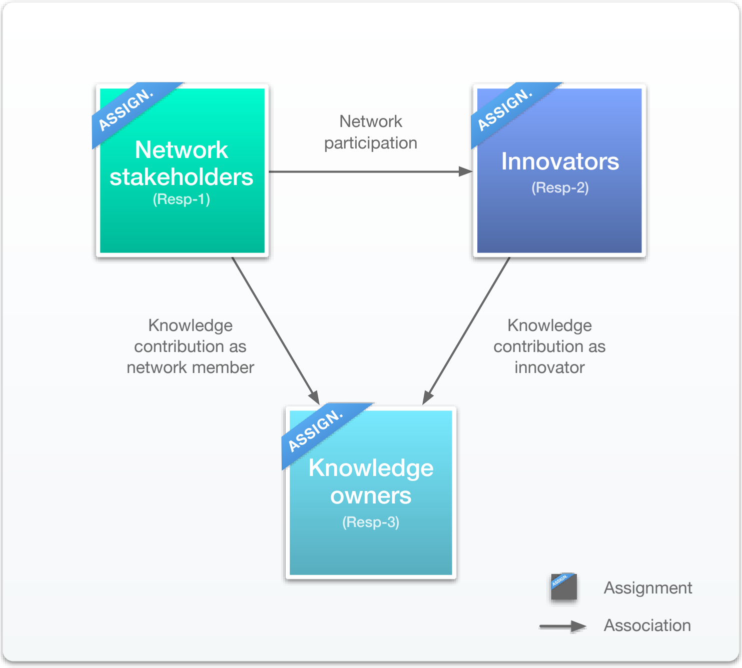


Figure 8.25: Basic responsibility definition model (C4, R2.1)

are involved in the IKN contribute knowledge as *knowledge owners* in either their capacity as *network stakeholders* or *innovators*.

These responsibility assignments are now defined in more detail in the applicable responsibility assignment definition models in section 8.5.6.4.

8.5.6.4 Responsibility assignment definition models

The responsibility assignment definition models presented in this section each define an identified responsibility assignment at an intermediate level of detail. The models therefore add new information for each assignment, while retaining all the information from the basic responsibility definition model (refer to Figure 8.25). Both textual and visual versions are provided to assist in defining the assignments. Deconstructions, relationships and integrations between concepts are expressed in terms of identification codes in side-notes for traceability purposes.

The models presented in this section are based on the interpretation of the *definition data* for each identified responsibility assignment, as obtained from the qualitative systematic review described in section 8.5.2.1.

8.5.6.4.1 Network stakeholders

This responsibility assignment definition model (C4, R2.2.1) defines *network stakeholders* (Resp-1) as a responsibility assignment at an intermediate level of the detail. The textual version of this model is initially presented in Table 8.15 and the visual version in Figure 8.26.

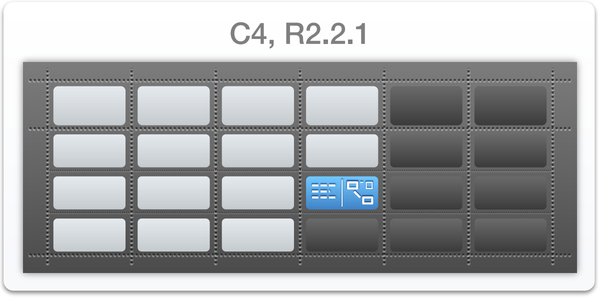


Table 8.15: Responsibility assignment definition model – Network stakeholders (C4, R2.2.1)

	Code	Name / definition
A. Concept	**Resp-1**	**Network stakeholders**
B. Definition		Organisations and individuals that participate in the integrated knowledge network.
C. Deconstruction	Resp-1.1	Industry enterprises
	Resp-1.2	Intermediaries
	Resp-1.3	Research institutions
	Resp-1.4	Market
	Resp-1.5	Government
D. Associations	Resp-2	Innovators
	Resp-3	Knowledge owners
E. Integrations	Inv-2	Innovation artefacts
	Inv-3	Resources
	Proc-1	Network building
	Dist-1	Strategic network
	Dist-2	Operational network
	Tim-2	Network life cycle
	Mot-3.4	Network building rules

The *network stakeholders* responsibility assignment is **defined** (refer to Table 8.15) (B) as:

> *Organisations and individuals that participate in the integrated knowledge network.*

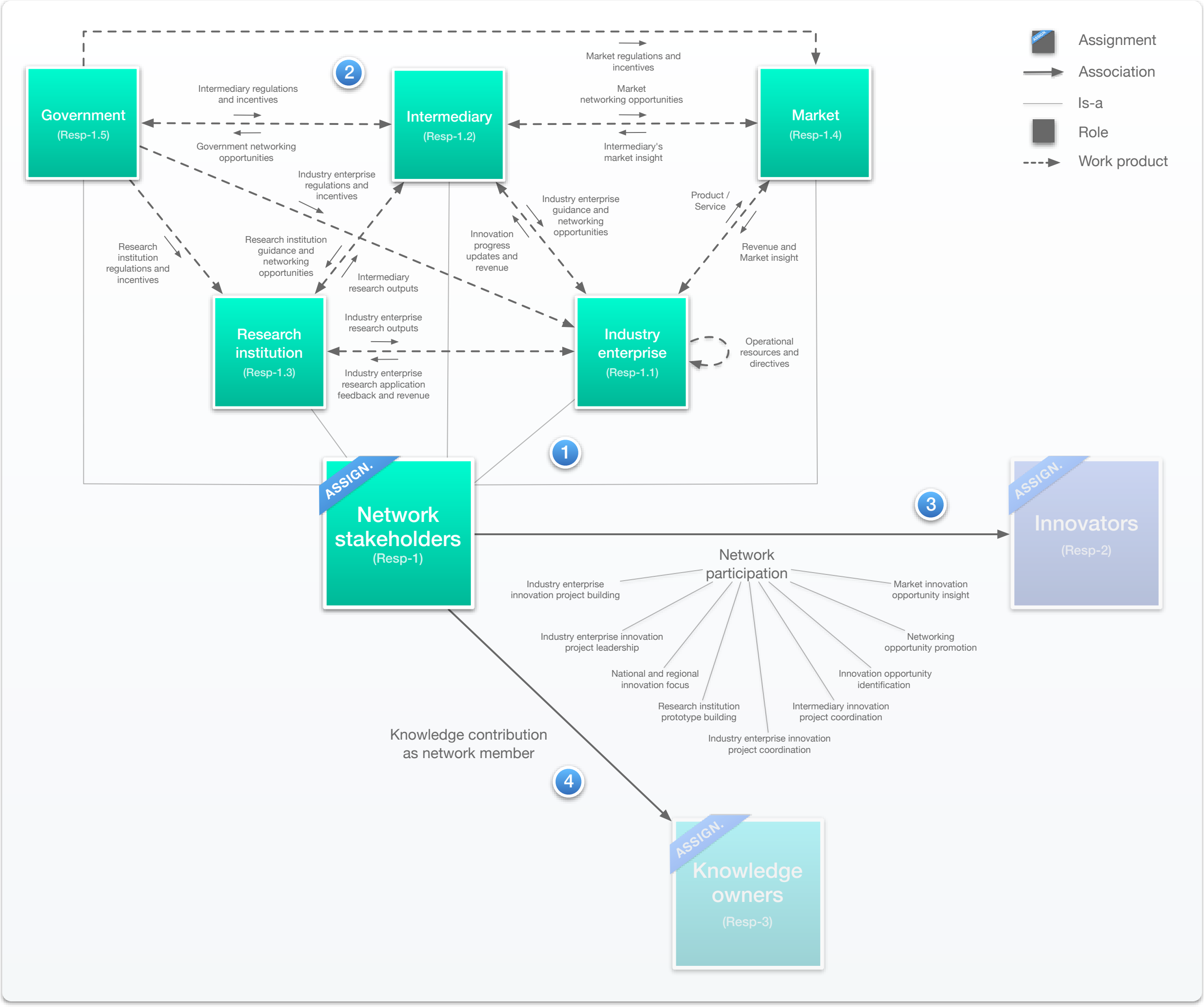


Figure 8.26: Responsibility assignment definition model – Network stakeholders (C4, R2.2.1)

The *network stakeholders* responsibility assignment is **deconstructed** (refer to Table 8.15 (C), Figure 8.26 (1)) into five constituent roles, namely *industry enterprises*, *intermediaries*, *research institutions*, *market* and *government*, via "is-a" relationships.

Resp-1 → Resp-1.1, Resp-1.2, Resp-1.3, Resp-1.4, Resp-1.5

Industry enterprises (Resp-1.1) are:

> *Complex systems of cultural, process, and technological components that interact to accomplish strategic goals in the private sector; under the ownership or control of an organisation; which ultimately strives to create value for its stakeholders; and operates at one or several locations.*

The role of *industry enterprises* can also be understood under the synonyms of "companies" or "firms" as used in other literature. In general, the intention is to refer to organisations that are delivering some kind of product or service to the *market* in order to generate value for their stakeholders. In the context of IKNs, *industry enterprises* also provide *research institutions* with feedback on the application of their research, and *intermediaries* with feedback on their innovative progress.

When collaborating in an IKN, *industry enterprises* also provide each other with operational resources and directives. As *innovators*, *industry enterprises* can be *leaders*, *builders*, *coordinators* and *anthropologists*. *Industry enterprises* are also required to act as *knowledge owners* in IKNs.

Intermediaries (Resp-1.2) are:

> *Boundary organisations that help to facilitate and coordinate the flow of information and technology transfer in an inter-organisational network. These actors fill gaps in information and knowledge in these networks by facilitating the exchange of information about innovation amongst companies. Intermediaries provide "surrogate ties" by serving as functional substitutes for the lack of "bridging ties" between network stakeholders.*

Intermediaries provide networking opportunities to *industry enterprises*, *research institutions*, *government* and the *market.* In addition to these networking opportunities, *intermediaries* also provide *industry enterprises* with guidance related to research outputs that are available for commercialisation, i.e. advanced phases of the *innovation* process. *Intermediaries* furthermore provide *research institutions* with guidance related to issues and problems arising in industry that are in need of research to provide solutions.

As *innovators*, *intermediaries* can act as *networkers* through the promotion of networking opportunities, and *coordinators* through their coordination of collaborative innovation *projects*. In this capacity, *intermediaries* are often also the custodians or managers of shared intellectual property developed in IKNs.

Research institutions (Resp-1.3) are:

> *Establishments endowed for doing research, which may specialise in basic research or may be oriented to applied research.*

Research institutions provide research outputs to *industry enterprises* and *intermediaries*, thereby executing activities that are associated with the initial phases of the *innovation* process. These research outputs may be commercialised by *industry enterprises*, or become the subject of networking and *network building* initiated by *intermediaries* in order to find an actor that has the capacity to perform the commercialisation process.

In terms of their contribution as *innovators*, *research institutions* can be associated with the *builder* role by transforming *ideas* into tangible and refined *concepts* and *prototypes*.

The *market* (Resp-1.4) is:

> *Actors that constitute a demand for a particular product or service.*

The *market* is seen as the end-users of innovations, and as such can provide insight into their requirements to *industry enterprises* and *intermediaries*. In cases where the nature of these innovations are commercial, the *market*'s acquisition and consumption of products and services results in revenue that is initially received by *industry enterprises*, from where these *funds* may be redistributed in accordance to contractual agreement in the IKN.

As *innovators*, the *market* can be seen as an *anthropologist* that provides insight into how people interact physically and emotionally with products, services, one another and their environment.

Government (Resp-1.5) is:

> *The group of people with the authority to govern a domain under their jurisdiction, e.g. a country, or a state, thereby constituting the system by which this domain is governed.*

A *government* can be directly or indirectly involved in an IKN by stimulating *innovation* through the provision of regulations and incentives to other organisations involved in the network. A *government* can be seen as a *leader* in an innovative context by providing a national and regional research focus.

All of the above constituent roles of the *network stakeholders* responsibility assignment can contribute knowledge to the operations of the IKN as *knowledge owners*.

Based on the preceding discussion, the constituent roles of the *network stakeholders* (Resp-1) responsibility assignment share the following work products (refer to Figure 8.26) (2):

• *Industry enterprises* collaborate by providing each other with operational resources and directives.	Resp-1.1→Resp-1.1
• *Industry enterprises* deliver products and services to the *market*.	Resp-1.1→Resp-1.4
• *Industry enterprises* provide *intermediaries* with updates on their innovative progress, as well as their share of revenue generated from innovative outputs.	Resp-1.1→Resp-1.2
• *Industry enterprises* provide *research institutions* with feedback on the application of their research as innovations, as well as their share of revenue generated from innovative outputs.	Resp-1.1→Resp-1.3
• *Intermediaries* provide networking opportunities to *industry enterprises*, *research institutions*, the *market* and *government*.	Resp-1.2→ Resp-1.1, Resp-1.3, Resp-1.4, Resp-1.5
• *Intermediaries* provide guidance to *industry enterprises* related to research outputs that are available for commercialisation	Resp-1.2→Resp-1.1

- *Intermediaries* provide guidance to *research institutions* related to issues and problems arising in industry that are in need of research to provide solutions. Resp-1.2→Resp-1.3
- *Research institutions* provide research outputs to *industry enterprises* and *intermediaries*. Resp-1.3→ Resp-1.1, Resp-1.2
- The *market* provides market insight and revenue to *industry enterprises*. Resp-1.4→Resp-1.1
- The *market* provides market insight to *intermediaries*. Resp-1.4→Resp-1.2
- *Government* provides regulations and incentives to *industry enterprise*, *intermediaries*, *research institutions* and the *market.* Resp-1.5→ Resp-1.1, Resp-1.2, Resp-1.3, Resp-1.4

Network stakeholders (Resp-1) and its constituent roles show **associations** (refer to Table 8.15) (D) with the following assignments in the responsibility abstraction (column):

- *Innovators* (Resp-2; refer to Figure 8.26 (3)):
  - *Industry enterprises* and *intermediaries* acting as *coordinators* through innovation project coordination. Resp-1.1, Resp-1.2 → Resp-2.4
  - *Industry enterprises* acting as *builders* through innovation project building. Resp-1.1→Resp-2.3
  - *Industry enterprises* acting as *leaders* by providing innovation project leadership. Resp-1.1→Resp-2.1
  - *Intermediaries* acting as *networkers* through networking opportunity promotion. Resp-1.2→Resp-2.2
  - *Research institutions* acting as *builders* through idea, concept and prototype building. Resp-1.2→Resp-2.3
  - *Industry enterprises* and the *market* acting as *anthropologists* by identifying innovation opportunities. Resp-1.1, Resp-1.4 → Resp-2.5
  - *Government* acting as a *leader* by providing national and regional innovation focus. Resp-1.5→Resp-2.1
- *Knowledge owners* (Resp-3; refer to Figure 8.26 (4)):
  - *Industry enterprises*, *intermediaries*, *research institutions*, the *market* and *government* that contribute knowledge as *knowledge owners.* Resp-1.1, Resp-1.2, Resp-1.3, Resp-1.4, Resp-1.5 → Resp-3

Network stakeholders (Resp-1) and their constituent roles furthermore show **integrations** (refer to Table 8.15) (E) with the following concepts in other abstractions (columns) of the reference architecture for IKNs:

- *Knowledge – network stakeholders* have access to *tacit knowledge* through their roles as *knowledge owners*. Resp-1 ↔ (Resp-3 →) Inv-1.3
- *Innovation artefacts – network stakeholders* construct *innovation artefacts* Resp-1 ↔ (Resp-2 →) Inv-2

through their roles as *innovators*.

- *Resources* – *network stakeholders* may contribute *information systems*, *workspaces*, *equipment* and/or *funding* to the IKN.
  Resp-1 ↔ Inv-3
- *Network building* – this process leads to alterations in *network stakeholders* of the IKN.
  Resp-1 ↔ Proc-1
- *Knowledge creation and transfer* – *network stakeholders* are involved in this process through their roles as *knowledge owners*.
  Resp-1 ↔ (Resp-3 →) Proc-2
- *Innovation* – *network stakeholders* are involved in this process through their roles as *innovators*.
  Resp-1 ↔ (Resp-2 →) Proc-3
- *Strategic network* – the geographical distribution of *network stakeholders* via their *network stakeholder locations* in the *strategic network* impacts various IKN operations.
  Resp-1 ↔ (Dist-2.1 →) Dist-1
- *Operational network* – *network stakeholders* are situated at *network stakeholder locations*.
  Resp-1 ↔ Dist-2.1
- *Network life cycle* – this timing cycle cues and synchronises the alteration of *network stakeholders* of the IKN with other timing cycles that are of significance to the network.
  Resp-1 ↔ Tim-2
- *Network building rules* – these motivations dictate the IKN's approach to management and governance of its *network stakeholders.*
  Resp-1 ↔ Mot-3.4

8.5.6.4.2 Innovators

This responsibility assignment definition model (C4, R2.2.2) defines *innovators* (Resp-2) as a responsibility assignment at an intermediate level of the detail. The textual version of this model is presented in Table 8.16, and the visual version in Figure 8.27.

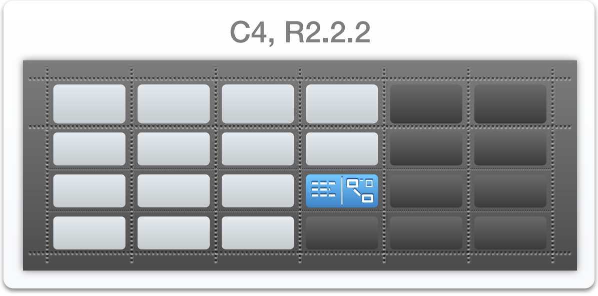


Table 8.16: Responsibility assignment definition model – Innovators (C4, R2.2.2)

	Code	**Name / definition**
A. Concept	**Resp-2**	**Innovators**
B. Definition		Role-players who are as an organisation or an individual responsible for various aspects of the innovation process within the network.
C. Deconstruction	Resp-2.1	Leaders

	Code	**Name / definition**
	Resp-2.2	Networkers
	Resp-2.3	Builders
	Resp-2.4	Coordinators
	Resp-2.5	Anthropologists
D. Associations	Resp-1	Network stakeholders
	Resp-3	Knowledge owners
E. Integrations	Inv-1.2	Innovation artefacts
	Proc-3	Innovation
	Tim-4	Innovation life cycle
	Mot-3.7	Innovation role rules

The *innovators* responsibility assignment is **defined** (refer to Table 8.16) (B) as:

> *Role-players who are as an organisation or an individual responsible for various aspects of the innovation process within the network.*

The *innovators* responsibility assignment is **deconstructed** (refer to Table 8.16 (C), Figure 8.27 (1)) into five constituent roles, namely *leaders*, *networkers*, *builders*, *coordinators* and *anthropologists*, via “is-a” relationships.

Resp-2 → Resp-2.1, Resp-2.2, Resp-2.3, Resp-2.4, Resp-2.5

Leaders (Resp-2.1) are:

> *Innovators that align innovative activities in the network with strategy and objectives, building and involving teams of the ‘right’ individuals and/or organisations at the ‘right’ time. Leaders evaluate and prioritise opportunities and ideas against a standard framework considering all business requirements. They guide progress, monitor metrics, instigate corrective action and attempt to build synergy into projects and the network at large.*

Leaders lead other *innovators,* especially *coordinators*, *networkers* and *builders*, to align their activities with the network’s innovation strategy and objectives.

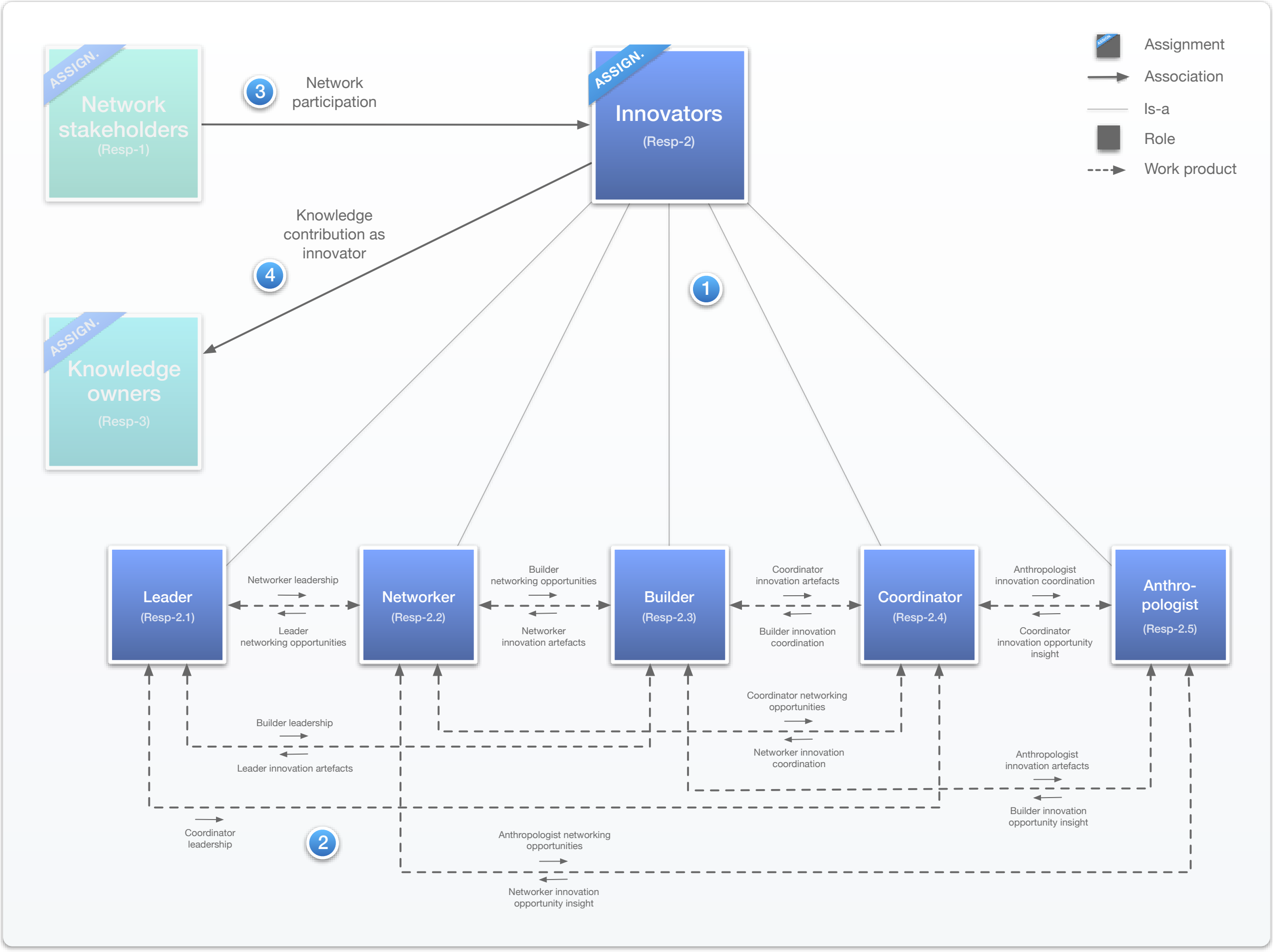


Figure 8.27: Responsibility assignment definition model – Innovators (C4, R2.2.2)

Networkers (Resp-2.2) are:

> *Innovators that scan market, industry, technology, regulatory and societal trends to understand potential futures and identify latent opportunities. Networkers create connections between individuals, teams and organisations internal and external to the network that have common or complementary objectives.*

Networkers provide *leaders*, *builders*, *coordinators* and *anthropologists* with networking opportunities that allow them to communicate and create opportunities for *knowledge creation and transfer* in the IKN.

Builders (Resp-2.3) are:

> *Innovators that make tangible concepts of ideas, demonstrate concepts, obtain feedback from colleagues and customers, and refine concepts. Builders build, test and refine working 'products' and ensure 'production' readiness. Builders strive towards the initial vision of the concept with minimal compromise for design, production and delivery.*

Builders provide *leaders*, *networkers*, *coordinators* and *anthropologists* with innovation artefacts that are relevant to their context and particular role as *innovators*.

Coordinators (Resp-2.4) are:

> *Innovators that balance project objectives, resources and risk against the broader network context. Coordinators contextualise, position and promote opportunities and concepts. They also prioritise, plan, coordinate, schedule, and assure completion of projects that are executed collaboratively in the network. Coordinators overcome or outsmart obstacles faced during the challenging nature of collaborative innovation projects.*

Coordinators coordinate the activities of other *innovators*, especially those of *builders*, *networkers* and *anthropologists*, in the networked context of an IKN to align them with the strategy and objectives provided by the *leader*. The *coordinator* role has increased importance in an inter-organisational innovation context, given the challenges involved in producing innovations in a collaborative fashion. The *coordinator,* for instance, plays an important role in both the *network building* and *innovation* processes, as well as throughout the *network* and *innovation life cycles,* to ensure that the IKN has access to the required capabilities that is required to innovate.

Anthropologists (Resp-2.5) are:

> *Innovators that develop an understanding of how people interact physically and emotionally with products, services, one another and their environment, thereby identifying opportunities for innovation. Anthropologists transform the physical environment into a tool to influence behaviour and attitude, enabling individuals to do their best work. They also anticipate and service the needs of colleagues in the network, customers, suppliers and other stakeholders.*

Anthropologists provide *coordinators*, *builders* and *networkers* with insight into innovation opportunities, given their insight into how people interact with products, services, one another and their environment.

Based on the preceding discussion, the constituent roles of the *innovators* (Resp-2) responsibility assignment share the following work products (refer to Figure 8.27) (2):

- *Leaders* provide leadership to *coordinators*, *networkers* and *builders*.

Resp-2.1 → Resp-2.2, Resp-2.3, Resp-2.4

- *Networkers* provide networking opportunities to *leaders*, *builders*, *coordinators* and *anthropologists*.

Resp-2.2 → Resp-2.1, Resp-2.3, Resp-2.4, Resp-2.5

- *Builders* provide innovation artefacts to *leaders*, *networkers*, *coordinators*

Resp-2.3 → Resp-2.1, Resp-2.2,

and *anthropologists*.

Resp-2.4, Resp-2.5

- *Coordinators* provide innovation coordination to *networkers*, *builders* and *anthropologists*.

Resp-2.4 → Resp-2.2, Resp-2.3, Resp-2.5

- *Anthropologists* provide insight into innovation opportunities to *networkers*, *builders* and *coordinators*.

Resp-2.5 → Resp-2.2, Resp-2.3, Resp-2.4

Innovators (Resp-2) and its constituent roles show **associations** (refer to Table 8.16) (D) with the following assignments in the responsibility abstraction (column):

- *Network stakeholders* (Resp-1; refer to Figure 8.27 (3)):
  - *Network stakeholders* participate in the activities of an IKN by being assigned one or more responsibilities as an *innovator* (for more detail refer to section 8.5.6.4.1 on *network stakeholders*).

Resp-1 → Resp-2

- *Knowledge owners* (Resp-3; refer to Figure 8.27 (4)):
  - Organisations and/or individuals that are involved in the IKN contribute knowledge as *knowledge owners* in their capacity as *innovators*.

Resp-2 → Resp-3

Innovators (Resp-2) and its constituent roles furthermore show **integrations** (refer to Table 8.16) (E) with the following concepts in other abstractions (columns) of the reference architecture for IKNs:

- *Innovation artefacts* – *innovators* produce *innovation artefacts*.

Resp-2 ↔ Inv-2

- *Innovation* – *innovators* execute the *innovation* process flow, thereby producing the *innovation artefacts* indicated above.

Resp-2 ↔ Proc-3

- *Innovation life cycle* – this timing cycle cues and synchronises the activities of *innovators* in the *innovation* process itself, but also with other timing cycles that are of significance to the IKN.

Resp-2 ↔ (Proc-3 →) Tim-4

- *Innovation role rules* – this motivational end governs the actions of *innovators*.

Resp-2 ↔ Mot-3.7

8.5.6.4.3 Knowledge owners

This responsibility assignment definition model (C4, R2.2.3) defines *knowledge owners* (Resp-3) as a responsibility assignment at an intermediate level of the detail. The textual version of this model is presented in Table 8.17, and the visual version in Figure 8.28.

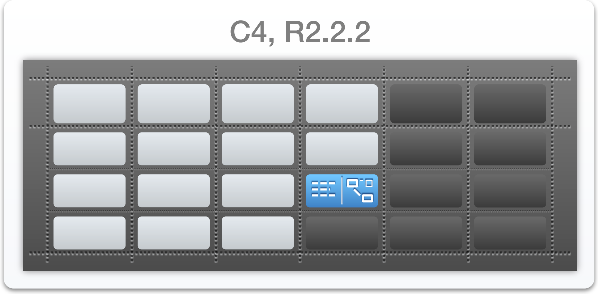


Table 8.17: Responsibility assignment definition model – Knowledge owners (C4, R2.2.3)

	Code	Name / definition
A. Concept	**Resp-3**	**Knowledge owners**
B. Definition		Individuals that possess tacit knowledge.
C. Associations	Resp-1	Network stakeholders
	Resp-2	Innovators
D. Integrations	Inv-1.3	Tacit knowledge
	Inv-2	Innovation artefacts
	Proc-2	Knowledge creation and transfer
	Tim-3	Knowledge life cycle
	Mot-3	Innovation intention

The *knowledge owners* responsibility assignment is **defined** (refer to Table 8.17) (B) as:

> *Individuals (and by implication organisations) that possess tacit knowledge.*

Knowledge owners are intrinsically linked to *tacit knowledge.* Having *network stakeholders* that contribute *tacit knowledge*, and subsequently also *explicit knowledge*, to the network's operations is part of the mandate and purpose of an IKN. These *knowledge owners* play a particularly important role in the *knowledge creation and transfer* process and the production of *innovation artefacts*. This is reflected in the *innovation strategy* of an IKN.

Knowledge owners (Resp-3) and its constituent roles show **associations** (refer to Table 8.17) (C) with the following assignments in the responsibility abstraction (column):

- *Network stakeholders* (Resp-1; refer to Figure 8.28 (1)):
    - Organisations and/or individuals that are involved in the IKN contribute knowledge as *knowledge owners* in their capacity as *network stakeholders*.

Resp-1 → Resp-3

- *Innovators* (Resp-3; refer to Figure 8.28 (2)):
    - Organisations and/or individuals that are involved in the IKN contribute knowledge as *knowledge owners* in their capacity as *innovators*.

Resp-2 → Resp-3

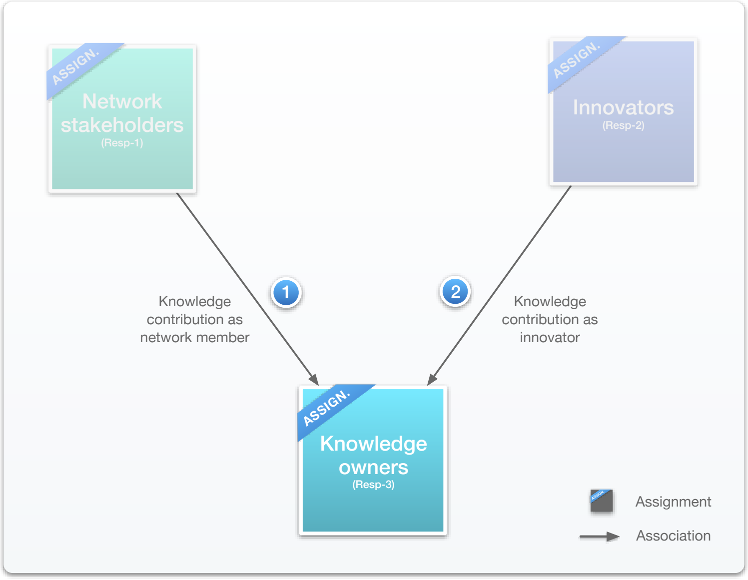
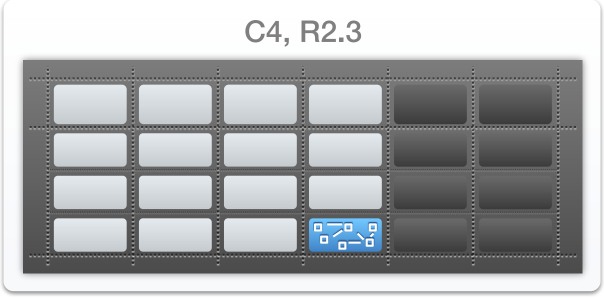


Figure 8.28: Responsibility assignment definition model – Knowledge owners (C4, R2.2.3)

Knowledge owners (Resp-2) and its constituent roles furthermore show **integrations** (refer to Table 8.17) (D) with the following concepts in other abstractions (columns) of the reference architecture for IKNs:

- *Tacit knowledge* – *knowledge owners* are carriers of *tacit knowledge.*
  Resp-3 ↔ Inv-1.3
- *Innovation artefacts* – *knowledge owners* make valuable contributions to the production of *innovation artefacts* through their possession of *tacit knowledge*.
  Resp-3 ↔ (Inv-1.3 →) Inv-2
- *Knowledge creation and transfer* – *knowledge owners* make valuable contributions to the *knowledge creation and transfer* process through their possession of *tacit knowledge*.
  Resp-3 ↔ (Inv-1.3 →) Proc-2
- *Knowledge life cycle* – this timing cycle cues and synchronises the activities of *knowledge owners* in the *knowledge creation and transfer* process itself, but also with other timing cycles that are of significance to the IKN.
  Resp-3 ↔ (Proc-2 →) Tim-3
- *Innovation intention* – this motivation determines the influence and management of *knowledge owners* in the IKN.
  Resp-3 ↔ Mot-3

8.5.6.5 Extended responsibility definition model



The extended responsibility definition model (C4, R2.3) defines the responsibility assignments identified in the responsibility identification model (C4, R1; refer to Table 8.14) at a high level of detail by representing them visually according to the abstraction meta-model (refer to Figure 8.24). This model is based on the interpretation of the *definition data* from the qualitative systematic review described in section 8.5.2.1, with regard to the "*Who?*" review question.

The various constituent roles are indicated and organised into swim lanes according to their overarching high-level responsibility assignments (refer to Figure 8.29). The detailed work products between them are also labelled appropriately. This model therefore retains all the defining deconstructions and associations from the responsibility assignment definition models in the preceding sections.

No deconstruction of roles into sub-roles is done for the extended responsibility definition model. To ensure that the model remains a primitive (single-variable) model, as well as for readability reasons, integrations with concepts in other abstractions (columns) are not indicated in the model.

Figure 8.29 should be noted in particular as it shows that *network stakeholders* can have more than one *innovator* role, and that more than one *network member* can have the same *innovator* role.

Other emergent effects commonly found in collaborative networked organisations and IKNs in particular can also be understood given the model presented in Figure 8.29. One example of this is the concept of "hub firms" that play a central and dominating role in networks. These organisations play a leading role in the strategic direction of the network, and may also provide the bulk of the connections and resources.

In such cases, a particular *industry enterprise* may also have the role of *intermediary*, and therefore act as *leader*, *networker*, *builder* and *coordinator* at once. This architectural configuration would explain the behaviour of such "hub firms" in IKNs.

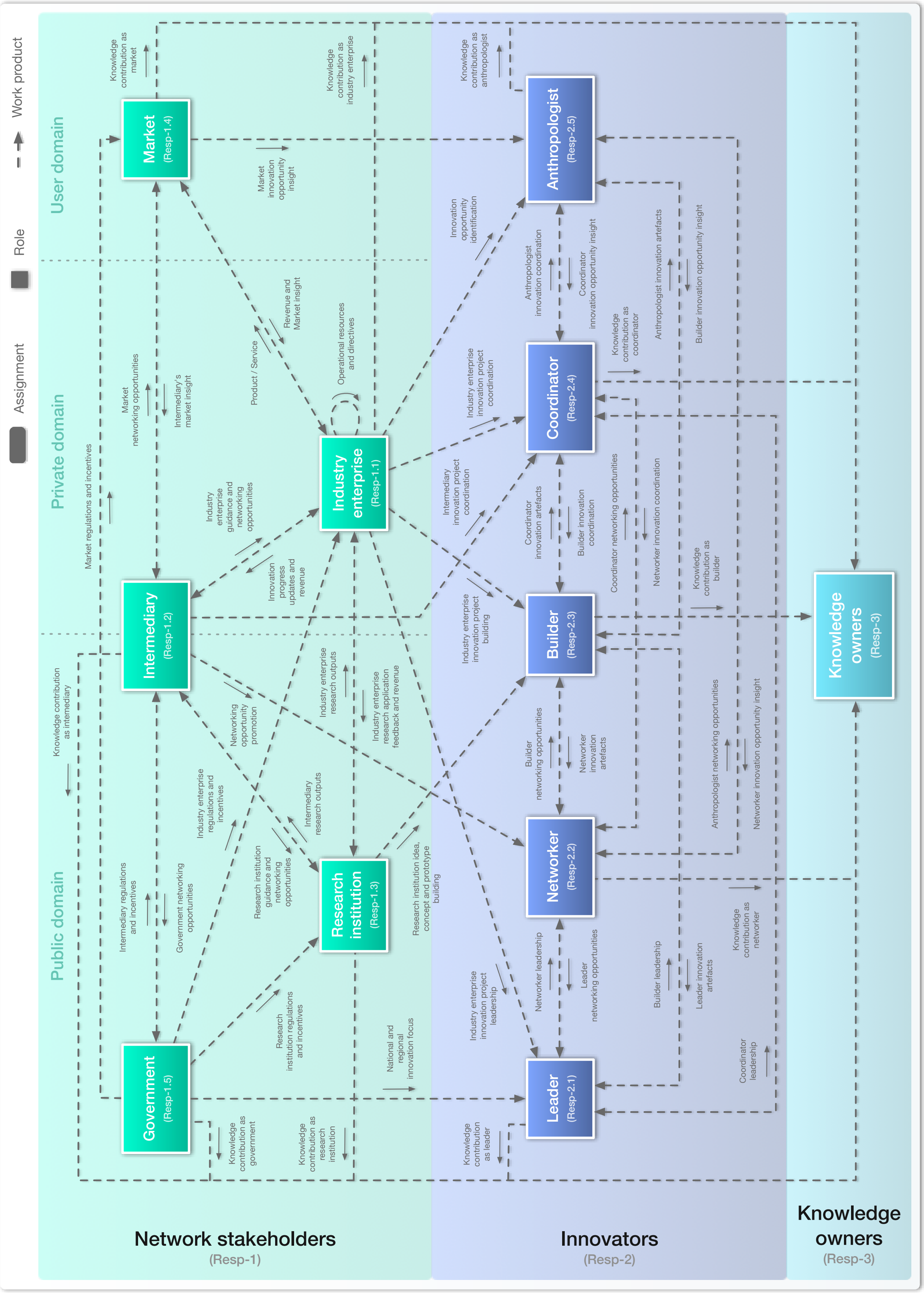
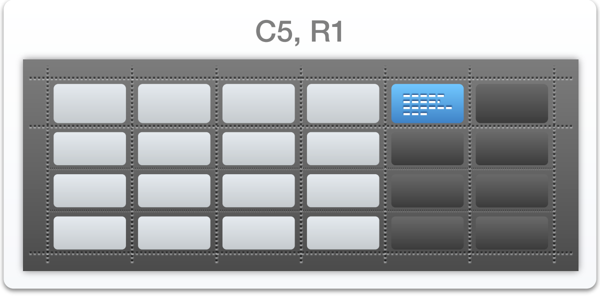


Figure 8.29: Extended responsibility definition model

8.5.7 Timing cycle models

This section discusses the timing abstraction (fifth column) of the artefact, and the models presented therefore answer the interrogative of "when" for the engineering of IKNs. The top-row reference model is presented and motivated first. Thereafter the meta-model for the second row of the fifth column is presented, followed by the various second-row reference models according to increasing levels of detail. Where the description of a model includes a reference to another formalised concept in the reference architecture, the name of the concept is printed in italics.

8.5.7.1 Timing identification model



The timing identification model (C5, R1; refer to Table 8.18) lists timing types that are significant and common to the engineering of IKNs. This model is based on the interpretation of the *identification data* from the qualitative systematic review described in section 8.5.2.1, with regard to the "*When?*" review question.

At the lowest level of detail, four main timing types are identified, being *technology life cycle*, *network life cycle*, *knowledge life cycle* and *innovation life cycle*.

Table 8.18: Timing identification model (C5, R1)

Code	Timing type	Code	Constituent timing sub-types
Tim-1	**Technology life cycle**	Tim-1.1	Technology identification timing
		Tim-1.2	Technology solution architecture timing
		Tim-1.3	Technology development or acquisition timing
		Tim-1.4	Technology implementation timing
		Tim-1.5	Technology exploitation timing
		Tim-1.6	Technology decommissioning timing
		Tim-1.7	Technology life cycle conclusion timing
Tim-2	**Network life cycle**	Tim-2.1	Network design timing
		Tim-2.2	Network implementation timing
		Tim-2.3	Network operation timing

Code	Timing type	Code	Constituent timing sub-types
		Tim-2.4	Network refinement timing
		Tim-2.5	Network phase-out timing
		Tim-2.6	Network life cycle conclusion timing
Tim-3	**Knowledge life cycle**	Tim-3.1	Knowledge identification and extraction timing
		Tim-3.2	Knowledge structuring and formalisation timing
		Tim-3.3	Knowledge refinement and development timing
		Tim-3.4	Knowledge dissemination timing
		Tim-3.5	Knowledge maintenance timing
		Tim-3.6	Knowledge life cycle conclusion timing
Tim-4	**Innovation life cycle**	Tim-4.1	Innovation trigger timing
		Tim-4.2	Idea filtering timing
		Tim-4.3	Concept filtering timing
		Tim-4.4	Project funding timing
		Tim-4.5	Project launch timing
		Tim-4.6	Project implementation timing
		Tim-4.7	Innovation exploitation timing
		Tim-4.8	Innovation life cycle conclusion timing

The *technology life cycle* is identified as a key timing type that influences timing in IKNs, as it directs the various waves of development and innovation that IKNs are aimed at. The *technology life cycle* timing type is deconstructed into timing subtypes of *technology identification timing*, *technology solution architecture timing*, *technology development or acquisition timing*, *technology implementation timing*, *technology exploitation timing*, *technology decommissioning timing* and *technology life cycle conclusion timing*.

The *network life cycle* is also identified as fundamental timing type in IKNs, which directs the development of the network itself, as well as the timing of its key processes. The *network life cycle* timing type is deconstructed into timing subtypes of *network design timing*, *network implementation timing*, *network operation timing*, *network refinement timing*, *network phase-out timing* and *network life cycle conclusion timing*.

The *knowledge life cycle* is a next key timing type in IKNs, as it directs the development and sharing of knowledge that is created in the network. The *knowledge life cycle* timing type is deconstructed into timing subtypes of *knowledge identification and extraction timing*, *knowledge structuring and formalisation timing*, *knowledge refinement and development timing*, *knowledge dissemination timing*, *knowledge maintenance timing* and *knowledge life cycle conclusion timing*.

The *innovation life cycle* is identified as another key timing type in IKNs, as it directs the innovation process which is fundamental to the purpose of an IKN. The *innovation life cycle* timing type is deconstructed into timing subtypes of *innovation trigger timing*, *idea filtering timing*, *concept filtering timing*, *project funding timing*, *project launch timing*, *project implementation timing*, *innovation exploitation timing* and *innovation life cycle conclusion timing*.

8.5.7.2 Meta-model for definition models in timing abstraction

The primitive meta-model presented in Figure 8.30 serves as the semi-formal representational language for the various second-row definition models in the timing abstraction, or '*when*' column, of the reference architecture for IKNs.

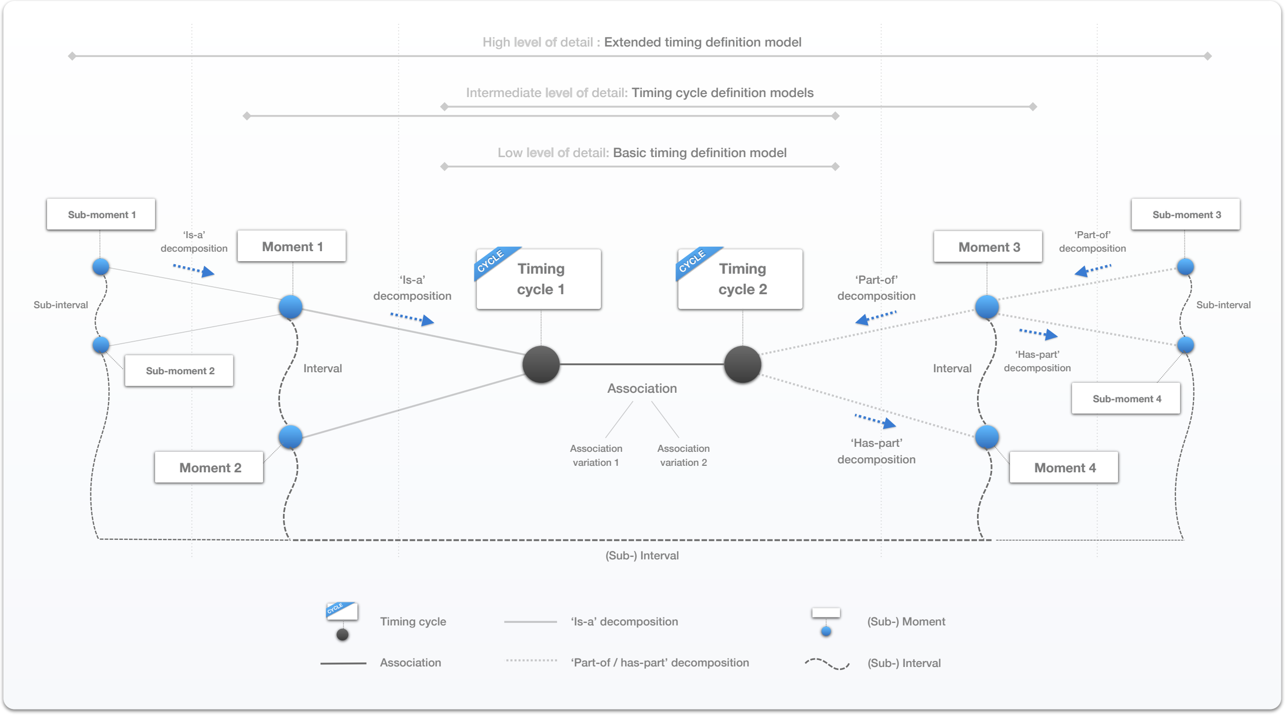
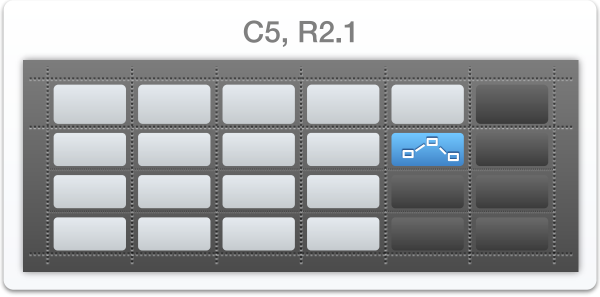


Figure 8.30: Meta-model for definition models in timing abstraction

This meta-model assists in defining *timing cycles* as concepts and follows the single-variable structure of "*moment-interval-moment*" amongst its elements, links, sub-elements and sub-links. The meta-model therefore deconstructs and represents timing cycles into moments and sub-moments, along with their respective intervals and sub-intervals.

This representation is done at various levels of detail in the basic timing definition model (refer to section 8.5.7.3), timing cycle definition models (refer to section 8.5.7.4) and extended timing definition model (refer to section 8.5.7.5). For all of these models, the meta-model scope indication shown at the top of Figure 8.30 applies.

8.5.7.3 Basic timing definition model



The basic timing definition model (C5, R2.1) defines the timing cycles that were identified as timing types in the timing identification model (C5, R1; refer to Table 8.18) at a low level of detail by representing them visually according to the abstraction meta-model (refer to Figure 8.30). This model is based on the interpretation of the *definition data* from the qualitative systematic review described in section 8.5.2.1, with regard to the “*When?*” review question.

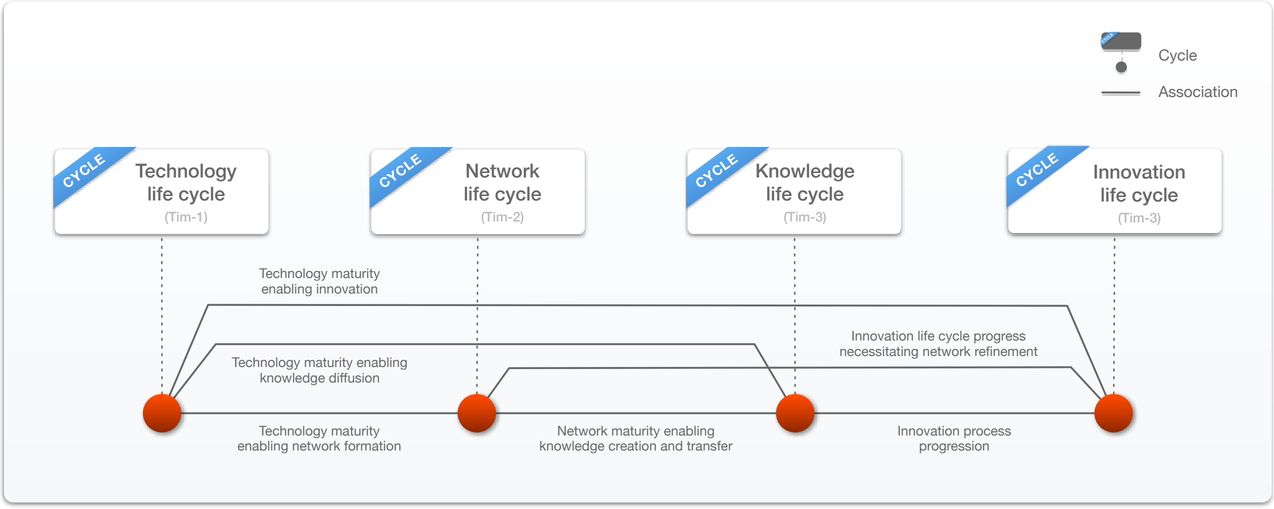


Figure 8.31: Basic timing definition model (C5, R2.1)

The model depicts the various identified timing cycles, as well as the associations between them (refer to Figure 8.31). The model shows that the *technology life cycle* associates with the *network life cycle* through technology maturity levels that enable the formation of networks with different missions. These networks could be based on immature technology that needs to be developed further, or on mature technology that could be applied to new environments. The *technology life cycle* also associates with the *knowledge life cycle* through technology maturity enabling knowledge development and diffusion regarding the technology itself. The *technology life cycle* furthermore associates with the *innovation life cycle* through technology maturity levels that enable innovation in terms of new and immature technologies, or innovative application of mature technologies. The *network life cycle* associates with the *knowledge life cycle* through network maturity that enables knowledge development and diffusion in the given domain of network formation. The *network life cycle* also associates with the *innovation life cycle* through network maturity that enables the network to pursue its innovative purpose. The *knowledge life cycle* associates with the *innovation life cycle* through synchronisation between the two life cycles as the innovation process progresses, and as both life cycles are initiating new instances of the other.

These timing cycles are now defined in more detail in the relevant timing cycle definition models in section 8.5.7.4.

8.5.7.4 Timing cycle definition models

The timing cycle definition models presented in this section each define an identified timing cycle at an intermediate level of detail. The models therefore add new information for each cycle, while retaining all the information from the basic timing definition model (refer Figure 8.31). Both textual and visual versions are provided to assist in defining the cycles. Deconstructions, relationships and integrations between concepts are expressed in terms of identification codes in side-notes for traceability purposes.

The models presented in this section are based on the interpretation of the *definition data* for each identified timing cycle, as obtained from the qualitative systematic review described in section 8.5.2.1.

8.5.7.4.1 Technology life cycle

This timing cycle definition model (C5, R2.2.1) defines the *technology life cycle* (Tim-1) as a timing cycle at an intermediate level of the detail. The textual version of this model is initially presented in Table 8.19 and the visual version in Figure 8.32.

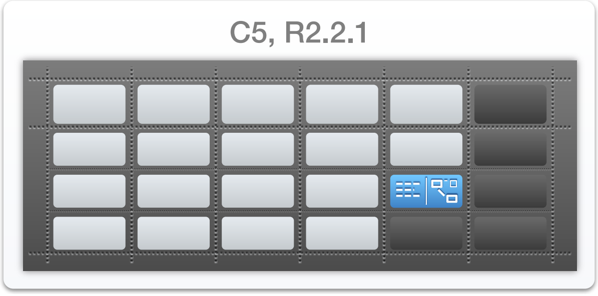


Table 8.19: Timing cycle definition model – Technology life cycle (C5, R2.2.1)

	Code	Name / definition
A. Concept	**Tim-1**	**Technology life cycle**
B. Definition		The series of stages through which a certain technology passes during its inception, development, application and phase-out.
C. Deconstruction	Tim-1.1	Technology identification moment
	Tim-1.2	Technology solution architecture moment
	Tim-1.3	Technology development or acquisition moment

	Code	Name / definition
	Tim-1.4	Technology implementation moment
	Tim-1.5	Technology exploitation moment
	Tim-1.6	Technology decommissioning moment
	Tim-1.7	Technology life cycle conclusion moment
D. Associations	Tim-2	Network life cycle
	Tim-3	Knowledge life cycle
	Tim-4	Innovation life cycle
E. Integrations	Inv-1	Knowledge
	Inv-2	Innovation artefacts
	Inv-3	Resources
	Proc-1	Network building
	Proc-2	Knowledge creation and transfer
	Mot-3.3	Life cycle rules

The *technology life cycle* (Tim-1) timing cycle is **defined** (refer to Table 8.19) (B) as:

> *The series of stages through which a certain technology passes during its inception, development, application and phase-out.*

The *technology life cycle* timing cycle is **deconstructed** (refer to Table 8.19 (C), Figure 8.32 (1)) into seven constituent moments, namely *technology identification moment*, *technology solution architecture moment*, *technology development or acquisition moment*, *technology implementation moment*, *technology exploitation moment*, *technology decommissioning moment* and *technology life cycle conclusion moments*, via “part-of/has-part” relationships.

Tim-1 →
Tim-1.1, Tim-1.2,
Tim-1.3, Tim-1.4,
Tim-1.5, Tim-1.6,
Tim-1.7

The *technology identification moment* (Tim-1.1) is:

> *The moment when the identification of a technology is initiated.*

This moment is followed by a period where the technology identification is performed, and which is concluded with a *technology solution architecture moment*.

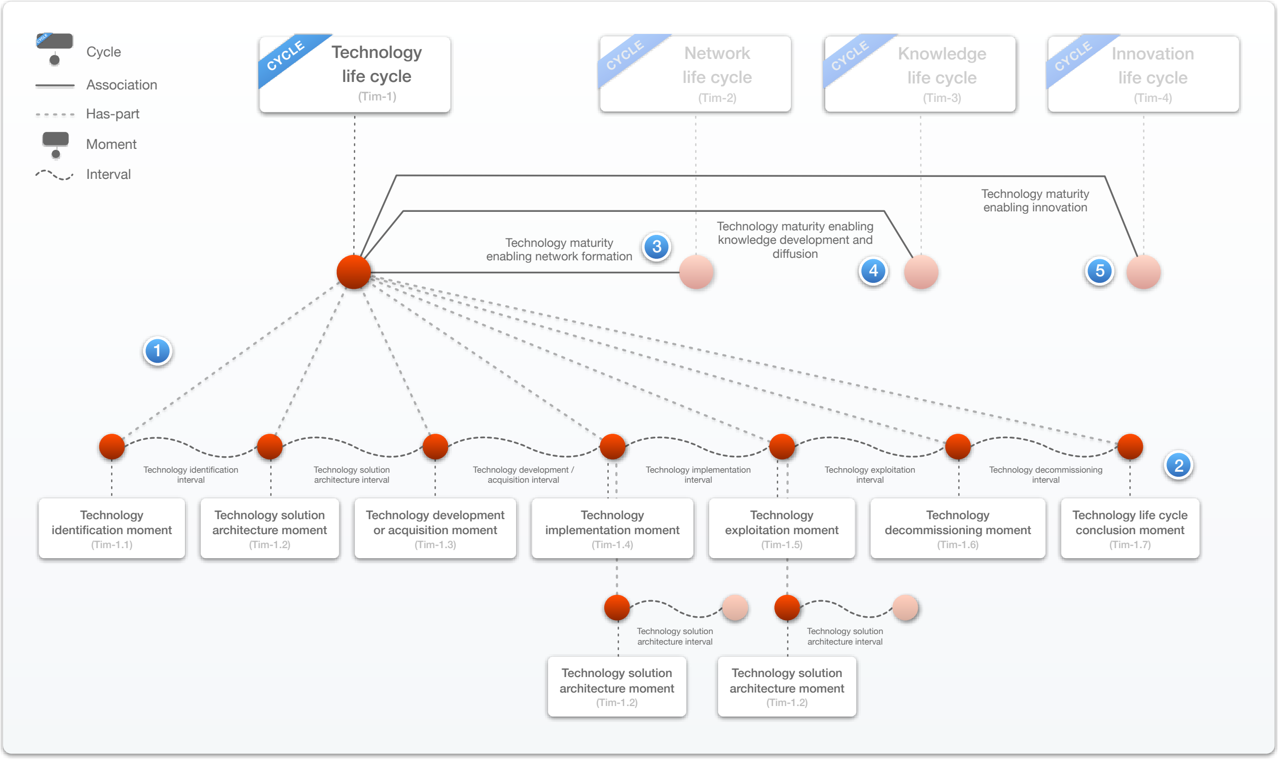


Figure 8.32: Timing cycle definition model – Technology life cycle (C5, R2.2.1)

The *technology solution architecture moment* (Tim-1.2) is:

> *The moment when the packaging of a technology as a feasible solution to a real-world problem is initiated.*

This moment is followed by a period during which the packaging of the technology as a solution architecture is performed, and which is concluded with a *technology development or acquisition moment*.

The *technology development or acquisition moment* (Tim-1.3) is:

> *The moment when the development or acquisition of the identified and packaged technology is initiated.*

This moment is followed by a period during which the technology development or acquisition is performed. This period may be concluded with a *technology implementation moment*, or a feedback loop may be initiated where the technology is repackaged as a solution architecture due to a newly identified need. This repackaging is initiated by a *technology solution architecture moment* as discussed above.

The *technology implementation moment* (Tim-1.4) is:

> *The moment when the implementation of the developed or acquired technology is initiated.*

This moment is followed by a period where the technology implementation is performed. This period may be concluded with a *technology exploitation moment*, or another feedback loop may be initiated where the technology is repackaged as a solution architecture due to another newly identified need. This repackaging is initiated by a *technology solution architecture moment* as discussed above.

The *technology exploitation moment* (Tim-1.5) is:

> *The moment when the exploitation of the implemented technology is initiated.*

This moment is followed by a period where the technology exploitation is performed, and which is concluded with a *technology decommissioning moment*.

The *technology decommissioning moment* (Tim-1.6) is:

> *The moment when the decommissioning of an exploited technology is initiated.*

This moment is followed by a period during which the technology decommissioning is performed, and which is concluded with a *technology life cycle conclusion moment*.

The *technology life cycle conclusion moment* (Tim-1.7) is:

> *The moment when the decommissioning of a technology is complete.*

This moment signals the conclusion of the *technology life cycle*.

Based on the preceding discussion, the constituent moments of the *technology life cycle* (Tim-1) timing cycle share the following intervals (refer to Figure 8.32) (2):

- A technology identification interval between the *technology identification moment* and the *technology solution architecture moment*. Tim-1.1 → Tim-1.2
- A technology solution architecture interval between the *technology solution architecture moment* and the *technology development or acquisition moment*. Tim-1.2 → Tim-1.3
- A technology development or acquisition interval between the *technology development or acquisition moment* and the *technology implementation moment*. Tim-1.3 → Tim-1.4
- A technology implementation interval between the *technology implementation moment* and the *technology exploitation moment*. Tim-1.4 → Tim-1.5
- A technology exploitation interval between the *technology exploitation moment* and the *technology decommissioning moment*. Tim-1.5 → Tim-1.6

- A technology decommissioning interval between the *technology decommissioning moment* and the *technology life cycle conclusion moment*.

Tim-1.6 → Tim-1.7

The *technology life cycle* (Tim-1) and its constituent moments show **associations** (refer to Table 8.19) (D) with the following cycles in the timing abstraction (column):

- *Network life cycle* (Tim-2; refer to Figure 8.32 (3)):
  - The *technology life cycle* associates with the *network life cycle* through technology maturity levels that enable the formation of networks with different missions

Tim-1 → Tim-2

- *Knowledge life cycle* (Tim-3; refer to Figure 8.32 (4)):
  - The *technology life cycle* associates with the *knowledge life cycle* through technology maturity enabling knowledge development and diffusion regarding the technology itself.

Tim-1 → Tim-3

- *Innovation life cycle* (Tim-4; refer to Figure 8.32 (5)):
  - The *technology life cycle* associates with the *innovation life cycle* through technology maturity levels that enable innovation in terms of new and immature technologies, or innovative application of mature technologies.

Tim-1 → Tim-4

The *technology life cycle* (Tim-1) and its constituent moments furthermore show **integrations** (refer to Table 8.19) (E) with the following concepts in other abstractions (columns) of the reference architecture for IKNs:

- *Knowledge* – the cycle integrates with *knowledge* through its association with the *knowledge life cycle*.

Tim-1 ↔
(Tim-3 →) Inv-1

- *Innovation artefacts* – the cycle enables or instigates the production of *innovation artefacts*.

Tim-1 ↔ Inv-2

- *Resources* – the cycle dictates when certain *resources* will be available for the production and transformation of *innovation artefacts*.

Tim-1 ↔ Inv-3

- *Network building* – the cycle influences the process of building the network through its association with the *network life cycle*.

Tim-1 ↔
(Tim-2 →) Proc-1

- *Knowledge creation and transfer* – the cycle integrates with *knowledge creation and transfer* through its association with the *knowledge life cycle*.

Tim-1 ↔
(Tim-3 →) Proc-2

- *Life cycle rules* – this motivational end determines the hierarchy, sequence, interaction and progression of life cycles in the IKN.

Tim-1 ↔ Mot-3.3

8.5.7.4.2 Network life cycle

This timing cycle definition model (C5, R2.2.2) defines the *network life cycle* (Tim-2) as a timing cycle at an intermediate level of the detail. The textual version of this model is initially presented in Table 8.20 and the visual version in Figure 8.33.

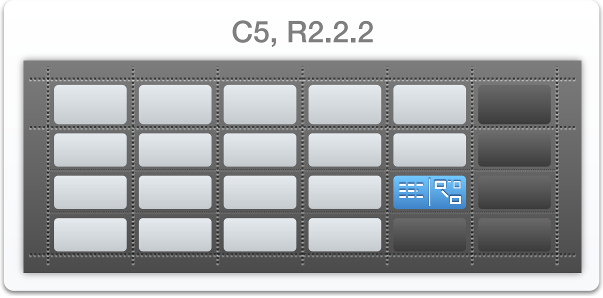


Table 8.20: Timing cycle definition model – Network life cycle (C5, R2.2.2)

	Code	**Name / definition**
A. Concept	**Tim-2**	**Network life cycle**
B. Definition		The series of stages through which an integrated knowledge network passes during its design, development, operation and phase-out.
C. Deconstruction	Tim-2.1	Network design moment
	Tim-2.2	Network implementation moment
	Tim-2.3	Network operation moment
	Tim-2.4	Network refinement moment
	Tim-2.5	Network phase-out moment
	Tim-2.6	Network life cycle conclusion moment
D. Associations	Tim-1	Technology life cycle
	Tim-3	Knowledge life cycle
	Tim-4	Innovation life cycle
E. Integrations	Proc-1	Network building
	Resp-1	Network stakeholders
	Mot-3.3	Life cycle rules

The *network life cycle* (Tim-2) timing cycle is **defined** (refer to Table 8.20) (B) as:

> *The series of stages through which an integrated knowledge network passes during its design, development, operation and phase-out.*

The *network life cycle* timing cycle is **deconstructed** (refer to Table 8.20 (C), Figure 8.33 (1)) into six constituent moments, namely *network design moment*, *network implementation moment*, *network operation moment*, *network refinement moment*, *network phase-out moment* and *network life cycle conclusion moment*, via “part-of/has-part” relationships.

Tim-2 →
Tim-2.1, Tim-2.2,
Tim-2.3, Tim-2.4,
Tim-2.5, Tim-2.6

The *network design moment* (Tim-2.1) is:

> *The moment when design of the network is initiated.*

This moment is followed by a period during which network design is performed, and which is concluded with a *network implementation moment*.

The *network implementation moment* (Tim-2.2) is:

> *The moment when network implementation is initiated.*

This moment is followed by a period where network implementation is performed, and which is concluded with a *network operation moment*.

The *network operation moment* (Tim-2.3) is:

> *The moment when network operation is initiated.*

This moment is followed by a period during which network operation is performed. This period may be concluded with a *network phase-out moment*, or a feedback loop may be initiated where the network is refined. This refinement is initiated by a *network refinement moment*.

The *network refinement moment* (Tim-2.4) is:

> *The moment when network refinement is initiated.*

This moment is followed by a period during which network refinement is performed, which is concluded with a *network operation moment*.

The *network phase-out moment* (Tim-2.5) is:

> *The moment when network phase-out is initiated.*

This moment is followed by a period during which network phase-out is performed, which is concluded with a *network life cycle conclusion moment*.

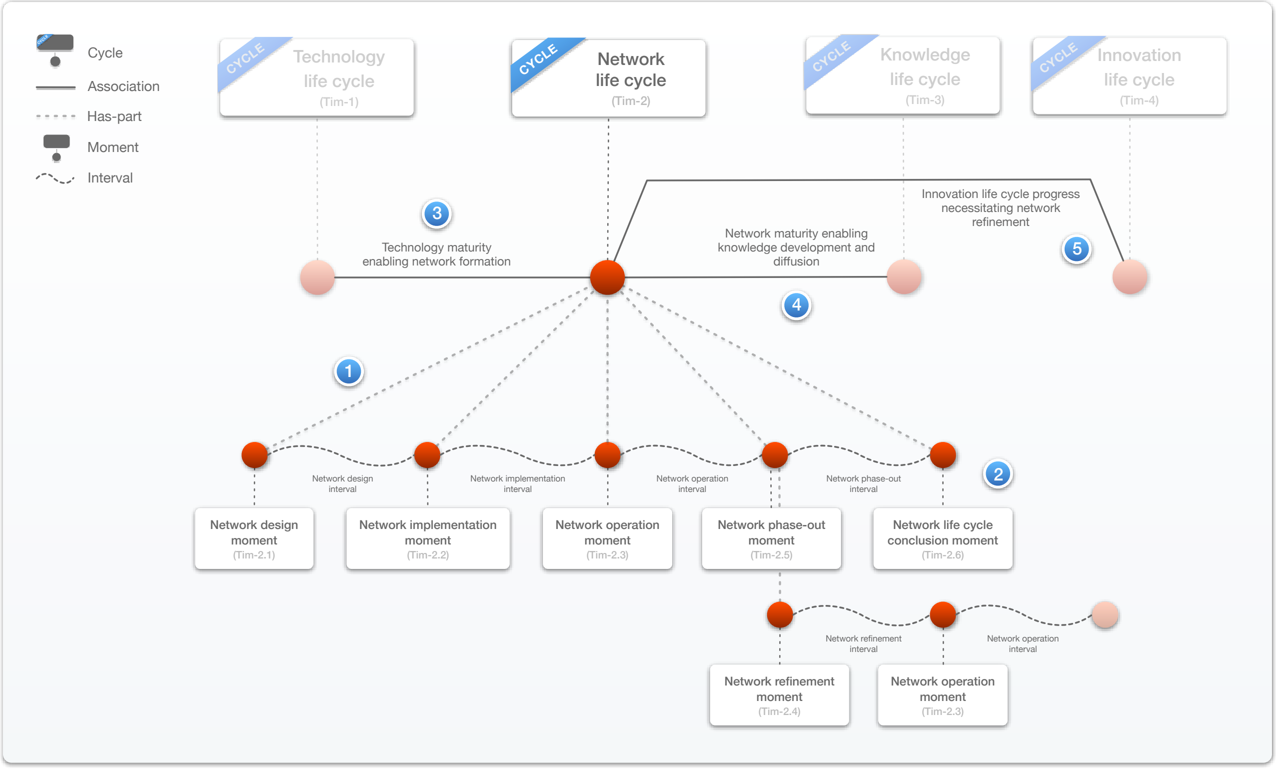


Figure 8.33: Timing cycle definition model – Network life cycle (C5, R2.2.2)

The *network life cycle conclusion moment* (Tim-2.6) is:

> *The moment when network phase-out is completed.*

This moment signals the conclusion of the *network life cycle*.

Based on the preceding discussion, the constituent moments of the *network life cycle* (Tim-2) timing cycle share the following intervals (refer to Figure 8.33) (2):

- A network design interval between the *network design moment* and the *network implementation moment*.
  Tim-2.1 → Tim-2.2
- A network implementation interval between the *network implementation moment* and the *network operation moment*.
  Tim-2.2 → Tim-2.3
- A network operation interval between the *network operation moment* and the *network phase-out moment* or *network refinement moment*.
  Tim-2.3 → Tim-2.4, Tim-2.5
- A network refinement interval between the *network refinement moment* and the *network operation moment*.
  Tim-2.4 → Tim-2.3
- A network phase-out interval between the *network refinement moment* and the *network life cycle conclusion moment*.
  Tim-2.5 → Tim-2.6

The *network life cycle* (Tim-2) and its constituent moments show **associations** (refer to Table 8.20) (D) with the following cycles in the timing abstraction (column):

- *Technology life cycle* (Tim-1; refer to Figure 8.33 (3)):
    - The *network life cycle* associates with the *technology life cycle* through technology maturity levels that enable the formation of networks with different missions.

Tim-2 → Tim-1

- *Knowledge life cycle* (Tim-3; refer to Figure 8.33 (4)):
    - The *network life cycle* associates with the *knowledge life cycle* through network maturity that enables knowledge development and diffusion in the given domain of network formation.

Tim-2 → Tim-3

- *Innovation life cycle* (Tim-4; refer to Figure 8.33 (5)):
    - The *network life cycle* associates with the *innovation life cycle* through network maturity that enables the network to pursue its innovative purpose.

Tim-2 → Tim-4

The *network life cycle* (Tim-2) and its constituent moments furthermore show **integrations** (refer to Table 8.20) (E) with the following concepts in other abstractions (columns) of the reference architecture for IKNs:

- *Network building* – the cycle shares deep ties with this process, as it provides the timing for the execution of the process.

Tim-2 ↔ Proc-1

- *Network stakeholders* – the cycle determines the timing of the involvement of new *network stakeholders*.

Tim-2 ↔ Resp-1

- *Life cycle rules* – this motivational end determines the hierarchy, sequence, interaction and progression of life cycles in the IKN.

Tim-2 ↔ Mot-3.3

8.5.7.4.3 Knowledge life cycle

This timing cycle definition model (C5, R2.2.3) defines the *knowledge life cycle* (Tim-3) as a timing cycle at an intermediate level of the detail. The textual version of this model is initially presented in Table 8.21 and the visual version in Figure 8.34.

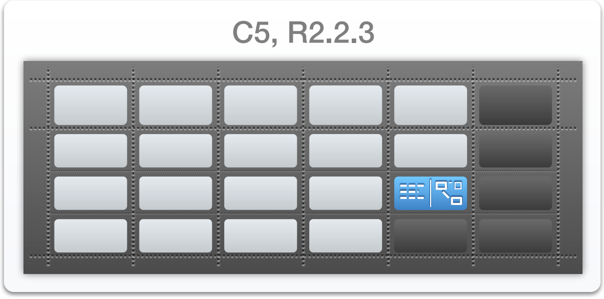


Table 8.21: Timing cycle definition model – Knowledge life cycle (C5, R2.2.3)

	Code	Name / definition
A. Concept	**Tim-3**	**Knowledge life cycle**
B. Definition		The series of stages through which knowledge is captured, refined, disseminated and maintained in an integrated knowledge network.
C. Deconstruction	Tim-3.1	Knowledge identification and extraction moment
	Tim-3.2	Knowledge structuring and formalisation moment
	Tim-3.3	Knowledge refinement and development moment
	Tim-3.4	Knowledge dissemination moment
	Tim-3.5	Knowledge maintenance moment
	Tim-3.6	Knowledge life cycle conclusion moment
D. Associations	Tim-1	Technology life cycle
	Tim-2	Network life cycle
	Tim-4	Innovation life cycle
E. Integrations	Inv-1	Knowledge
	Proc-2	Knowledge creation and transfer
	Resp-3	Knowledge owners
	Mot-3.3	Life cycle rules
	Mot-3.5	Knowledge creation and transfer rules

The *knowledge life cycle* (Tim-3) timing cycle is **defined** (refer to Table 8.21) (B) as:

> *The series of stages through which knowledge is captured, refined, disseminated and maintained in an integrated knowledge network.*

The *knowledge life cycle* timing cycle is **deconstructed** (refer to Table 8.21 (C), Figure 8.34 (1)) into six constituent moments, namely *knowledge identification and extraction moment*, *knowledge structuring and formalisation moment*, *knowledge refinement and development moment*, *knowledge dissemination moment*, *knowledge maintenance moment* and *knowledge life cycle conclusion moment*, via "part-of/has-part" relationships.

Tim-3 →
Tim-3.1, Tim-3.2,
Tim-3.3, Tim-3.4,
Tim-3.5, Tim-3.6

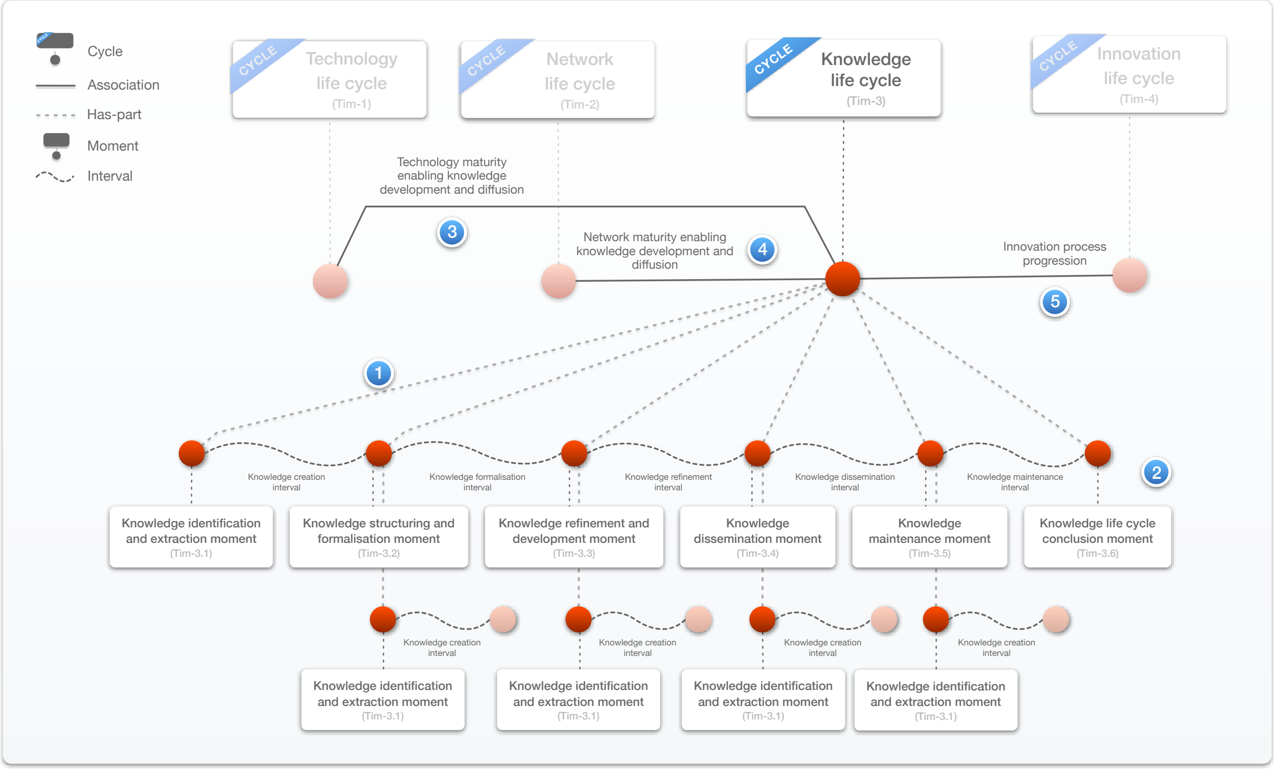


Figure 8.34: Timing cycle definition model - Knowledge life cycle (C5, R2.2.3)

The *knowledge identification and extraction moment* (Tim-3.1) is:

> *The moment when knowledge identification and extraction is initiated*.

This moment is followed by a period during which knowledge identification and extraction is performed, which is concluded with a *knowledge structuring and formalisation moment*.

The *knowledge structuring and formalisation moment* (Tim-3.2) is:

> *The moment when knowledge structuring and formalisation is initiated.*

This moment is followed by a period during which knowledge structuring and formalisation is performed, which is concluded with a *knowledge refinement and development moment*.

The *knowledge refinement and development moment* (Tim-3.3) is:

> *The moment when knowledge refinement and development is initiated.*

This moment is followed by a period during which knowledge refinement and development is performed, which is concluded with a *knowledge dissemination moment*.

The *knowledge dissemination moment* (Tim-3.4) is:

> *The moment when knowledge dissemination is initiated.*

This moment is followed by a period where knowledge dissemination is performed, which is followed by a *knowledge maintenance moment*.

The periods following the *knowledge identification and extraction moment*, *knowledge structuring and refinement moment*, *knowledge refinement and development moment* and *knowledge dissemination moment* may however also be concluded with another *knowledge creation and extraction moment*. This moment initiates a feedback loop in which new knowledge is identified and extracted based on existing knowledge for a given period.

The *knowledge maintenance moment* (Tim-3.5) is:

> *The moment when knowledge maintenance is initiated.*

This moment is followed by a period during which knowledge maintenance is performed, which is followed by a *knowledge life cycle conclusion moment*.

The *knowledge life cycle conclusion moment* (Tim-3.6) is:

> *The moment when knowledge maintenance is completed.*

This moment signals the end of the *knowledge life cycle*.

Based on the preceding discussion, the constituent moments of the *knowledge life cycle* (Tim-3) timing cycle share the following intervals (refer to Figure 8.33) (2):

- A knowledge creation interval between the *knowledge creation and extraction moment* and the *knowledge structuring and formalisation moment*, or *knowledge creation and extraction moment*.

  Tim-3.1 → Tim-3.2, Tim-3.1
- A knowledge formalisation interval between the *knowledge structuring and formalisation moment* and the *knowledge refinement and development moment*, or *knowledge creation and extraction moment*.

  Tim-3.2 → Tim-3.3, Tim-3.1
- A knowledge refinement interval between the *knowledge refinement and development moment* and *knowledge dissemination moment*, or *knowledge creation and extraction moment*.

  Tim-3.3 → Tim-3.4, Tim-3.1
- A knowledge dissemination interval between the *knowledge dissemination moment* and the *knowledge maintenance moment*, or *knowledge creation and extraction moment*.

  Tim-3.4 → Tim-3.5, Tim-3.1
- A knowledge maintenance interval between the *knowledge maintenance*

  Tim-3.5 → Tim-3.6

moment and the *knowledge life cycle conclusion moment*.

Based on the preceding discussion, the *knowledge life cycle* (Tim-3) and its constituent moments show **associations** (refer to Table 8.21) (D) with the following cycles in the timing abstraction (column):

- *Technology life cycle* (Tim-1; refer to Figure 8.34 (3)):
    - The *knowledge life cycle* also associates with the *technology life cycle* through technology maturity enabling knowledge development and diffusion regarding the technology itself.

Tim-3 → Tim-1

- *Network life cycle* (Tim-2; refer to Figure 8.34 (4)):
    - The *knowledge life cycle* associates with the *network life cycle* through network maturity that enables knowledge development and diffusion in the given domain of network formation.

Tim-3 → Tim-2

- *Innovation life cycle* (Tim-4; refer to Figure 8.34 (5)):
    - The *knowledge life cycle* associates with the *innovation life cycle* through synchronisation between the two life cycles as the innovation process progresses.

Tim-3 → Tim-4

The *knowledge life cycle* (Tim-3) and its constituent moments furthermore show **integrations** (refer to Table 8.21) (E) with the following concepts in other abstractions (columns) of the reference architecture for IKNs:

- *Knowledge* – the cycle directs the development and management of entities of the *knowledge* inventory set.

Tim-3 ↔ Inv-1

- *Knowledge creation and transfer* – the cycle shares deep ties with this process, as it provides the timing for the execution of the process.

Tim-3 ↔ Proc-2

- *Knowledge owners* – the cycle provides the timing for the involvement of *knowledge owners* in the activities of the IKN.

Tim-3 ↔ Resp-3

- *Life cycle rules* – this motivational end determines the hierarchy, sequence, interaction and progression of life cycles in the IKN.

Tim-3 ↔ Mot-3.3

- *Knowledge creation and transfer rules* – this motivational end influences the management and governance of the *knowledge life cycle*.

Tim-3 ↔ Mot-3.5

8.5.7.4.4 Innovation life cycle

This timing cycle definition model (C5, R2.2.4) defines the *innovation life cycle* (Tim-4) as a timing cycle at an intermediate level of the detail. The textual version of this model is initially presented in Table 8.22 and the visual version in Figure 8.35.

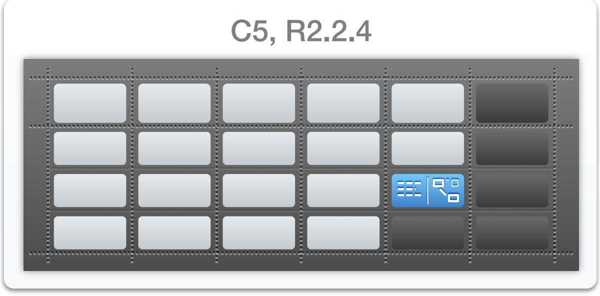


Table 8.22: Timing cycle definition model – Innovation life cycle (C5, R2.2.4)

	Code	Name / definition
A. Concept	**Tim-4**	**Innovation life cycle**
B. Definition		The series of stages through which an innovation passes from invention, concept development, feasibility testing, implementation to market exploitation.
C. Deconstruction	Tim-4.1	Innovation trigger moment
	Tim-4.2	Idea filtering moment
	Tim-4.3	Concept filtering moment
	Tim-4.4	Concept re-defining moment
	Tim-4.5	Project funding moment
	Tim-4.6	Project launch moment
	Tim-4.7	Project implementation moment
	Tim-4.8	Project re-planning moment
	Tim-4.9	Innovation exploitation moment
	Tim-4.10	Innovation life cycle conclusion moment
D. Associations	Tim-1	Technology life cycle
	Tim-2	Network life cycle
	Tim-3	Knowledge life cycle
E. Integrations	Inv-2	Innovation artefacts
	Inv-3	Resources
	Proc-3	Innovation
	Resp-2	Innovators

	Code	Name / definition
	Mot-3.3	Life cycle rules
	Mot-3.6	Innovation process rules

The *innovation life cycle* (Tim-4) timing cycle is **defined** (refer to Table 8.22) (B) as:

> *The series of stages through which an innovation passes from invention, concept development, feasibility testing, implementation to market exploitation.*

The *innovation life cycle* timing cycle is **deconstructed** (refer to Table 8.22 (C), Figure 8.35 (1)) into eight constituent moments, namely *innovation trigger moment*, *idea filtering moment*, *concept filtering moment*, *project funding moment*, *project launch moment*, *project implementation moment*, *innovation exploitation moment* and *innovation life cycle conclusion moment*, via “part-of/has-part” relationships.

Tim-4→
Tim-4.1, Tim-4.2,
Tim-4.3, Tim-4.4,
Tim-4.5, Tim-4.6,
Tim-4.7, Tim-4.8

The *idea generation moment* (Tim-4.1) is:

> *The moment when idea generation and identification is initiated.*

This moment is followed by a period during which idea collection and capture is performed, which is concluded with an *idea filtering moment*.

The *idea filtering moment* (Tim-4.2) is:

> *The moment when ideas are filtered to select those that align with the network innovation strategy.*

This moment is followed by a period during which concept definition is performed, which is concluded with a *concept filtering moment*.

The *concept filtering moment* (Tim-4.3) is:

> *The moment when concepts are filtered to select those that are most promising and require further evaluation in order to determine their feasibility.*

This moment is followed by a period during which concept feasibility testing and refinement is performed. This period may be concluded with a *project funding moment*, or a feedback loop may be initiated where the concept is re-defined to improve its feasibility. This re-definition is initiated by a *concept re-defining moment*.

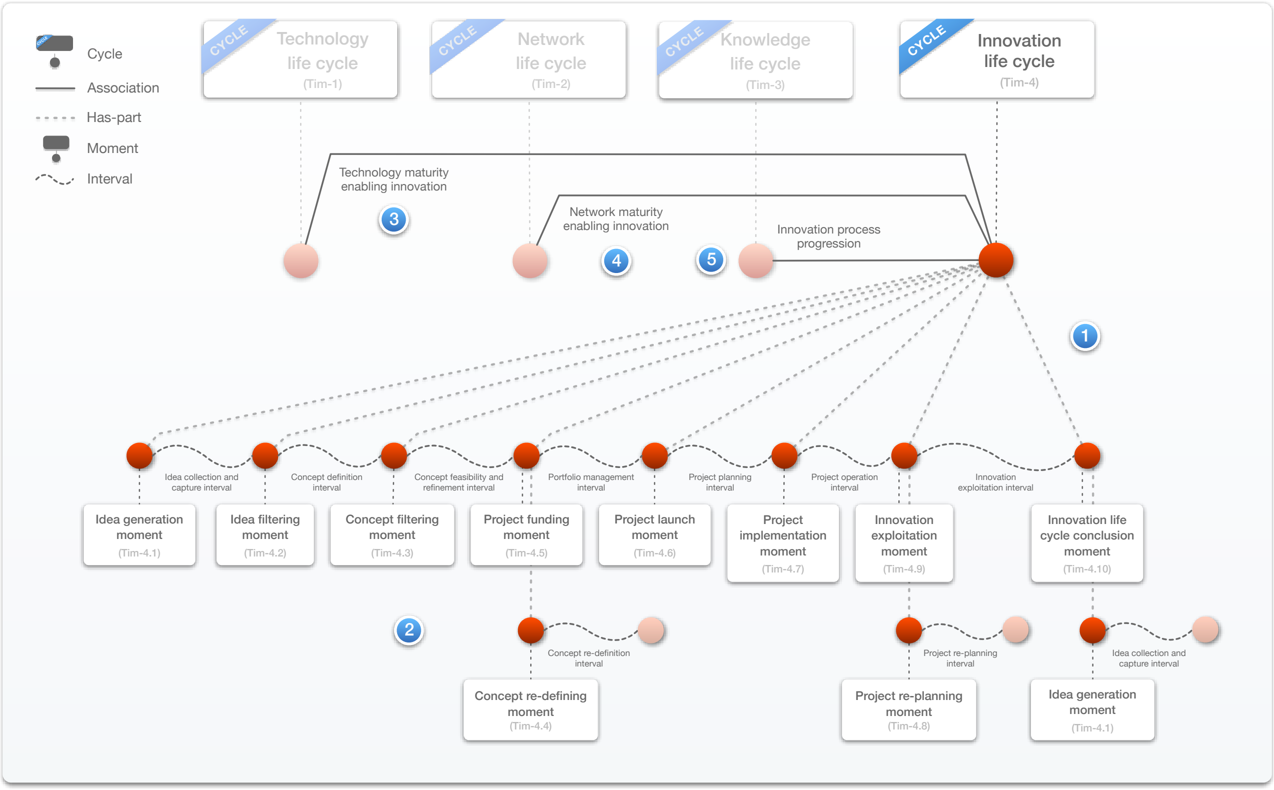


Figure 8.35: Timing cycle definition model – Innovation life cycle (C5, R2.2.4)

The *concept re-defining moment* (Tim-4.4) is:

> *The moment when an innovative concept is re-defined in order to improve its feasibility.*

This moment is followed by a period of concept re-definition, and could be concluded with a *project funding moment* or another similar feedback loop could be initiated.

The *project funding moment* (Tim-4.5) is:

> *The moment when the funding of an innovation project for implementation is approved.*

This moment is followed by a period of portfolio management during which holistic management of the network's innovation initiatives is performed. This period is concluded with a *project launch moment* that occurs at a point in time as determined as part of the portfolio management procedure.

The *project launch moment* (Tim-4.6) is:

> *The moment when an innovation project is launched.*

This moment is followed by a period during which project planning is performed, which is concluded with a *project implementation moment* that occurs at a point in time as determined as part of the project planning procedure.

The *project implementation moment* (Tim-4.7) is:

> *The moment when an innovation project is implemented.*

This moment is followed by a period where the innovation project is operational. This period may be concluded with an *innovation exploitation moment*, or a feedback loop may be initiated where the project is re-planned to improve its performance. This re-planning is initiated by a *project re-planning moment*.

The *project re-planning moment* (Tim-4.8) is:

> *The moment when re-planning of the innovation project is initiated.*

This moment is followed by a period of project re-planning, which is concluded with a *project implementation moment* or another similar feedback loop could be initiated.

The *innovation exploitation moment* (Tim-4.9) is:

> *The moment when exploitation of the existing innovation is initiated.*

This moment is followed by a period during which the existing innovation is exploited, either in a technical or business sense. This period may be concluded with an *innovation life cycle conclusion moment*, or a feedback loop may be initiated where the existing innovation is the inspiration for a new innovative idea. This new iteration of the *innovation life cycle* is initiated by an *idea generation moment*.

The *innovation life cycle conclusion moment* (Tim-4.10) is:

> *The moment when the innovation life cycle is completed.*

This moment signals the end of the *innovation life cycle*.

Based on the preceding discussion, the constituent moments of the *knowledge life cycle* (Tim-3) timing cycle share the following intervals (refer to Figure 8.33) (2):

- An idea collection and capture interval between the *idea generation moment* and the *idea filtering moment.* Tim-4.1 → Tim-4.2
- A concept definition interval between the *idea filtering moment* and the *concept filtering moment*. Tim-4.2 → Tim-4.3
- A concept feasibility and refinement interval between the *concept filtering* Tim-4.3 → Tim-4.5

moment and the *project funding moment*, or another *concept re-defining moment*.

- A concept re-definition interval between the *concept re-defining moment* and the *project funding moment*, or another *concept re-defining moment*. Tim-4.4 → Tim-4.5, Tim-4.4
- A portfolio management interval between the *project funding moment* and the *project launch moment*. Tim-4.5 → Tim-4.6
- A project planning interval between the *project launch moment* and the *project implementation moment*. Tim-4.6 → Tim-4.7
- A project operation interval between the *project implementation moment* and the *innovation exploitation moment*, or the *project re-planning moment*. Tim-4.7 → Tim-4.9, Tim-4.8
- A project re-planning interval between the *project re-planning moment* and the *project implementation moment*, or another *project re-planning moment*. Tim-4.8 → Tim-4.7, Tim-4.8
- An innovation exploitation interval between the *innovation exploitation moment* and the *innovation life cycle conclusion moment*, or a new *idea generation moment*. Tim-4.9 → Tim-4.10, Tim-4.1

The *innovation life cycle* (Tim-4) and its constituent moments show **associations** (refer to Table 8.22) (D) with the following cycles in the timing abstraction (column):

- *Technology life cycle* (Tim-1; refer to Figure 8.35 (3)):
  - The *innovation life cycle* associates with the *technology life cycle* through innovations that may accelerate existing instances of the *technology life cycle*, or initiate new instances of it. Tim-4 → Tim-1

- *Network life cycle* (Tim-2; refer to Figure 8.35 (4)):
  - The *innovation life cycle* associates with the *network life cycle* through network maturity that enables the network to pursue its innovative purpose. Tim-4 → Tim-2

- *Knowledge life cycle* (Tim-3; refer to Figure 8.35 (5)):
  - The *knowledge life cycle* associates with the *innovation life cycle* through synchronisation between the two life cycles, as well as both life cycles initiating new instances of the other. Tim-4 → Tim-3

The *innovation life cycle* (Tim-4) and its constituent moments furthermore show **integrations** (refer to Table 8.22) (E) with the following concepts in other abstractions (columns) of the reference architecture for IKNs:

- *Innovation artefacts* – the cycle directs the development of entities from this inventory set. Tim-4 ↔ Inv-2

- *Resources* – the cycle determines when entities from this inventory set are required to support the *innovation process*, and also when the *innovation* process contributes to these *resources*.

Tim-4 ↔ (Proc-3 ⟶) Inv-3

- *Innovation* – the cycle shares deep ties with this process, as it provides the timing for the execution of the process.

Tim-4 ↔ Proc-3

- *Innovators* – the cycle determines when the various constituent roles of this responsibility assignment are required to contribute to the *innovation* process.

Tim-4 ↔ (Proc-3 ⟶) Resp-2

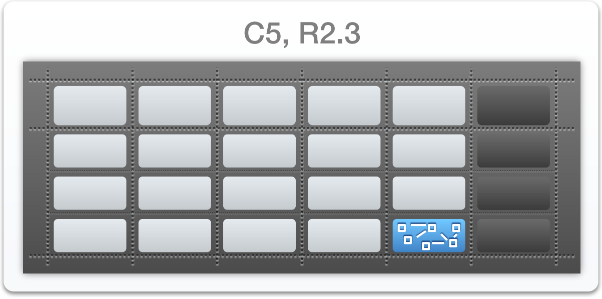
- *Life cycle rules* – this motivational end determines the hierarchy, sequence, interaction and progression of life cycles in the IKN.

Tim-4 ↔ Mot-3.3

- *Innovation process rules* – this motivational end influences the management and governance of the *knowledge life cycle*.

Tim-4 ↔ Mot-3.6

8.5.7.5 Extended timing definition model



The extended timing definition model (C5, R2.3) defines the timing cycles identified in the timing identification model (C5, R1; refer to Table 8.18) at a high level of detail by representing them visually according to the abstraction meta-model (refer to Figure 8.30). This model is based on the interpretation of the *definition data* from the qualitative systematic review described in section 8.5.2.1, with regard to the "*When?*" review question.

The various constituent moments are indicated and organised into swim lanes according to their overarching high-level timing cycles (refer to Figure 8.36). The detailed intervals between them are also labelled appropriately. This model therefore retains all the defining deconstructions and associations from the timing cycle definition models in the preceding sections.

No deconstruction of roles into sub-moments is done for the extended timing definition model. To ensure that the model remains a primitive (single-variable) model, as well as for readability reasons, integrations with concepts in other abstractions (columns) are not indicated in the model.

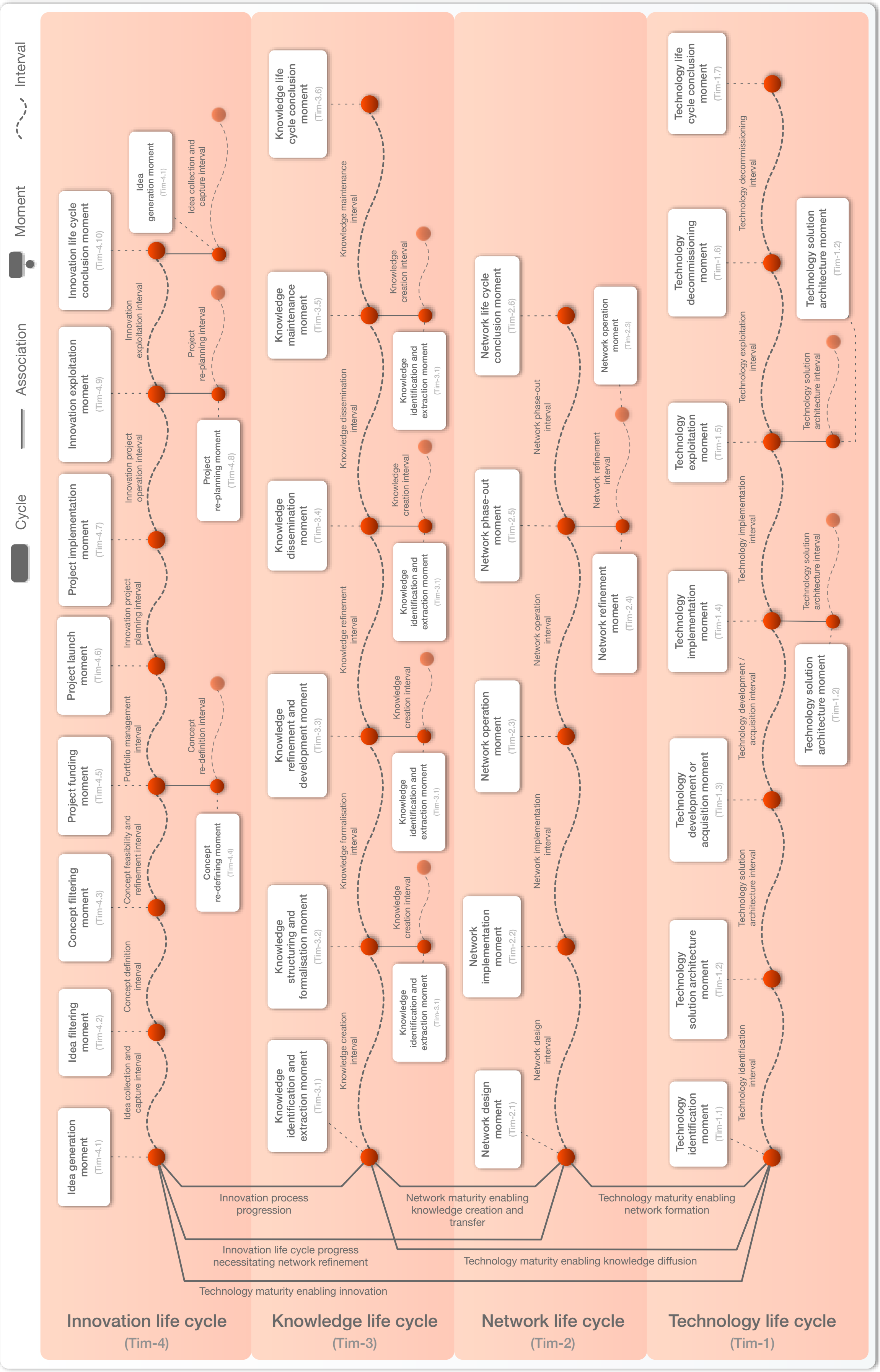


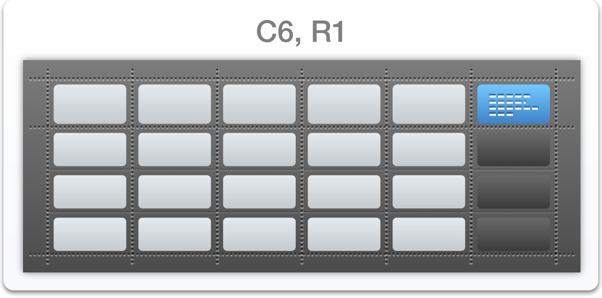
Figure 8.36: Extended timing definition model

8.5.8 Motivation intention models

This section discusses the motivation abstraction (sixth column) of the artefact, and the models presented therefore answer the interrogative of "Why?" in the engineering of IKNs. The top-row reference model is presented and motivated first. Thereafter the meta-model for the second row of the sixth column is presented, followed by the various second-row reference models according to increasing levels of detail.

Where the description of a model includes a reference to another formalised concept in the reference architecture, the name of the concept is printed in italics.

8.5.8.1 Motivation identification model



The motivation identification model (C6, R1; refer to Table 8.23) lists motivation types that are significant and common to the engineering of IKNs. This model is based on the interpretation of the *identification data* from the qualitative systematic review described in section 8.5.2.1, with regard to the "*Why?*" review question.

At the lowest level of detail, three main motivation types are identified, namely *value creation*, *competitiveness* and *innovation intention*. Constituent motivation subtypes are also identified that classify these types at a higher level of detail.

Table 8.23: Motivation identification model (C6, R1)

Code	Motivation type	Code	Constituent motivation subtypes
Mot-1	**Value creation**	Mot-1.1	Stakeholder wealth
		Mot-1.2	Stakeholder prosperity
Mot-2	**Competitiveness**		
Mot-3	**Innovation intention**	Mot-3.1	Innovation intention
		Mot-3.2	Network domain and purpose
		Mot-3.3	Life cycle rules
		Mot-3.4	Network building rules
		Mot-3.5	Knowledge creation and transfer rules
		Mot-3.6	Innovation role rules

Code	Motivation type	Code	Constituent motivation subtypes
		Mot-3.7	Innovation process rules
		Mot-3.8	Geographical strategy
		Mot-3.9	Operational requirement rules

Value creation is ultimately the primary motivation for the existence of an IKN, as in all cases the purpose of the network is to create value for its stakeholders in some form or another. This value could either be of pecuniary or non-pecuniary nature, and the *value creation* motivation type is therefore deconstructed into motivation subtypes of *stakeholder wealth* and *stakeholder prosperity*. The second related key motivation type for IKNs is *competitiveness*. In principle, *network stakeholders* collaborate in an IKN in order to improve their *competitiveness*, which in turn enables them to *create value.* A third key motivation type for IKNs is their *innovation intention* that allows them to innovate, thereby improving the *competitiveness* of their members. The *innovation intention* of an IKN contains all the motivations for the rest of the network's configuration and operation, and is therefore deconstructed into motivation subtypes of *life cycle rules*, *network building rules*, *knowledge creation and transfer rules*, *innovation role rules*, *innovation process rules*, *geographical strategy* and *operational requirement rules*.

Definitions for the motivation types and subtypes identified in the motivation identification model are provided in the second-row motivation definition models presented in sections 8.5.8.3, 8.5.8.4 and 8.5.8.5.

8.5.8.2 Meta-model for definition models in motivation abstraction

The primitive meta-model presented in Figure 8.37 serves as the semi-formal representational language for the various second-row definition models in the motivation abstraction, or '*why*' column, of the reference architecture for IKNs.

This meta-model assists in defining *motivation intentions* as concepts and follows the single-variable structure of "*end-means-end"* amongst its elements, links, sub-elements and sub-links. The meta-model therefore deconstructs and represents motivation intentions into ends and sub-ends, along with their respective means and sub-means.

This representation is done at various levels of detail in the basic motivation definition model (refer to section 8.5.8.3), motivation intentions definition models (refer to section 8.5.8.4) and extended motivation definition model (refer to section 8.5.8.5). For all of these models, the meta-model scope indication shown at the top of Figure 8.37 applies.

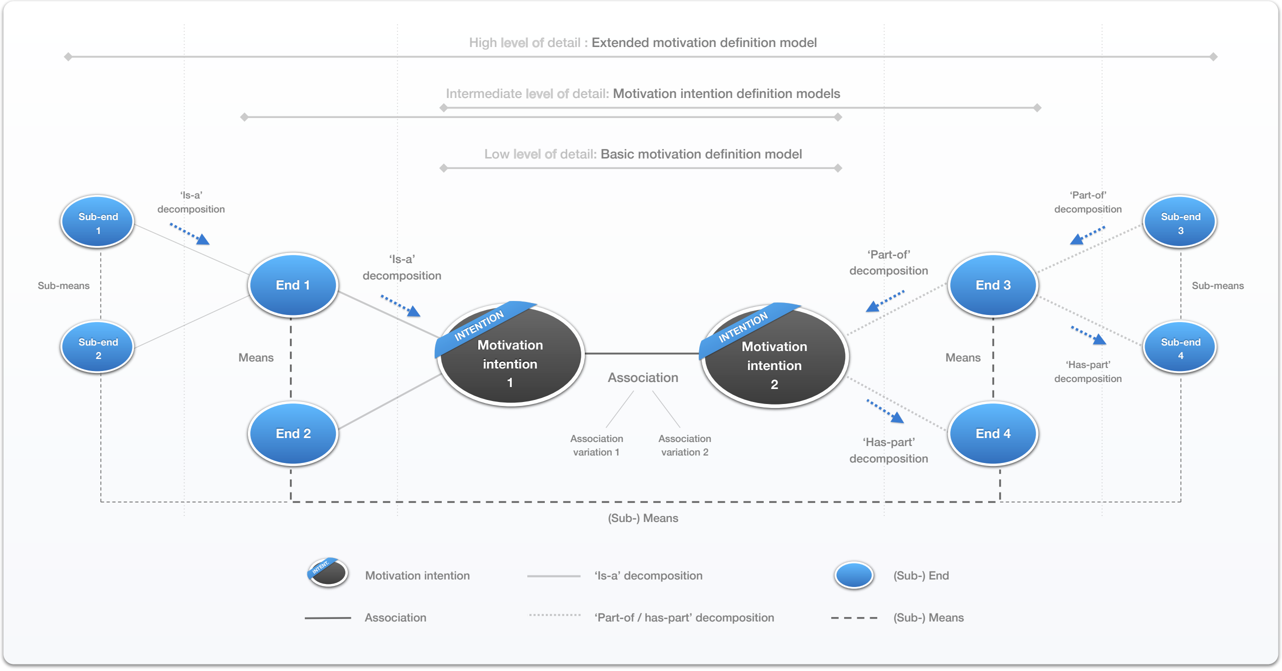
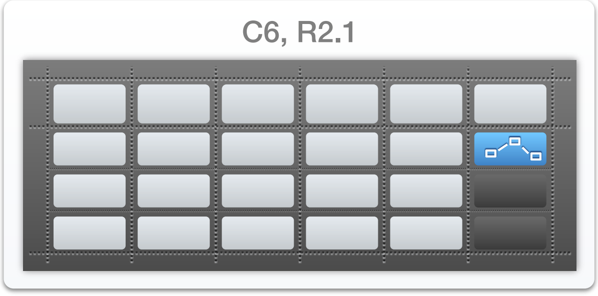


Figure 8.37: Meta-model for definition models in motivation abstraction

8.5.8.3 Basic motivation definition model



The basic motivation definition model (C6, R2.1) defines the motivation intentions that were identified as motivation types in the motivation identification model (C6, R1; refer to Table 8.23) at a low level of detail by representing them visually according to the abstraction meta-model (refer to Figure 8.37). This model is based on the interpretation of the *definition data* from the qualitative systematic review described in section 8.5.2.1, with regard to the "*Why?*" review question.

The model depicts the various identified motivation intentions, as well as the associations between them (refer to Figure 8.8). This model shows that the *innovation intention* of an IKN aims at delivering improved *competitiveness* to the members of the network through sustainable innovation. This allows the stakeholders in the network to profit from *value created* through their improved *competitiveness*. The *innovation intention* of an IKN, however, may also deliver *value creation* to the stakeholders in the network without doing it through the improvement of their *competitiveness*. This occurs in cases where the stakeholders in an IKN profit from a radical innovation that does not compete with any existing offerings.

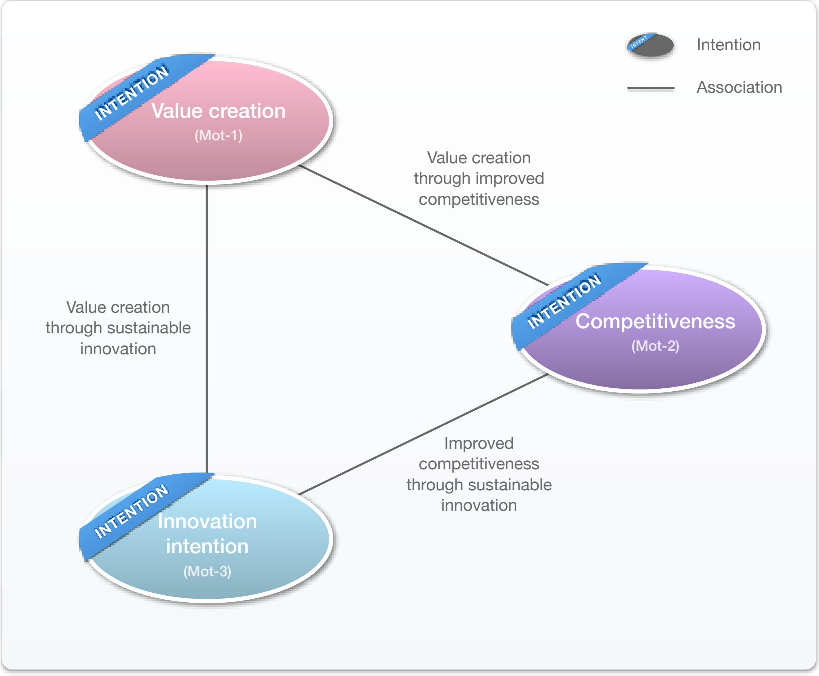


Figure 8.38: Basic motivation definition model (C6, R2.1)

These motivation intentions are now defined in more detail in the relevant motivation intention definition models in section 8.5.8.4.

8.5.8.4 Motivation intention definition models

The motivation intention definition models presented in this section each define an identified motivation intention at an intermediate level of detail. The models therefore add new information for each intention, while retaining all the information from the basic inventory definition model (refer to Figure 8.8). Both textual and visual versions are provided to assist in defining the intentions. Deconstructions, relationships and integrations between concepts are expressed in terms of identification codes in side-notes for traceability purposes.

The models presented in this section are based on the interpretation of the *definition data* for each identified motivation intention, as obtained from the qualitative systematic review described in section 8.5.2.1.

8.5.8.4.1 Value creation

This motivation intention definition model (C6, R2.2.1) defines *value creation* (Mot-1) as a motivation intention at an intermediate level of the detail. The textual version of this model is initially presented in Table 8.24 and the visual version in Figure 8.39.

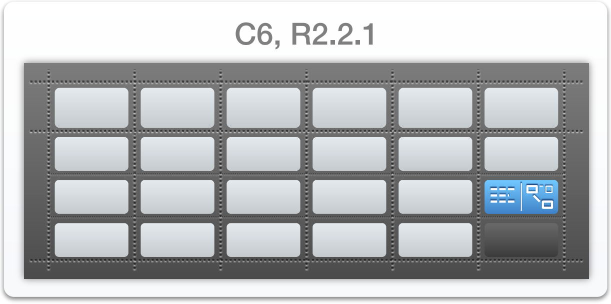


Table 8.24: Motivation intention definition model – Value creation (C6, R2.2.1)

	Code	**Name / definition**
A. Concept	**Mot-1**	**Value creation**
B. Definition		The act of creating and delivering value to stakeholders.
C. Deconstruction	Mot-1.1	Stakeholder wealth
	Mot-1.2	Stakeholder prosperity
D. Associations	Mot-2	Competitiveness
	Mot-3	Innovation intention
E. Integrations	Resp-1	Network stakeholders

The *value creation* motivation intention is **defined** (refer to Table 8.24) (B) as:

> *The act of creating and delivering value to stakeholders.*

The *value creation* motivation intention is **deconstructed** (refer to Table 8.24 (C), Figure 8.39 (1)) into two constituent ends, namely *stakeholder wealth* and *stakeholder prosperity*, via “is-a” relationships.

Mot-1 →
Mot-1.1, Mot-1.2

Stakeholder wealth (Mot-1.1) is:

> *Pecuniary value created for stakeholders in an IKN.*

Value created for the stakeholders in an IKN can be of a pecuniary nature, e.g. the revenue created by innovative *products* or *services*. It is highly unlikely that organisations or individuals that are not stakeholders in the network will share in this form of *created value* due to measures taken by *network stakeholders* as part of their *innovation intention* to protect their financial interests.

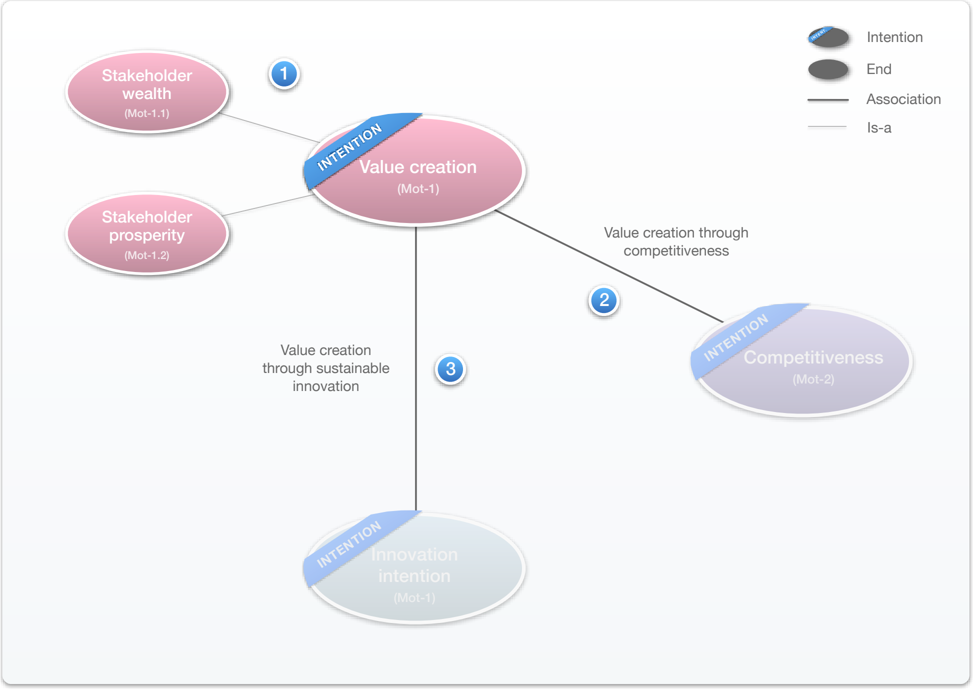


Figure 8.39: Motivation intention definition model – Value creation (C6, R2.2.1)

Stakeholder prosperity (Mot-1.2) is:

> *Non-pecuniary value created for stakeholders in an IKN.*

Value created for the stakeholders in an IKN, however, can also be of a non-pecuniary nature, e.g. an increase in living standards as a result of innovation *products* or *services*. Organisations or individuals that are not members of the network may also share in this form of *created value*.

Based on the preceding discussion, *value creation* (Mot-1) and its constituent ends show **associations** (refer to Table 8.24) (D) with the following intentions in the motivation abstraction (column):

- *Competitiveness* (Mot-2; refer to Figure 8.39 (2)):
    - The increased *competitiveness* of organisations that partake in IKNs *creates value* for the stakeholders in these networks.

Mot-2 → Mot-1

- *Innovation intention* (Mot-3; refer to Figure 8.39 (3)):
    - *Value creation* is achieved through the sustained innovation in an IKN, enabled by the pursuit of the network's *innovation intention*.

Mot-1 → Mot-3

Value creation (Mot-1) and its constituent ends furthermore show **integrations** (refer to Table 8.24) (E) with the following concepts in other abstractions (columns) of the reference architecture for IKNs:

- *Network stakeholders* – this responsibility assignment benefits from the *value created* by an IKN.

Mot-1 ↔ Resp-1

8.5.8.4.2 Competitiveness

This motivation intention definition model (C6, R2.2.2) defines *competitiveness* (Mot-2) as a motivation intention at an intermediate level of the detail. The textual version of this model is initially presented in Table 8.25 and the visual version in Figure 8.40.

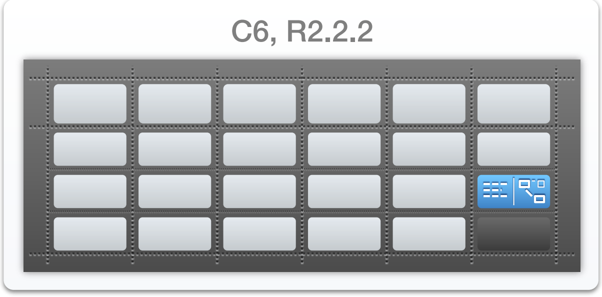


Table 8.25: Motivation intention definition model – Competitiveness (C6, R2.2.2)

	Code	**Name / definition**
A. Concept	**Mot-2**	**Competitiveness**
B. Definition		An organisation's ability to create value in the market on a level that is as good as or better than organisations of comparable nature.
C. Associations	Mot-1	Value creation
	Mot-3	Innovation intention
D. Integrations	Resp-1	Network stakeholders

The *competitiveness* motivation intention is **defined** (refer to Table 8.25) (B) as:

> *An organisation's ability to create value in the market on a level that is as good as or better than organisations of comparable nature.*

The *competitiveness* of members of an IKN, be it organisations or individuals, is increased due to the pursuit of the network's *innovation intention* . This *competitiveness* could manifest in the form of increased ability of *network stakeholders* to deliver innovative *products* or *services* and hence *create* pecuniary *value*. Increased *competitiveness,* however, could also manifest in an increased ability of *network stakeholders*, or organisations or individuals close to the network, to gain access to higher living standards and hence *create* non-pecuniary *value*.

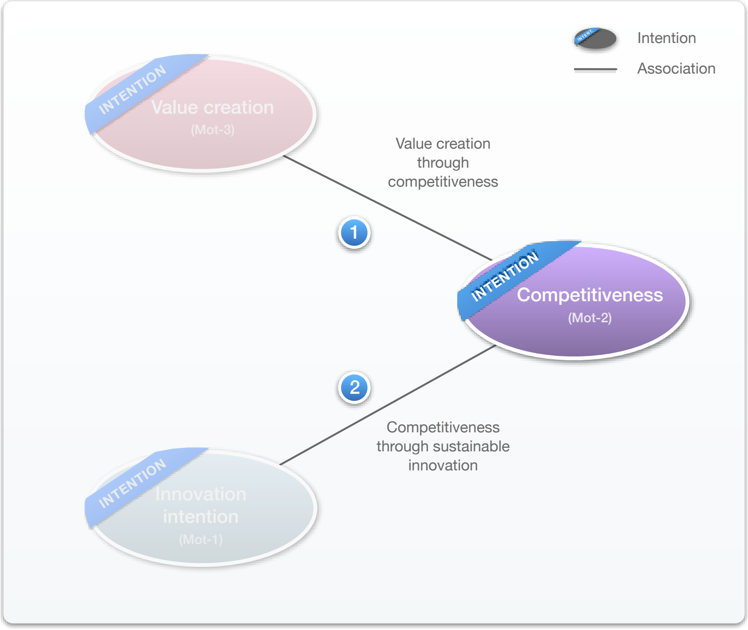


Figure 8.40: Motivation intention definition model – Competitiveness (C6, R2.2.2)

Competitiveness (Mot-2) shows **associations** (refer to Table 8.25) (C) with the following intentions in the motivation abstraction (column):

- *Value creation* (Mot-1; refer to Figure 8.40 (1)):
    - The increased *competitiveness* of organisations that partake in IKNs *creates value* for these stakeholders.

Mot-2 → Mot-1

- *Innovation intention* (Mot-3; refer to Figure 8.40 (2)):
    - When the *innovation intention* of an IKN is pursued, it leads to the increased *competitiveness* of its members.

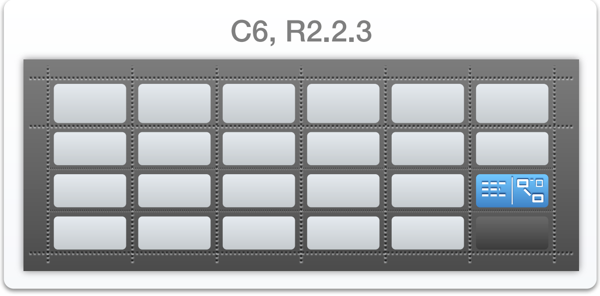
Mot-3 → Mot-2

Competitiveness (Mot-2) furthermore shows **integrations** (refer to Table 8.25) (D) with the following concepts in other abstractions (columns) of the reference architecture for IKNs:

- *Network stakeholders* – it is the *competitiveness* of the constituent members of an IKN that is improved through their participation in the network.

Mot-2 ↔ Resp-1

8.5.8.4.3 Innovation intention



This motivation intention definition model (C6, R2.2.3) defines *innovation intention* (Mot-3) as a motivation intention at an intermediate level of the detail. The textual version of this model is initially presented in Table 8.26 and the visual version in Figure 8.41.

Table 8.26: Motivation intention definition model – Innovation intention (C6, R2.2.3)

	Code	**Name / definition**
A. Concept	**Mot-3**	**Innovation intention**
B. Definition		The intention of an IKN to innovate in order to promote the competitiveness of network stakeholders and create value.
C. Deconstruction	Mot-3.1	Innovation strategy
	Mot-3.2	Network domain and purpose
	Mot-3.3	Life cycle rules
	Mot-3.4	Network building rules
	Mot-3.5	Knowledge creation and transfer rules
	Mot-3.6	Innovation role rules
	Mot-3.7	Innovation process rules
	Mot-3.8	Geographical strategy
	Mot-3.9	Operational requirement rules
D. Associations	Mot-1	Value creation
	Mot-2	Competitiveness
E. Integrations	Inv-1	Knowledge
	Inv-2	Innovation artefacts
	Inv-3	Resources
	Proc-1	Network building
	Proc-2	Knowledge creation and transfer
	Proc-3	Innovation

	Code	Name / definition
	Dist-1	Strategic network
	Dist-2	Operational network
	Resp-1	Network stakeholders
	Resp-2	Innovators
	Resp-3	Knowledge owners
	Tim-1	Technology life cycle
	Tim-2	Network life cycle
	Tim-3	Knowledge life cycle
	Tim-4	Innovation life cycle

The *innovation intention* (Mot-3) motivation intention is **defined** (refer to Table 8.26) (B) as:

> *The intention of an IKN to innovate in order to promote the competitiveness of network stakeholders and create value.*

This pursued *innovation intention* is the concept that can promote sustainable innovation in the network, thereby achieving improved *competitiveness* and *value creation*.

The *innovation intention* motivation intention is **deconstructed** (refer to Table 8.26 (C), Figure 8.41 (1)) into nine constituent ends, namely *innovation strategy*, *network domain and purpose, life cycle rules*, *network building rules*, *knowledge creation and transfer rules*, *innovation role rules*, *innovation process rules*, *geographical strategy* and *operational requirement rules*, via ”part-of/has-part” relationships.

Mot-3 →
Mot-3.1, Mot-3.2,
Mot-3.3, Mot-3.4,
Mot-3.5, Mot-3.6,
Mot-3.7, Mot-3.8,
Mot-3.9

Innovation strategy (Mot-3.1) is:

> *An organisation’s innovation-specific strategy, including its innovation objectives and the means it uses to achieve them. The execution of the innovation strategy of an IKN entails the way in which various elements of the strategy are integrated to pursue the innovation intention of the network.*

The other constituent ends of the *innovation intention* motivation intention are ultimately not ends unto themselves and only become valuable when they are integrated into an *innovation strategy*.

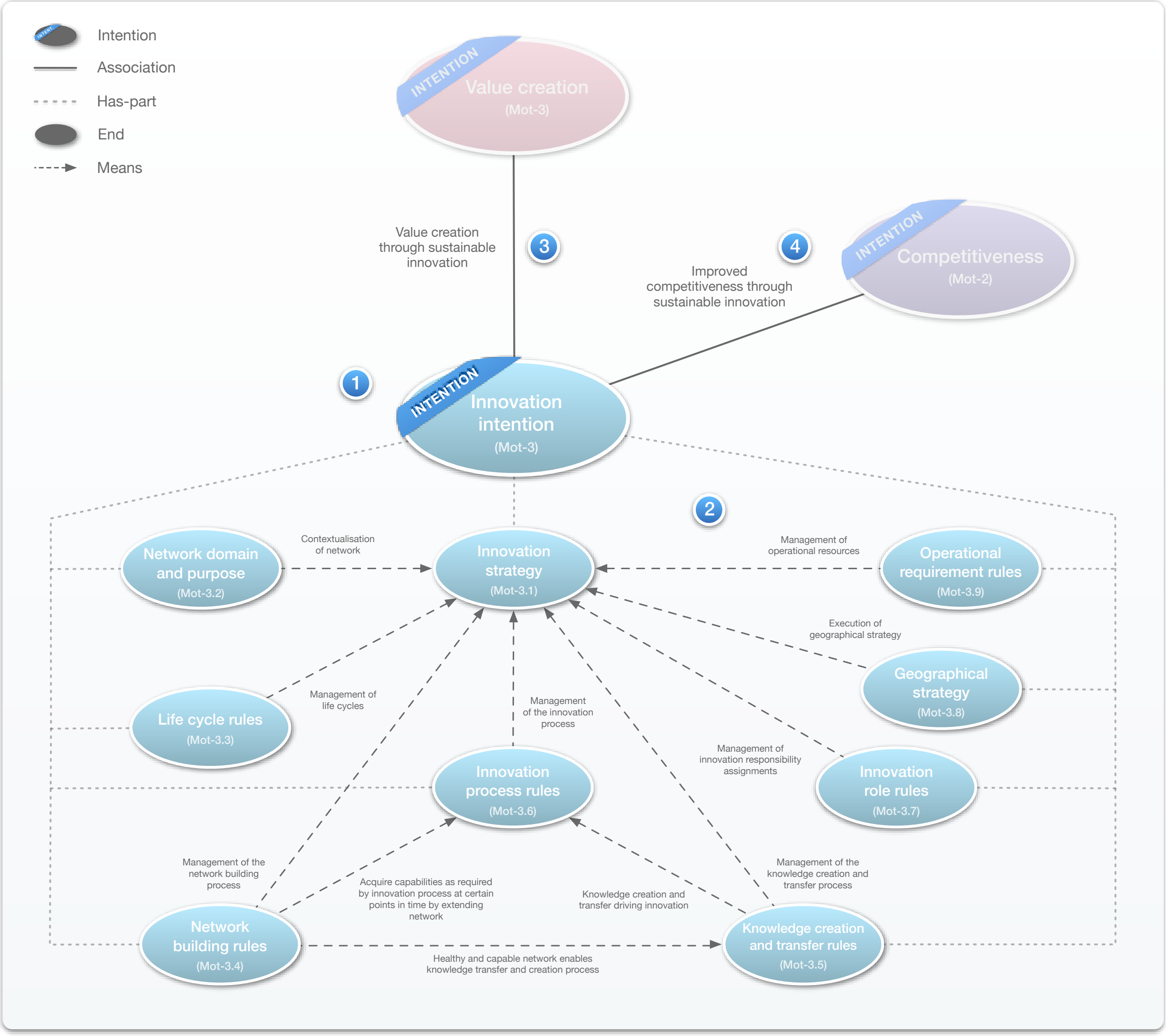


Figure 8.41: Motivation intention definition model – Innovation intention (C6, R2.2.3)

The *network domain and purpose* (Mot-3.2) is:

> *The context against which the network operates, including its domain and the domain-specific reason for its existence.*

While *value creation* and improved *competitiveness* have been identified and defined as the elements that justify the existence of an IKN, certain domain-specific elements provide a specific strategic context for the network. This could include a specific form of innovation that the network is expected to produce, as well as a particular domain-specific need for knowledge sharing.

Life cycle rules (Mot-3.3) are:

> *The rules that govern the hierarchy, sequence, interaction and progression of life cycles in an IKN.*

Although the exact hierarchy, sequence, interaction and progression of life cycles in IKNs will differ for every given instance of such a network, some version of *life cycle rules* will always exist, even if only as implicit logic. For instance, in some cases progress along the *technology life cycle* will initiate a *network life cycle*, while in other cases this hierarchy and sequence may reverse.

Network building rules (Mot-3.4) are:

> *The rules that govern the execution of the network building process.*

These rules govern the initiation, progress and conclusion of the *network building* process.

Knowledge creation and transfer rules (Mot-3.5) are:

> *The rules that govern the execution of the knowledge creation and transfer process.*

These rules govern the initiation, progress and conclusion of the *knowledge creation and transfer* process.

Innovation process rules (Mot-3.6) are:

> *The rules that govern the execution of the innovation process.*

These rules govern the initiation, progress and conclusion of the *innovation* process.

Innovation role rules (Mot-3.7) are:

> *The rules that govern the allocation of innovation roles to network stakeholders.*

These rules govern the assignment of innovative responsibilities to *network stakeholders*, and constitute the innovation products that they create and exchange.

Geographical strategy (Mot-3.8) is:

> *The strategy that governs how an IKN develops and integrates its strategic and operational networks of locations.*

This strategy determines how an IKN develops and populates its strategic network of locations to achieve its innovative objectives. It also determines how an IKN develops and populates its operational network of locations to integrate with its strategic network.

Operational requirement rules (Mot-3.9) are:

The rules that govern how operational requirements are addressed in an IKN.

These rules for instance govern the sharing of *resources* amongst *network stakeholders* to achieve operational objectives.

Based on the preceding discussion, the constituent ends of the *innovation intention* (Mot-3) motivation intention share the following means (refer to Figure 8.41) (2):

- The contextualisation of the network as a means of employing *network domain and purpose* to achieve the execution of the *innovation strategy* . Mot-3.2 → Mot-3.1
- The management of life cycles as a means of employing *life cycle rules* to achieve the execution of the *innovation strategy* . Mot-3.3 → Mot-3.1
- The management of the network building process as a means of employing *network building rules* to achieve the execution of the *innovation strategy.* Mot-3.4 → Mot-3.1
- A healthy and capable network that enables the knowledge creation and transfer process as a means of employing *network building rules* to achieve *knowledge creation and transfer rules.* Mot-3.4 → Mot-3.6
- The acquisition of capabilities as required by the innovation process at certain points in time by extending the network, as a means of employing *network building rules* to achieve *innovation process rules.* Mot-3.4 → Mot-3.5
- Knowledge creation and transfer driving innovation as a means of employing *knowledge creation and transfer rules* to achieve *innovation process rules.* Mot-3.5 → Mot-3.6
- The management of the knowledge creation and transfer process as a means of employing *knowledge creation and transfer rules* to achieve the execution of the *innovation strategy.* Mot-3.5 → Mot-3.1
- The management of the innovation process as a means of employing *innovation process rules* to achieve the execution of the *innovation strategy.* Mot-3.6 → Mot-3.1
- The management of the innovation responsibility assignments as a means of employing *innovation role rules* to achieve the execution of the *innovation strategy.* Mot-3.7 → Mot-3.1
- The execution of a geographical strategy as a means of employing a *geographical strategy* to achieve the execution of the *innovation strategy.* Mot-3.8 → Mot-3.1
- The management of operational resources as a means of employing *operational requirement rules* to achieve the execution of the *innovation strategy.* Mot-3.9 → Mot-3.1

The *innovation intention* (Mot-3) and its constituent ends show **associations** (refer to Table 8.26) (D) with the following intentions in the motivation abstraction (column):

- *Value creation* (Mot-1; refer to Figure 8.41 (7)):
    - *Value creation* is achieved through the sustained innovation in an IKN, enabled by the pursuit of the network's *innovation intention*.

      Mot-1 → Mot-3
- *Competitiveness* (Mot-2; refer to Figure 8.41 (8)):
    - When the *innovation intention* of an IKN is pursued, it leads to the increased *competitiveness* of its members.

      Mot-3 → Mot-2

An *innovation intention* (Mot-3) and its constituent ends furthermore show **integrations** (refer to Table 8.26) (E) with the following concepts in other abstractions (columns) of the reference architecture for IKNs:

- *Knowledge* – the creation and transfer of knowledge amongst IKN members is one of the fundamental motivations for the existence of the network, and therefore integrates with the *network domain and purpose*.

  Mot-3.2 ↔ Inv-1
- *Innovation artefacts* – *network domain and purpose* is the motivational end for the production and transformation of *innovation artefacts*, as well as their inter-relations.

  Mot-3.2 ↔ Inv-2
- *Resources* – *operational requirement rules* influence the network's approach to sourcing and sharing of *resources*.

  Mot-3.9 ↔ Inv-3
- *Network building* – *network building rules* impact the way in which *network building* is approached and governed in an IKN.

  Mot-3.4 ↔ Proc-1
- *Knowledge creation and transfer*; *knowledge creation and transfer rules* impact the way in which *knowledge creation and transfer* is approached and governed in an IKN.

  Mot-3.5 ↔ Proc-2
- *Innovation* – *innovation process rules* and *innovation role rules* impact the way in which *innovation* is approached and governed in IKNs.

  Mot-3.6, Mot-3.7 ↔ Proc-3
- *Strategic network* – the *geographical strategy* of an IKN determines the way in which its *strategic network* is managed and governed.

  Mot-3.8 ↔ Dist-1
- *Operational network* – the *geographical strategy* of an IKN determines the way in which its *operational network* is managed and governed.

  Mot-3.8 ↔ Dist-2
- *Network stakeholders* – *network building rules* dictate the IKN's approach to management and governance of its *network stakeholders* .

  Mot-3.4 ↔ Resp-1
- *Innovators* – elements of the *innovation strategy*, particularly *innovation role rules* govern the actions of *innovators*.

  Mot-3.7 ↔ Resp-2
- *Knowledge owners* – the *innovation intention*, including the *network domain and purpose*, determines the influence and management of *knowledge owners* in the IKN.

  Mot-3 ↔ Resp-3

- *Technology life cycle* – *life cycle rules* determine the hierarchy, sequence, interaction and progression of life cycles in the IKN.

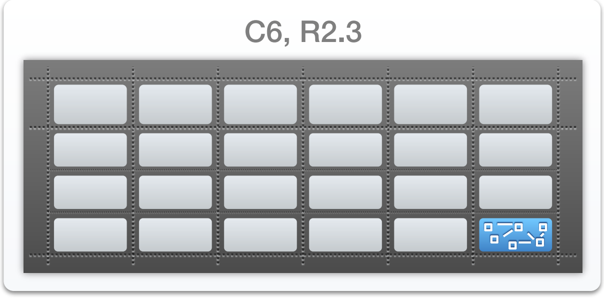
  Mot-3.3 ↔ Tim-1
- *Network life cycle* – *life cycle rules* determine the hierarchy, sequence, interaction and progression of life cycles in the IKN.

  Mot-3.3 ↔ Tim-2
- *Knowledge life cycle* – *life cycle rules* determine the hierarchy, sequence, interaction and progression of life cycles in the IKN, while *knowledge creation and transfer rules* influence the management and governance of the life cycle.

  Mot-3.3, Mot-3.5 ↔ Tim-3
- *Innovation life cycle* – *life cycle rules* determine the hierarchy, sequence, interaction and progression of life cycles in the IKN, while *innovation process rules* influence the management and governance of the life cycle.

  Mot-3.3, Mot-3.6 ↔ Tim-4

8.5.8.5 Extended motivation definition model



The extended motivation definition model (C6, R2.3) defines the motivation intentions identified in the motivation identification model (C6, R1; refer to Table 8.23) at a high level of detail by representing them visually according to the abstraction meta-model (refer to Figure 8.37). This model is based on the interpretation of the *definition data* from the qualitative systematic review described in section 8.5.2.1, with regard to the “*Why?*” review question.

This model retains all the defining deconstructions and associations from the motivation intention definition models in the preceding sections. Where applicable, the various constituent ends, however, are further deconstructed into sub-ends and are organised into swim lanes according to their overarching intention and end (refer to Figure 8.42). The detailed sub-means between these sub-ends are also labelled appropriately.

In the *innovation intention* (Mot-3) motivation intention, the following ends are deconstructed into sub-ends that are linked via sub-means (refer to Figure 8.23):

- *Network domain and purpose* (Mot-3.2): The *network domain and purpose* is achieved by means of determining the network domain, innovation objectives and knowledge sharing needs of the IKN.

- *Life cycle rules* (Mot-3.3): *Life cycle rules* are constituted by means of the management of the hierarchy, temporal relationships, interactions and progression of life cycles. The management of the progression of life cycles is achieved by means of the management of life cycle initiation, progress and conclusion.

The initiation of a life cycle is enabled through the creation of the required starting conditions for the particular life cycle, which could relate to the life cycle hierarchy, sequence and interactions. Progress along a particular life cycle is achieved by means of achieving the outcome of a current moment or interval, not achieving the outcome of a current moment or interval where a feedback loop is defined or initiating the next moment or interval as defined in the life cycle definition. The conclusion of a life cycle is achieved by means of creating the starting conditions for the life cycle conclusion, as well the creation of the starting conditions for a possible iteration of the life cycle.

- *Network building rules* (Mot-3.4): *Network building rules* are constituted by means of the management of the initiation, progress and conclusion of the network building process.

  The initiation of the network building process is enabled through the creation of the required starting conditions. Progress in the process is achieved by means of achieving the outcome of a current transform or I/O, not achieving the outcome of a current transform or I/O where a feedback loop is defined or initiating the next transform or I/O as defined in the process definition. The conclusion of the process is achieved by means of creating the required starting conditions.

- *Knowledge creation and transfer rules* (Mot-3.5): *Knowledge creation and transfer rules* are constituted by means of the management of the initiation, progress and conclusion of the knowledge creation and transfer process. These rules furthermore include the provision of physical connections that facilitate knowledge creation and transfer, as well as the presence of knowledge to drive innovation.

  The initiation of the knowledge creation and transfer process is enabled through the creation of the required starting conditions. Progress in the process is achieved by means of achieving the outcome of a current transform or I/O, not achieving the outcome of a current transform or I/O where a feedback loop is defined or initiating the next transform or I/O as defined in the process definition. The conclusion of the process is achieved by means of creating the required starting conditions for the process conclusion, as well the creation of the starting conditions for a possible iteration of the process.

  The above elements of the *knowledge creation and transfer rules* are achieved through the provision of physical connections that facilitate knowledge creation and transfer in the network. Tacit knowledge transfer is achieved by means of social contact, and explicit knowledge transfer by means of information systems.

  The network is equipped with knowledge to drive the innovation process by means of explicit knowledge captures in information artefacts, as well as tacit knowledge that resides with knowledge owners.

- *Innovation role rules* (Mot-3.6): *Innovation role rules* are constituted through network stakeholders that collaborate by creating and exchanging innovative work products. These coordinated innovation responsibility assignments are enabled by means of innovation roles that are allocated and innovative work products that are defined for these particular roles.

- *Innovation process rules* (Mot-3.7): *Innovation process rules* are constituted by means of the management of the initiation, progress and conclusion of the innovation process. The initiation of the innovation process is enabled through the creation of the required starting conditions. Progress in the process is achieved by means of achieving the outcome of a current transform or I/O, not achieving the outcome of a current transform or I/O where a feedback loop is defined or initiating the next transform or I/O as defined in the process definition. The conclusion of the process is achieved by means of creating the required starting conditions for the process conclusion, as well the creation of the starting conditions for a possible iteration of the process.

- *Geographical strategy* (Mot-3.8): The *geographical strategy* of an IKN is achieved my means of management of both its strategic and operational networks of locations. A strategic network that is aligned with the innovation objectives of the IKN is enabled through population of the network according particular instance-specific objectives. An operational network that is in integrated with this strategic network is achieved by means of knowledge of the intentions of the strategic network, as well as the population of the operational network.

- *Operational requirement rules* (Mot-3.9): *Operational requirement rules* are constituted by the requirements for shared operational resources as is applicable in each IKN instance.

For readability reasons, and to ensure that the model remains a primitive (single-variable) model, integrations with concepts in other abstractions (columns) are not indicated in the model.

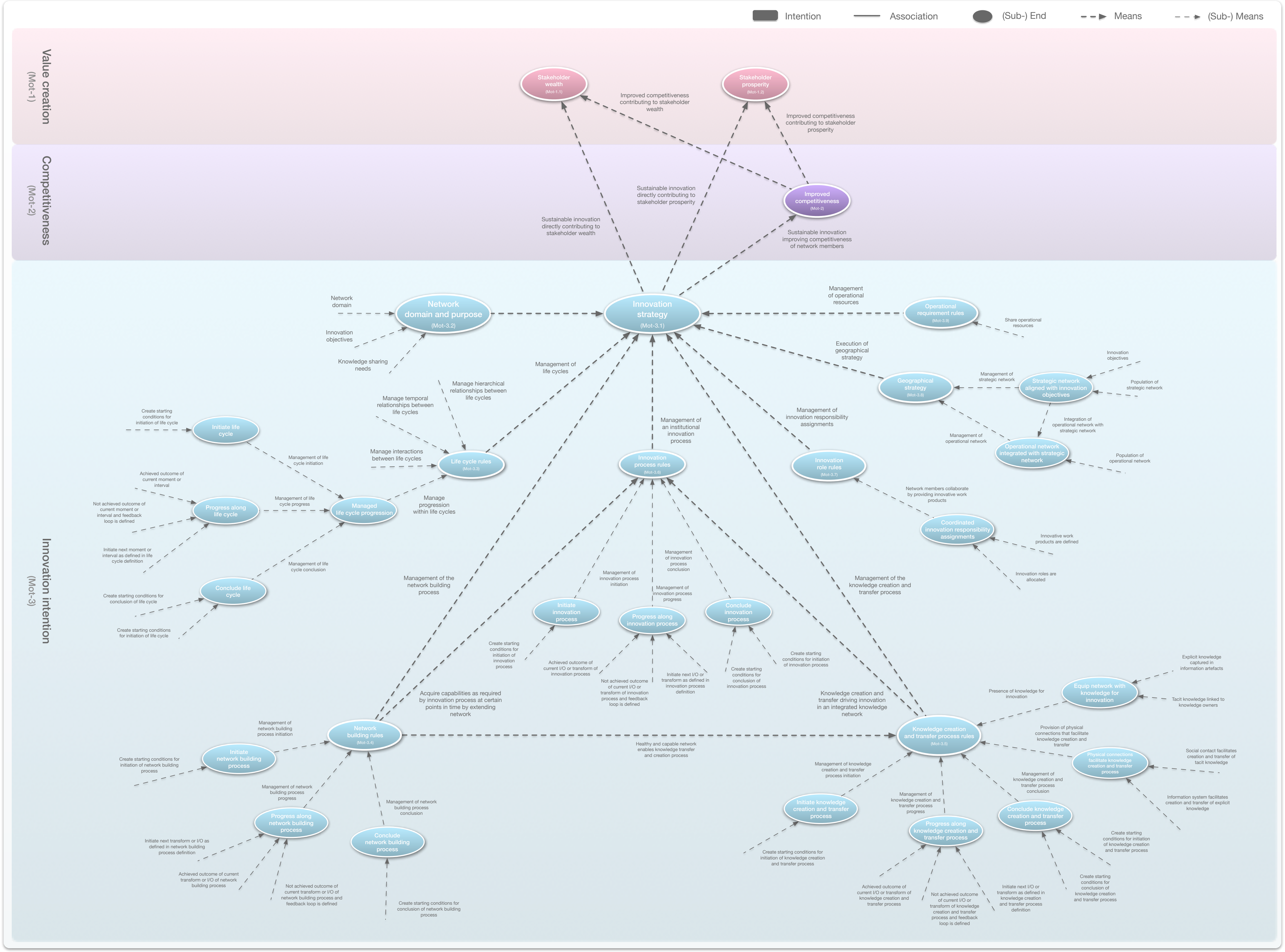


Figure 8.42: Extended motivation definition model (C6, R2.3)

8.5.9 Reference architecture for IKNs

The primitive reference models that populate the reference architecture for IKNs were presented in sections 8.5.3 to 8.5.8. Figure 8.43 provides an overview of the artefact by showing how the reference models populate the top two rows of artefact skeleton at various levels of detail to constitute the top two rows of the reference architecture for IKNs.

The artefact further includes directives for the development of particular models with system (third row) and technology (fourth row) perspectives levels based on previously developed particular business architecture descriptions. By implementing the Zachman framework, the reference architecture for IKNs specifies that the primitive business definition models (second row) of particular architecture descriptions be transformed into primitive system representation models (third row). These system representation models are in turn transformed into primitive technology specification models (fourth row).

The reference models that populate the top two rows of the reference architecture for IKNs, as well as the directives that extend to the third and fourth rows, are free from implementation bias towards any particular IKN instance. These reference models and directives are aimed at stakeholders that are involved in the design, refinement and phase-out phases of the IKN life cycle. The reference models furthermore provide inputs for, or assess, a current (as-is) environment, and provide inputs for a future (to-be) environment.

The reference models and directives included in the reference architecture for IKNs therefore adhere to the artefact behaviour specification.

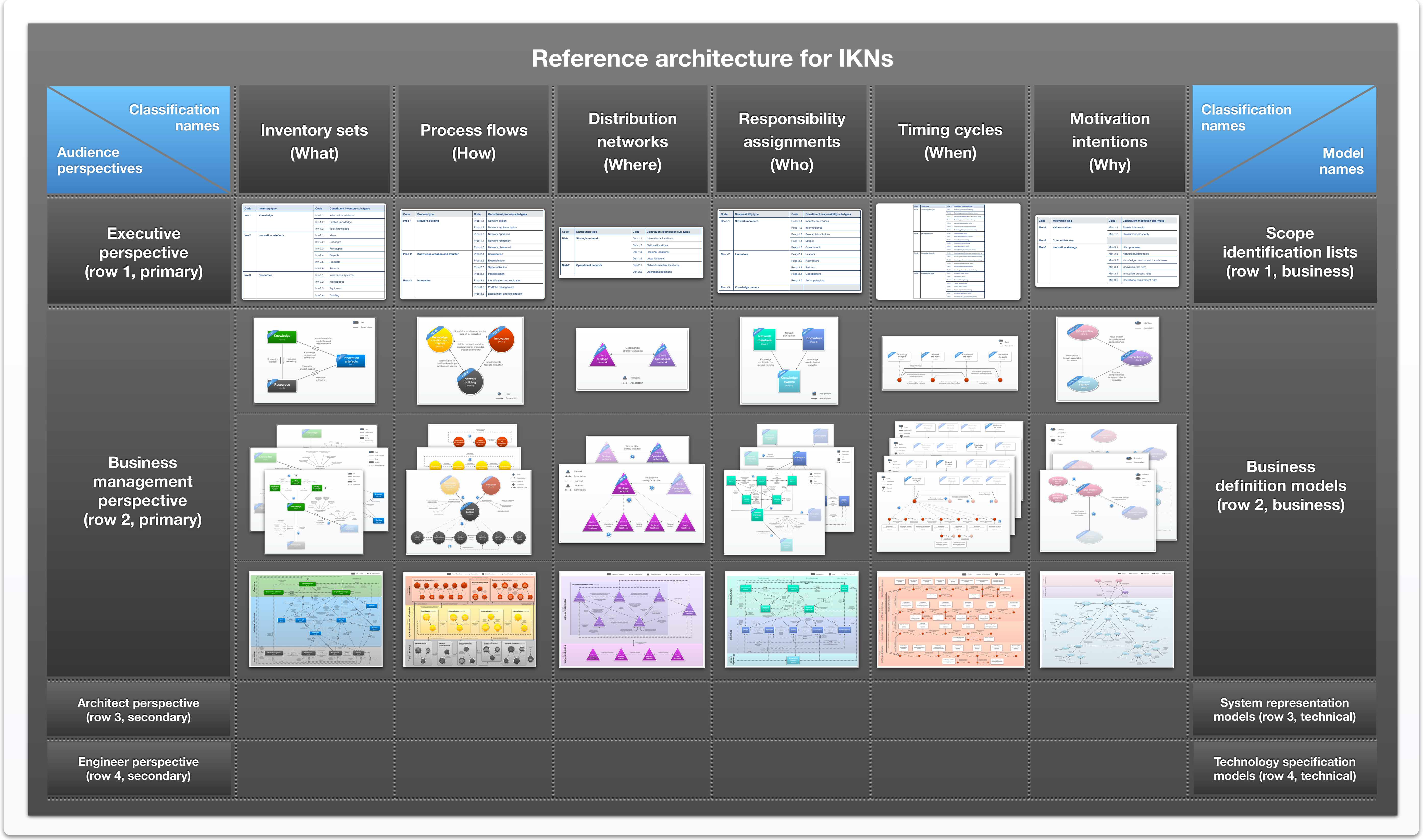
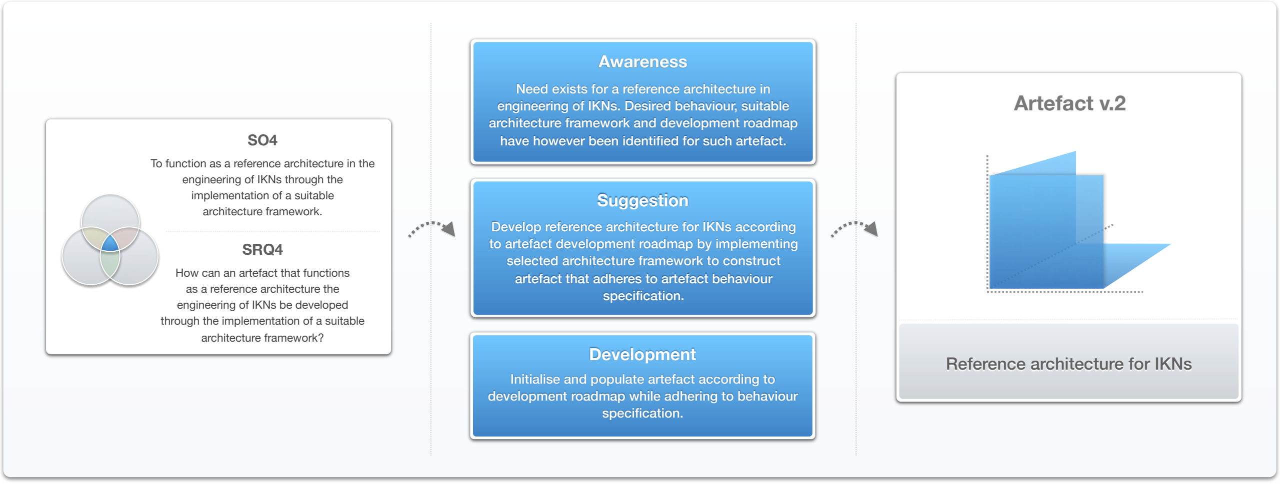


Figure 8.43: Reference architecture for IKNs

8.6 Conclusion

This chapter presented the fourth cycle in the incremental design and development process of the reference architecture for IKNs. The cycle considered the compound requirement perspective that combines the technical and functional requirements, as well as scenarios (refer to blue intersection in Figure 8.44).



SO4

To function as a reference architecture in the engineering of IKNs through the implementation of a suitable architecture framework.

SRQ4

How can an artefact that functions as a reference architecture the engineering of IKNs be developed through the implementation of a suitable architecture framework?

Awareness

Need exists for a reference architecture in engineering of IKNs. Desired behaviour, suitable architecture framework and development roadmap have however been identified for such artefact.

Suggestion

Develop reference architecture for IKNs according to artefact development roadmap by implementing selected architecture framework to construct artefact that adheres to artefact behaviour specification.

Development

Initialise and populate artefact according to development roadmap while adhering to behaviour specification.

Artefact v.2

Reference architecture for IKNs

Figure 8.44: Conclusion of fourth design cycle

The design cycle addressed the fourth solution objective (SO4), and the search process in this chapter was directed by the fourth secondary research question (SRQ4). This question asked *how an artefact that functions as a reference architecture in the engineering of IKNs can be developed through the implementation of a suitable architecture framework*.

The *awareness* of the problem stated that a need exists for a reference architecture in the engineering of IKNs. The desired behaviour of such an artefact (refer to chapter 5), a suitable architecture framework to be employed (refer to chapter 6) and a roadmap for its development (refer to chapter 7) had however previously been identified.

The *suggestion* was subsequently made to develop the reference architecture for IKNs according to the artefact development roadmap by implementing the selected architecture framework to construct an artefact that adheres to the behaviour specification.

The *development* of the reference architecture for IKNs was therefore done through initialising and populating the artefact according to the development roadmap, while adhering to the behaviour specification.

According to the development roadmap, the construction of the reference architecture for IKNs was divided into two phases, namely an *initialisation* phase in which the artefact skeleton was constructed, and a *population* phase during which this skeleton was populated with content. In both of these phases, the artefact was constructed in adherence to the artefact behaviour specification.

In the initialisation phase, the artefact skeleton was specified to function as *reference architecture* containing *primitive reference models*. The primary sources for these generalised reference models were thereafter presented, along with the method of analysis to obtain primitive data from these composite data sources. This method is discussed in detail in Appendix B.

The artefact skeleton was subsequently set up to view IKNs as *enterprises*, by including all six abstractions (columns) provided by the Zachman framework. Primary and secondary *audiences* were specified for the artefact, along with a delineation of the artefact population to the top two rows according to the *type* of information included. The primitive reference models were specified to be *transformation-agnostic,* therefore serving as the basis for the development of versioned particular models that describe either *current* or *future* transformations.

In the artefact population phase of the development roadmap, the artefact skeleton was populated with primitive (single-variable) reference models for all six abstractions included in the artefact scope. These reference models are based on the interpretation of the identification and definition data obtained from the qualitative systematic review (refer to section 8.5.2.1). The models feature varying levels of detail, with appropriate representation formats or meta-models for each abstraction.

In the *inventory* abstraction *knowledge, innovation artefacts* and *resources* were identified as key inventory types in the engineering of IKNs, and were defined as inventory sets in terms of entities and relationships.

In the *process* abstraction *network building, knowledge creation and transfer* and *innovation* were identified as key process types in the engineering of IKNs, and were defined as process flows in terms of transforms and inputs/outputs.

In the *distribution* abstraction *strategic network* and *operational network* were identified as key distribution types in the engineering of IKNs, and were defined as distribution networks in terms of locations and connections.

In the *responsibility* abstraction *network stakeholders, innovators* and *knowledge owners* were identified as key responsibility types in the engineering of IKNs, and were defined as responsibility assignments in terms of roles and work products.

In the *timing* abstraction the *technology life cycle, network life cycle*, *knowledge life cycle* and *innovation life cycle* were identified as key timing types in the engineering of IKNs, and were defined as timing cycles in terms of moments and intervals.

In the *motivation* abstraction *value creation, competitiveness* and *innovation intention* were identified as key motivation types in the engineering of IKNs, and were defined as motivation intentions in terms of ends and means.

In employing the artefact development roadmap, architecture framework selection and artefact behaviour specification, the construction of the reference architecture for IKNs represented the simultaneous investigation of the *relationships between the views, meta-relationships and IKN life cycle axes of the modelling framework*. This is illustrated with the blue planes of the modelling framework to the right of Figure 8.44.

The cycle showed how an artefact that functions as a reference architecture in the engineering of IKNs could be developed through the implementation of a suitable architecture framework. The construction of the reference architecture for IKNs therefore answers SQ4 and achieves SO4, and circumscribes the activities of the fourth design cycle.

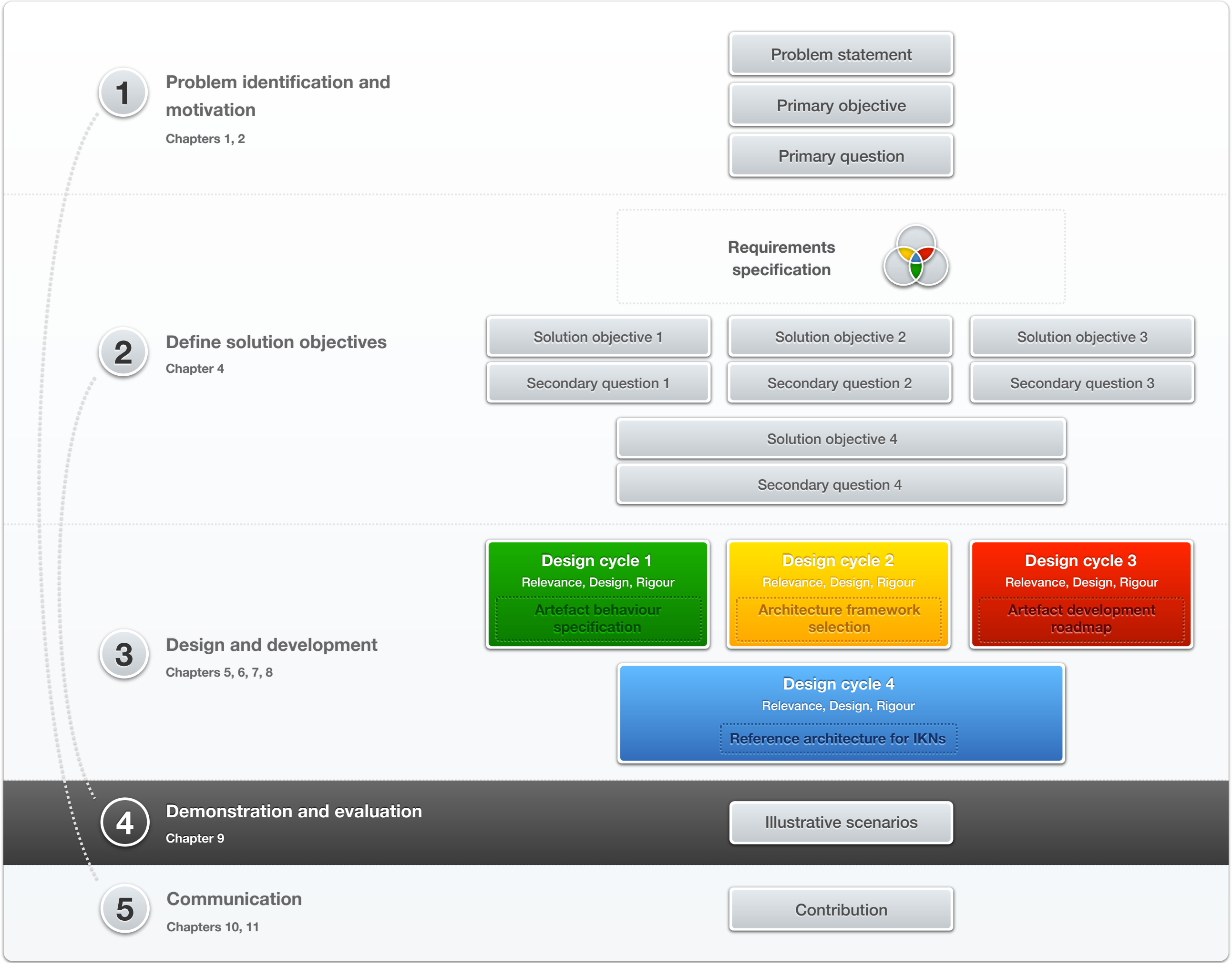
The reference architecture for IKNs has now been designed and developed by achieving and answering, respectively, SO1 and SRQ1 (refer to chapter 5), SO2 and SRQ2 (refer to chapter 6), SO3 and SRQ3 (refer to chapter 7), as well as SO4 and SRQ4 (in this chapter). The reference architecture for IKNs therefore answers the primary research question (PRQ), achieves the research objective (RO), and concludes the design and development phase of the research design employed in the study. This leads to the evaluation of the artefact in Part 4 of the document.

Part 4 – Demonstration and evaluation

This fourth part of the document contains a single chapter (chapter 9) and presents the fourth activity of the research design employed in this study. This activity demonstrates and evaluates the artefact that was designed and developed in the third activity of the research design.

The demonstration element of this activity shows how the artefact aims to achieve the solution objectives identified for it in Part 2, as it is implemented in a number of illustrative case studies. The evaluation element then reflects on the performance of the artefact and the extent to which it does in fact achieve these solution objectives.

Unsatisfactory artefact performance initiates a new iteration of the design and development activity (refer to Part 3), while satisfactory performance will be communicated in Part 5.



9. Artefact demonstration and evaluation

9.1 Introduction

This chapter presents the fourth activity in the research design employed in this study and demonstrates and evaluates the artefact that was designed and developed in the third activity of the research design (refer to Part 3). This activity is in keeping with the axiological view of pragmatism, which states that all claims to value have to be tested and proven in practice and that value is only determined by usefulness (refer to section 3.5.2).

In DSR, the overarching term “evaluation” is concerned with examining DSR outputs, including design artefacts (March & Smith 1995) and design theories (Walls et al. 1992; Gregor et al. 2007). Evaluation provides evidence that a new design artefact or theory developed through DSR achieves the purpose for which it was designed, thereby putting the “science” in “design science” (Venable et al. 2012). Venable et al. (2012) identify the primary purpose of evaluation in DSR to be to “evaluate an instantiation of a designed artefact to establish its utility and efficacy (or lack thereof) for achieving its stated purpose”. They furthermore remark that, given this purpose, rigorous evaluation in DSR spans both the “demonstration” and “evaluation” activities of the DSRM (refer to section 3.3.1.4) contributed by Peffers et al. (2008). The evaluation design employed in this study therefore includes both “demonstration” and “evaluation”, as discussed in this chapter..

The demonstration phase of this activity of the research design shows how the artefact aims to achieve the solution objectives identified in Part 2 of the document, as evident from a number of illustrative scenarios. The evaluation phase then reflects on the performance of the artefact and the extent to which it does in fact achieve these solution objectives. Unsatisfactory artefact performance initiates a new iteration of the design and development activity (refer to Part 3), while satisfactory performance will be communicated in Part 5 of the document.

The approach to the demonstration and evaluation of the artefact is presented in sections 9.2 and 9.3, after which the three illustrative scenarios are introduced in section 9.4. The artefact is demonstrated and evaluated in sections 9.5, 9.6, 9.7 and 9.8 before section 9.9 concludes the chapter.

9.2 Demonstration objective and technique

The objective of the artefact demonstration is to show how the artefact is used to generate solutions to the problem that inspired the original design (Peffers et al. 2008). It is important that this demonstration was done in a way that allows for meaningful evaluation afterwards, i.e. the comparison of the demonstration results with the objectives for a solution. The focus in the demonstration of the reference architecture for

IKNs is therefore on indicating how the artefact attempts to achieve its solution objectives as developed in chapter 4 (refer to Table 9.1), and designed and developed for in chapters 5, 6, 7 and 8. Given the socio-technical nature of the artefact, as well as the constituent design scenario, functional and technical design requirements, *illustrative scenarios* are selected as the demonstration technique for the artefact (refer to section 3.5.5.2). Illustrative scenarios demonstrate the utility of the artefact in environments that resemble the real-world environment it was designed for.

Table 9.1: Solution objectives for the reference architecture for IKNs

Code	Solution objective
SO1	To exhibit the desired behaviour of a reference architecture in the engineering of IKNs.
SO2	To employ an architecture framework that is suitable to the engineering of IKNs.
SO3	To implement the selected architecture framework in a way that enables the artefact to function as a reference architecture.
SO4	To function as a reference architecture in the engineering of IKNs through the implementation of a suitable architecture framework.

SO4, in particular, shows that the demonstration technique is required to demonstrate how the artefact functions as a reference architecture in the engineering of IKNs, through the implementation of a suitable architecture framework. Given this compound objective, the approach to the illustrative scenarios is to implement the reference architecture for IKNs as it would be used in practice, as a reference architecture that is consulted as an input during the engineering of an IKN.

Representatives from the organisations involved in the illustrative network scenarios were engaged in workshops in which the aim was to describe the scenario network’s architecture (refer to Figure 9.1). Using the reference architecture for IKNs as an input, focus was placed on identifying the *scope contexts* of the scenario networks, i.e. the particular first-row models or lists, during the workshops. In keeping with the architecture framework employed by the reference architecture for IKNs, these particular models were developed by considering six basic interrogatives at a strategic level for each of the networks, i.e. “What?”, “How?”, “Where?”, “When?”, “Who?” and “Why?”.

The researcher, performing the role that would in practice be filled by an architect, then defined the identified types as *business concepts*, i.e. developed the particular second-row models. The particular first-row lists, additional definition data collected from discussions in the workshops, and the reference architecture for IKNs were used as inputs for the development of these particular second-row definition models. The particular first-row and second-row models form the architecture description of the scenario networks, and could be transformed into particular models for the third, fourth and fifth rows for each

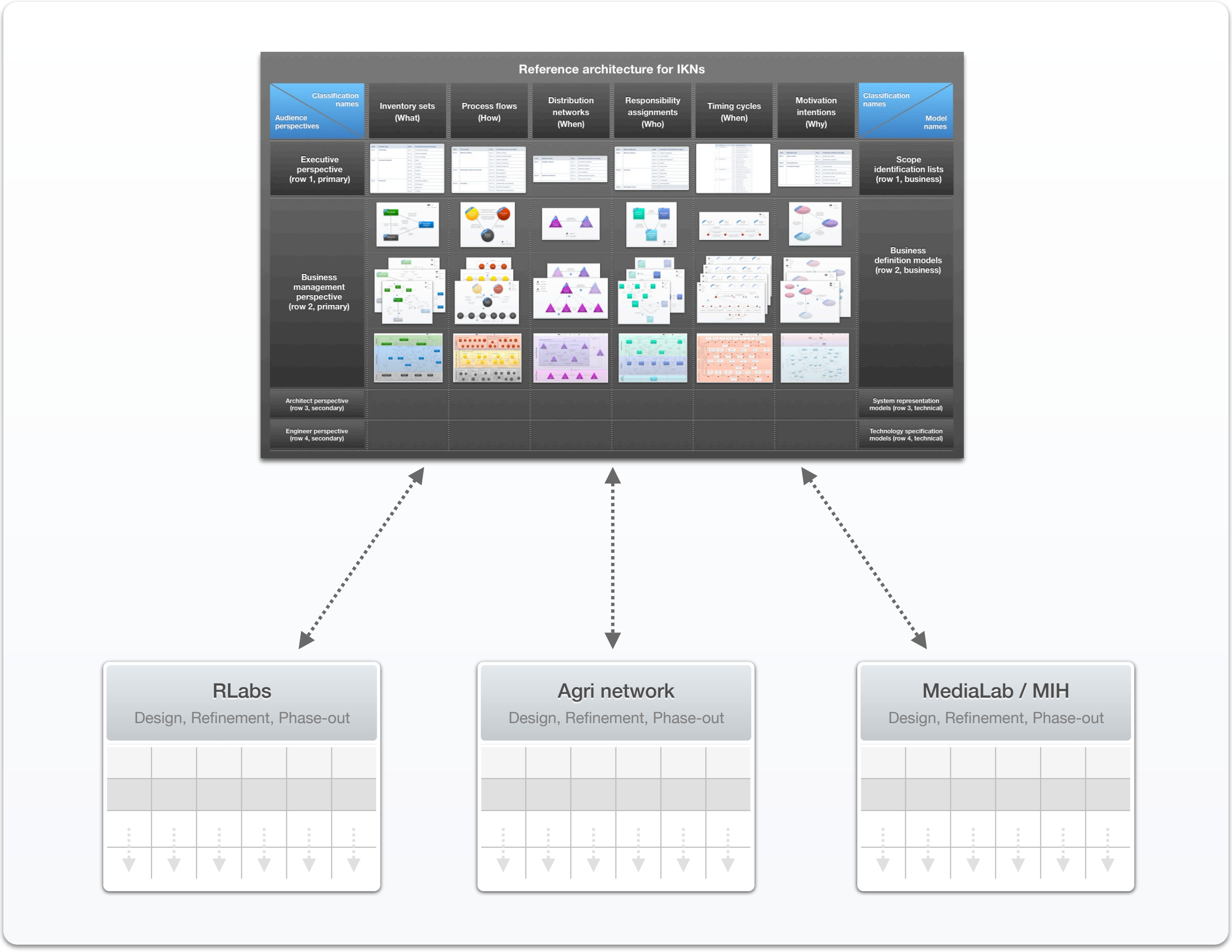


Figure 9.1: Illustrative scenarios

scenario network. The reference architecture for IKNs, however, does not provide reference models for these rows. The developed architecture descriptions were presented and discussed with the network representatives to verify the integrity of the collected demonstration data for evaluation. This is regarded as an accurate reflection of the problem scenario that the artefact was designed for and therefore an appropriate technique to demonstrate its ability to generate solutions.

In documenting the demonstration of how the artefact attempts to meet each of its solution objectives, checklists are employed to structure the presentation of results. Each item to be demonstrated in such a list is accompanied by the relevant requirement that has to be met to achieve an element of the solution objective. These checklists ensure that sufficient and pertinent perspectives on the artefact performance relating to the particular solution objective are available in order to enable meaningful evaluation.

The fourth solution objective (SO4), which considers the artefact from a holistic point of view, is demonstrated first (refer to section 9.5), since the evidence from this demonstration is relevant to other solution objectives as well.

9.3 Evaluation approach

The artefact is evaluated by comparing the objectives of a solution to the observed results from use of the artefact in the demonstration (Hevner & Chatterjee 2010a). The objectives for a solution to the problem considered in this study are reviewed in Table 9.1.

The demonstration results are therefore evaluated to determine the following (note that SO4 is listed first below, since it is demonstrated first):

- *The success of the artefact* in functioning as a reference architecture in the engineering of IKNs through the implementation of a suitable architecture framework (SO4).
- *The success of the artefact* in exhibiting the desired behaviour of a reference architecture in the engineering of IKNs (SO1).
- *The success of the artefact* in employing an architecture framework that is suitable to the engineering of IKNs (SO2).
- *The success of the artefact* in implementing the selected architecture framework that enables it to function as a reference architecture (SO3).

With the readability of this chapter in mind, these comparisons are performed directly after each of the sections in which an element of the demonstration of the artefact is presented. The checklists that structure the demonstration of the artefact therefore also guide the evaluation process.

9.4 Illustrative scenarios

The cooperation of a number of IKNs operating in the Western Cape province of South Africa was enlisted to serve as illustrative scenarios (refer to section 3.5.5.2) in the illustration of the utility of the reference architectiure for IKNs. The eligibility of these networks as potential illustrative scenarios was determined through various definitive criteria, including:

- A network of organisations that collaborate within the knowledge supply chain,
- A focus on fostering sustainable innovation that promotes the competitiveness of network stakeholders,
- Willingness to provide access to architectural information.
- Willingness to commit time from employees with a strategic and business focus, to the evaluation process.

An overview of each of these illustrative scenarios is now presented, highlighting the way in which each scenario network executes the collaborative innovation process.

9.4.1 RLabs

The Reconstructed Living Lab (RLabs) is a global movement and registered social enterprise founded in 2008 as an environment for community-driven innovation and reconstruction. RLabs' main focus is social development and it creates an environment in which people are empowered to make a difference in the lives of others. RLabs has its headquarters in Athlone, Cape Town, but is also active in the United Kingdom, Europe, Asia, South America and Africa (RLabs 2014). The Athlone community is faced with a range of social issues, including unemployment, gangsterism, narcotics and ineffective education.

The RLabs network consists of the RLabs organisation itself, community members, as well as several partners that share its innovative vision and mission. They include role players from the South African technology sector (e.g. Mxit and Afrihost), research institutions both local and abroad (e.g. London School of Economics and the Amsterdam Business School), as well as a range of social enterprises (RLabs 2014) and funders (e.g. USAID and The Bertha Foundation). RLabs is also considered to be a living lab and is a member of the Living Labs of Southern Africa network. Living labs are organisations that, along with their other activities, provide training, provide the framework and provide resources to start-ups and existing enterprises so these can become more innovative with the living labs' help. These living labs often have access to specialist skills like start-up mentoring, product prototyping and network of partners that start-ups can approach under their protection (RLabs 2014).

The collaborative innovation process as executed by the network is managed as follows:

RLabs runs a free academy that offers a wide range of courses, including leadership, entrepreneurship, social and new media, and technology development to members of the surrounding community. Students from the academy often go on to become trainers and facilitators of courses themselves. As a living lab, RLabs' activities also extend to their innovation incubator that provides community members (often graduates of their academy programmes) with the necessary resources to explore and develop their ideas. This includes workspace, funding, mentorship and, most importantly, access to RLabs' outside partners from both the South African and international public and private sectors. Between five and ten start-ups are incubated every year – all kick-started by the knowledge exchange within the RLabs network.

9.4.2 Agri network

The Agricultural network exists between Stellenbosch University, Winetech, Hortgro, SATI, agricultural producers in the wine, table grapes and fruit industries, as well as the South African government, albeit it is not an active network member. It is a stable network that has existed in some form or other since roughly 1985.

Wine producers in South Africa often face similar challenges in terms of pests, production methods and market access, to name but a few. They do, however, not have the individual or combined capacities to perform meaningful research to address these issues and research and development capacity must therefore be obtained from elsewhere. For this reason there exists a statutory levy system in South Africa in which all wine producers need to pay a levy to government, a large part of which is dedicated to research that will benefit the entire industry. This means that autonomous organisations that are often competitors are collaborating by funding research to improve their collective and individual competitiveness.

Winetech is an organisation that was set up with the mandate to manage this research funding on behalf of the wine industry and is also charged with knowledge and technology transfer back to the wine producers. A similar configuration exists in the table grapes and fruit growers industry, in which the South African Table Grapes Industry (SATI) and Hortgro are the respective funding managers.

South African universities perform the bulk of the research and receive funding from Winetech, SATI and Hortgro. Stellenbosch University is a major player in this research given its research capacity in the relevant areas, as well as its close proximity to Winetech, SATI and Hortgro. It is within 30 minutes' drive from all three.

Occasionally, e.g. with climate change, the situation gets even more involved as the same challenges impact all three of these sectors, and in such cases funding Winetech, SATI and Hortgro is combined to collaboratively fund research. Winetech, SATI and Hortgro also combine their resources to apply for funding from the South African government.

The collaborative innovation process as executed by the network is managed as follows:

Agricultural producers from the three sectors pay their mandatory levies to the South African government, from where it is passed on to the research management organisations, being Winetech, SATI and Hortgro. These research managers then each follow their own processes to collect knowledge regarding industry issues that should be addressed and transform these into priorities that accompany calls for proposals to research institutions. Research institutions then prepare their proposals and submit them to the respective research management organisations where technical committees including knowledgeable individuals from the producers evaluate and approve proposals as they see fit.

The research is then performed as proposed by the research institutions. In most cases, the innovative output generated is knowledge-based, e.g. in the form of new methods and techniques, and this knowledge is transferred back to the research managers via project reports. Winetech, SATI and Hortgro then disseminate this knowledge to the producers through informational events and online as well as physical publications.

In cases where the innovative output from the research process is something that can be patented or developed into a commercial product or service, the technology transfer office of the relevant research institution will become a more active network member. These technology transfer offices, e.g. Stellenbosch

University's InnovUS, have the know-how to assist in the filing of patents and the decision-making related to licensing to an existing industry enterprise that will further develop the product, or the founding and incubation of a start-up enterprise.

9.4.3 MediaLab/MIH network

The MediaLab/MIH network exists between the MediaLab research group at Stellenbosch University in South Africa and the MIH Group, a holding company of NASPERS, a major South African media organisation. MIH has major investments in media companies around the world, including DSTV in South Africa, ibibo in India, OLX in South America and Tencent in China (MIH 2014).

MIH, on the one hand, has a need for applied new media technology research to bolster its existing ventures, as well as access to new talent in the South African student market. Researchers and students at Stellenbosch University on the other hand would like to perform interdisciplinary applied research related to new media technologies, but this requires research guidance and funding from an industry partner. For this reason there exists a natural symbiosis between the two organisations and this lead to a formal networking agreement.

The collaborative innovation process as executed by the network is managed as follows:

Senior researchers at the MediaLab and representatives from MIH develop research themes and priorities collaboratively. Students wishing to start work on a master's or doctoral study then submit research proposals in line with these themes and priorities for approval by a committee comprised of representatives from both organisations. Approved projects are executed and funded by MIH, and on the academic side this leads to the conferring of the relevant degree on the student involved. In most cases, the innovative output generated is knowledge-based in the form of an exploratory study, e.g. a theoretical proof of concept, and this knowledge is transferred back to the MIH via project reports to which they have full access. Further knowledge transfer is also possible in cases where the student goes on to work for one of MIH's subsidiaries.

However, in cases in which the innovative output from the research process is something that can be prototyped, patented and developed into a commercial product or service, InnovUS will become a more active network member. Following on this decisions are made about whether the product or service development will occur within one of MIH's existing organisations or whether a start-up enterprise would be the best way forward. In the case of a start-up, an MIH-managed incubator environment used to exist in Stellenbosch, but is not active anymore.

9.5 Reference architecture for IKNs

This section presents the demonstration and evaluation of how the artefact attempts to achieve the blue solution objective that was identified in section 4.4.4 and designed for in chapter 8. The checklist for this particular solution objective is discussed, after which each of its items are demonstrated and evaluated.

The solution objective in focus in this section of the demonstration and evaluation of the artefact is as follows:

Solution objective: To function as a reference architecture in the engineering of IKNs through the implementation of a suitable architecture framework. (SO4)

The aim is therefore to show how the artefact attempts to simultaneously address its technical and functional requirements, as well as the scenarios in which it is expected to operate (refer to chapter 8), and to judge the success with which this is achieved.

The checklist used to demonstrate and evaluate the artefact's success in achieving the above solution objective is presented in Table 9.2. The content of this list is derived from the solution objective (SO4) itself, as well as knowledge of the selected architecture framework. The list contains a number of features of the artefact to be demonstrated, along with the related requirement that has to be met to achieve an element of the larger solution objective stated above.

Table 9.2: Checklist for demonstration and evaluation of reference architecture for IKNs

Check	Requirement
1. Primitives	Provide inputs for the creation and assessment of primitive particular models.
2. Composites	Provide inputs for the creation and assessment of composite particular models.

This checklist focuses on the success with which the artefact functions as a reference architecture in the engineering of IKNs through the implementation of an architecture framework. The intention is therefore to demonstrate how the artefact attempts to achieve its holistic solution objective (SO4).

In this section, as well as in sections 9.6, 9.7 and 9.8, the evidence from the demonstration is presented in as objective a fashion as possible. The interpretation of evidence regarding the performance of the reference architecture for IKNs is therefore limited to sections that focus on evaluation.

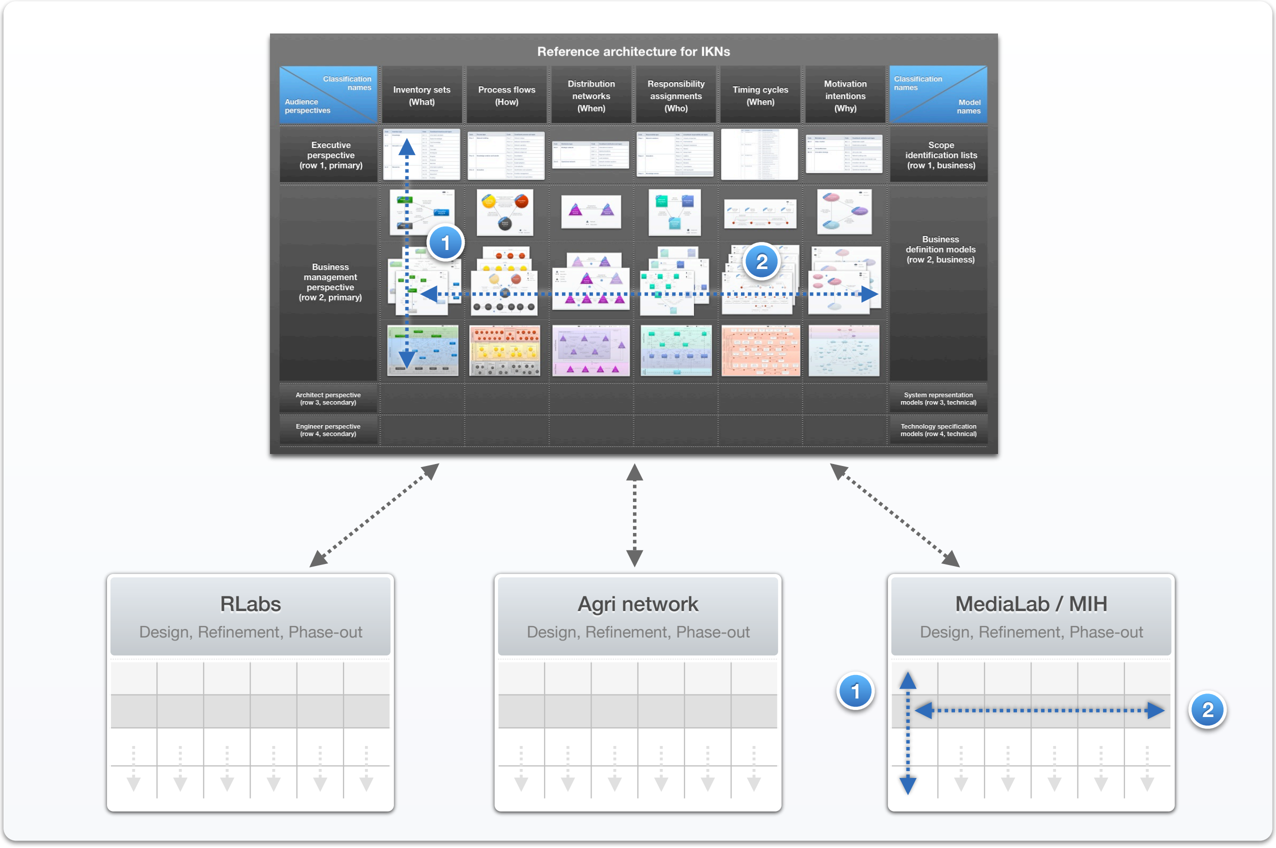


Figure 9.2: Demonstration and evaluation of reference architecture for IKNs

9.5.1 Primitives

The reference architecture for IKNs is required to provide inputs for the creation of primitive particular models that describe individual instances of networks (refer to Figure 9.2) (1).

9.5.1.1 Demonstration

*This section **presents** evidence of the ability of the reference architecture for IKNs to provide inputs for the creation and assessment of primitive particular models.*

The ability of the reference architecture for IKNs to provide inputs for the creation and assessment of primitive particular models was demonstrated in all three illustrative scenarios (refer to section 9.4). The top-row models of the architecture description of each scenario network were developed during architecture workshops conducted with representatives from the organisations involved in each network. These network representatives were assumed to be the owners of the architectural information required to develop the models. During each of the workshops the scope identification lists (top row) of a scenario network were developed through considering the network in terms of the six abstractions provided by the reference architecture for IKNs. The first-row models of the MediaLab/MIH scenario network are presented in Table 9.3.

Table 9.3: MediaLab/MIH network – Scope identification models

MediaLab/MIH Row 1	**Inventory sets** (Inventory identification model; C1, R1)	**Process flows** (Process identification model; C2, R1)	**Distribution networks** (Distribution identification model; C3, R1)
Executive perspective (Scope contexts in scope identification lists)	• **Knowledge** • Information artefacts • Documents • Research outputs • Contracts • Budgets • Websites • Blogs • Proposals • Patents • Explicit knowledge • Tacit knowledge • **Innovation artefacts** • Ideas • Concepts • Prototypes • Projects • Products • Services • **Resources** • Information systems • Hardware • Software • Applications • Internet • Workspaces • Laboratory • Offices • Equipment • Stationery • Amenities • Funding • Bursaries	• **Knowledge creation and transfer** • Socialisation • Externalisation • Combination • Internalisation • **Innovation** • Identification and evaluation • Idea generation • Concept development • Proof of concept • Incubation • Prototyping • Portfolio management • Deployment and exploitation • Patenting • Implementation • Exploitation • **Network building** • Network design • Network implementation • Understand roles • Secure funding • Build trust • Sourcing of knowledge • Operate and refine network • Network phase-out or metamorphosis	• **Strategic network** • International locations • National locations • Regional locations • Local locations • **Operational network** • Network stakeholder locations • Industry enterprise locations • Research institution locations • Intermediary locations • Government locations • Operational locations
MediaLab/MIH Row 1	**Responsibility assignments** (Responsibility identification model; C4, R1)	**Timing cycles** (Timing identification model; C5, R1)	**Motivation intentions** (Motivation identification model; C6, R1)
Executive perspective (Scope contexts in scope identification lists)	• **Network stakeholders** • Research institutions • Academic departments • Students • Academics/supervisors • Lab management • Industry enterprises • Technology organisations • Employers • Start-ups	• **Knowledge life cycle (ref.)** • Weekly demonstrations • Monthly talks • Dissertation writing intervals • Study timelines • Publication timelines • **Innovation life cycle (ref.)** • Patenting moments and intervals	• **Value creation** • Corporate social investment • Wealth creation • Impact on society • Research agenda • **Competitiveness** • Marketing for region, university • Attractiveness for students (lab competitiveness vs.

	• Funders • Intermediaries • Technology transfer office • Incubator • Government • National government • Local government • **Innovators** • Leaders • Networkers • Builders • Coordinators • Anthropologists • **Knowledge owners**	• **Technology life cycle (ref.)** • Technology exploitation moment • **Network life cycle (ref.)** • Network design moments • Network implementation • Network implementation partner selection moments • Marketing moments • Trust building intervals • Agreement moments • Network operation moments and intervals • Student application intervals • Student interview moments • Orientation intervals • Graduation moments • Budgeting cycles • Funding cycles • Network refinement moments • Network refinement partner selection moments • **Operational timing cycles** • Academic years • Workflows	other universities) • Increased research outputs (including students) • Contact with talent • **Innovation intention** • Innovation strategy • Network domain and purpose • Innovation process rules • Idea capture • Innovation role rules • Directives from industry • Knowledge creation and transfer rules • Education • Applied research • Network building rules • Operational requirement rules • Funding • IP legislation • Workspace reflects ethos • Life cycle rules • Geographical strategy

The reference architecture for IKNs was used as a starting point for the development of these particular first-row models. Some elements of the reference models were adopted verbatim, since they accurately described an element of the scenario network. Evidence of this is found in the inventory identification model (C1, R1) in Table 9.3 where the *Innovation artefacts* item was adopted as an inventory type without any changes.

In other cases, however, the reference models were augmented at various levels of detail in order to produce a particular model that more accurately describes the scenario networks. The timing identification model (C5, R1) presented in Table 9.3 features an *Operational timing cycle* element that was added at a low level of detail. Multiple additions and changes, however, were made at higher levels of detail, e.g. the adoption of an entire life cycle from the reference model, with particular moments or intervals being customised. Similar effects are evident in the inventory identification model (C1, R1) where several types of *information artefacts* and *resources* were identified as they pertain to the MediaLab/MIH network. In some particular models, the reference model was adopted, but with one or more elements being omitted. This is evident from the responsibility identification model (C4, R1) in Table 9.3, in which the *market* network member type has been omitted, as no organisations or individuals that represent the *market* participate in the MediaLab/MIH network.

For each scenario network, these first-row scope identification lists were transformed into business definition models (second row) according to the inputs provided by the reference architecture for IKNs. The first-row particular models and the second-row reference models therefore served as inputs for the development of second-row particular models.

Given the level of adoption of the first-row reference models, the lowest detail particular business definition models closely or exactly resembled the respective reference models. This is illustrated in the particular basic inventory definition model (C1, R2.1) presented in Figure 9.3, which is an exact copy of the corresponding reference model presented in Figure 8.8. The same is true for the particular basic responsibility definition model (C4, R2.1) presented in Figure 9.4, which is also an exact copy of the corresponding reference model presented in Figure 8.25.

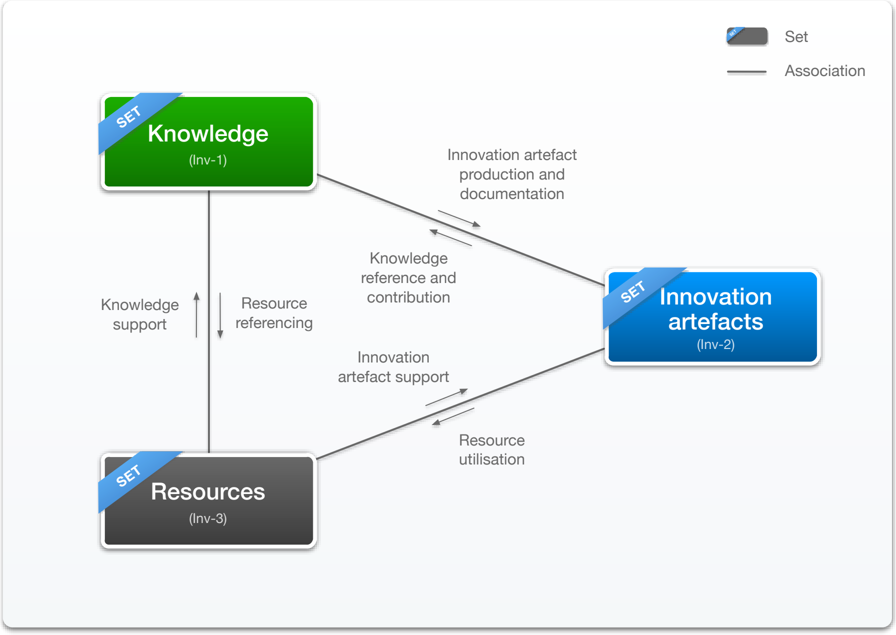


Figure 9.3: Basic inventory definition model (MediaLab/MIH; C1, R2.1)

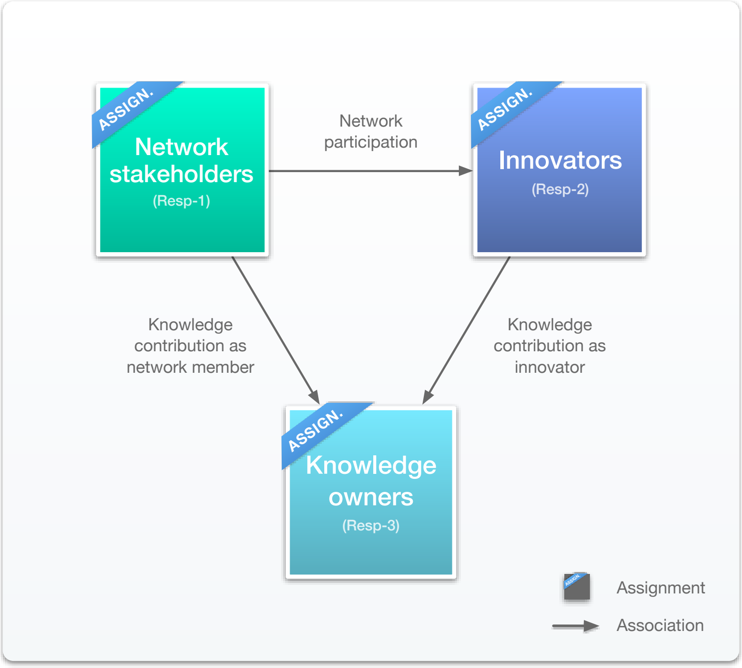


Figure 9.4: Basic responsibility definition model (MediaLab/MIH; C4, R1)

The particular business definition models with a higher level of detail, however, do show differentiation from the corresponding reference models. Figure 9.5 depicts the particular inventory set definition model describing *knowledge*. As in the reference model (refer to Figure 8.9), this model describes *knowledge* as an inventory set by indicating its deconstruction into *information artefacts*, *explicit knowledge* and *tacit knowledge* as inventory entities (1). The relationships between these entities are also indicated (2), but this particular model also includes a further deconstruction of *information artefacts* into various inventory sub-entities according to their definition in the MediaLab/MIH scenario network (3). These sub-entities maintain versions of the relationships that their parent inventory entity shared with other entities. The model also indicates the associations between *knowledge* and *innovation artefacts* (4), as well as between *knowledge* and *resources* (5).

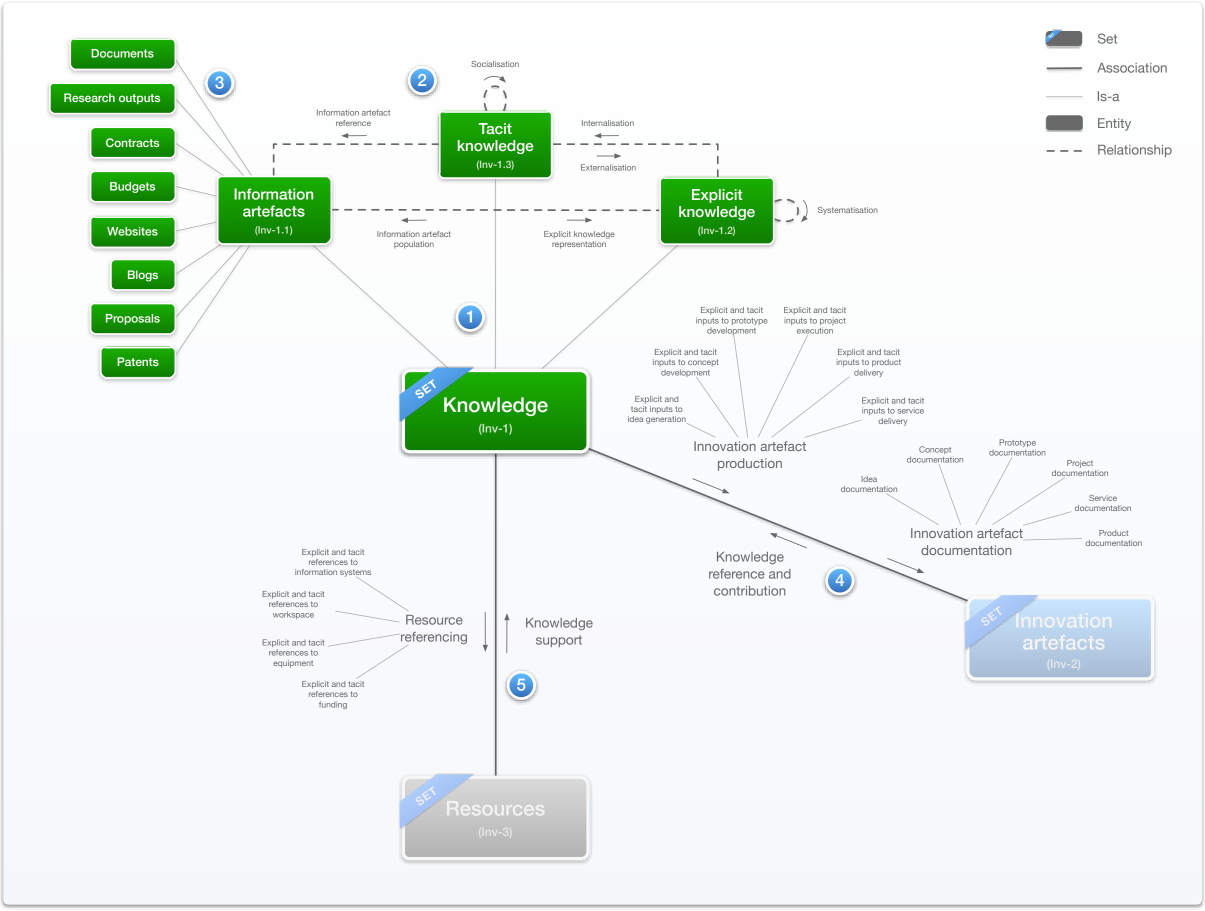


Figure 9.5: Inventory set definition model – Knowledge (MediaLab/MIH; C1, R2.2.1)

These inventory sub-entities form part of the high-detail particular extended inventory definition model (refer to Figure 9.6), which also shows resemblances to the corresponding reference model (refer to Figure 8.12). In this particular model however, the sub-entities are grouped into the parent *information artefacts* entity in order to maintain the readability of the model. Similar deconstructions are also visible in the constituent entities of the *resources* inventory set. An even more detailed particular model showing the deconstruction of all inventory entities, along with their more detailed relationships would be possible, but is not included in this discussion.

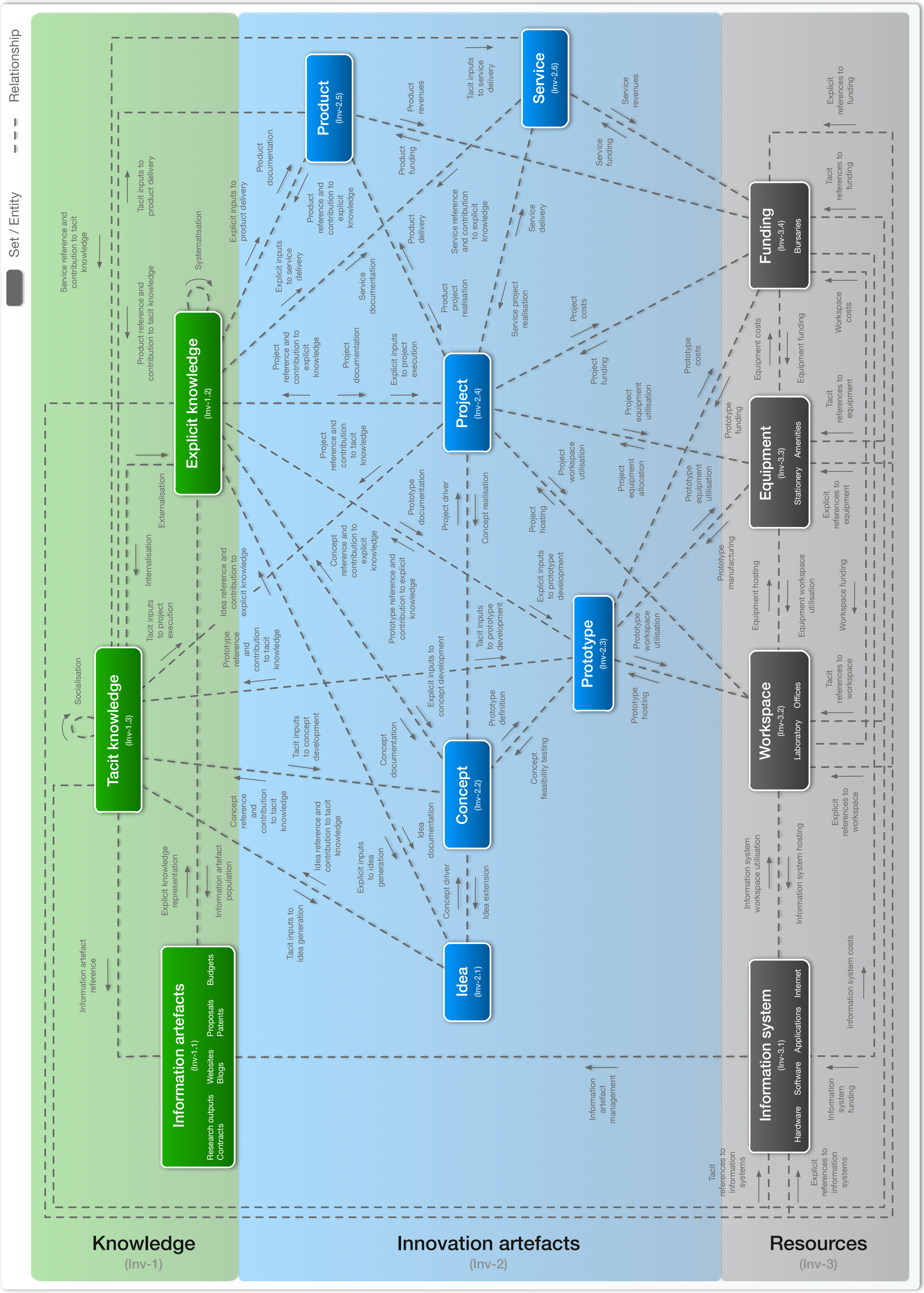


Figure 9.6: Extended inventory definition model (MediaLab/MIH; C1, R2.3)

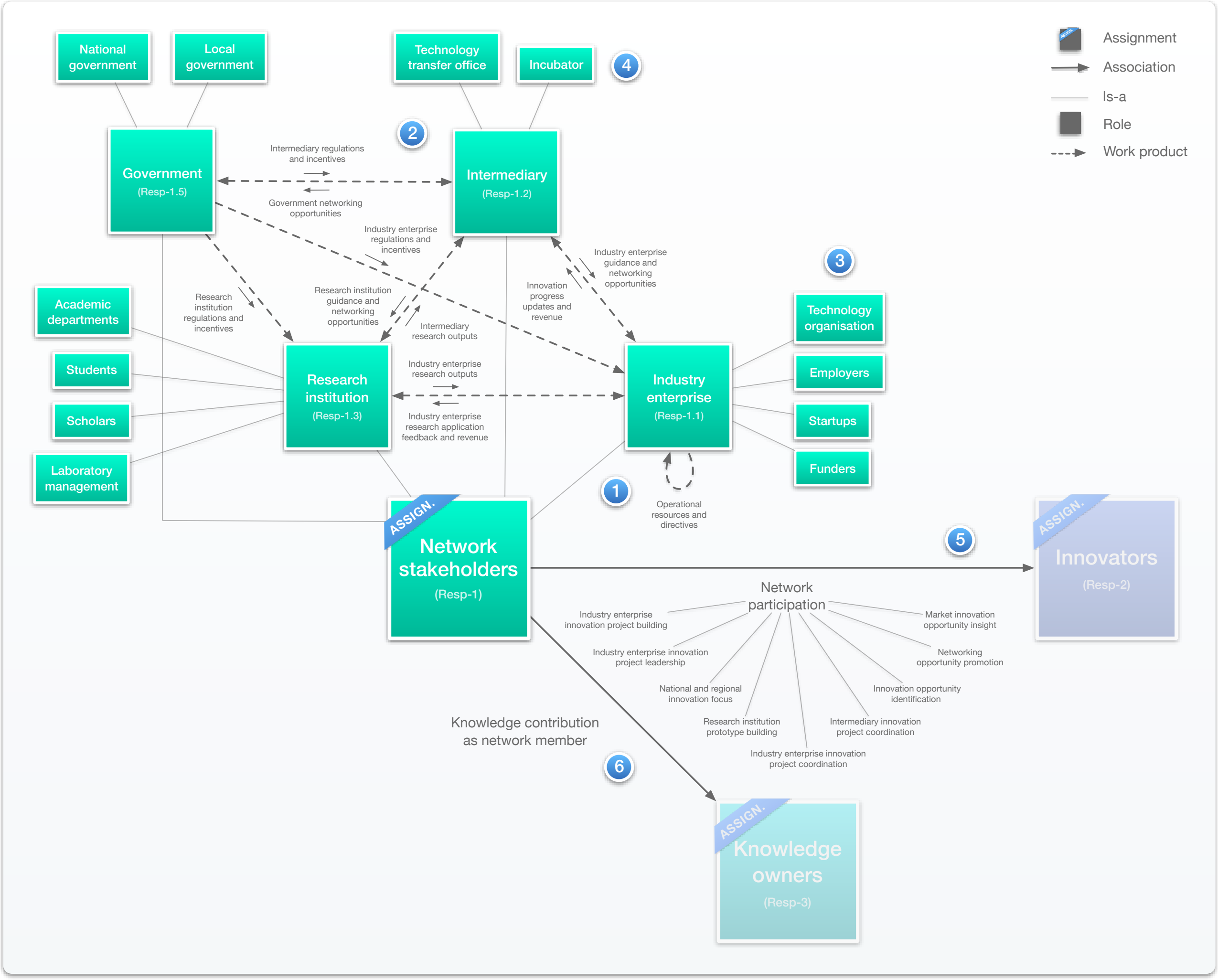


Figure 9.7: Responsibility assignment definition model – Network stakeholders (MediaLab/MIH; C4, R2.2.1)

Similar deconstructions of concepts as in the particular inventory set definition model for *knowledge* are visible in the particular responsibility assignment definition model for *network stakeholders* (C4, R2.2.1; refer to Figure 9.7). This model inherits various elements from the corresponding reference model (refer to Figure 8.26) by deconstructing the responsibility assignment into various roles (1) and indicating the work products they share (2). The particular model, however, omits the *market* role, as no such organisations or individuals participate in the network and were therefore also not identified in the first-row model. The particular model adds various constituent sub-roles (3) as identified in the responsibility identification model (C4, R1; refer to Table 9.3). The detailed sub-work products linking these sub-roles are instances of the work products shared by their parent roles.

This model is furthermore evidence of a future (to-be) model aimed at refining the network, as it adds the *incubator* sub-role that did not exist in the operational network instance at the time the scenario workshops were conducted (4). The current (as-is) primitive model, however, was assessed, and inputs

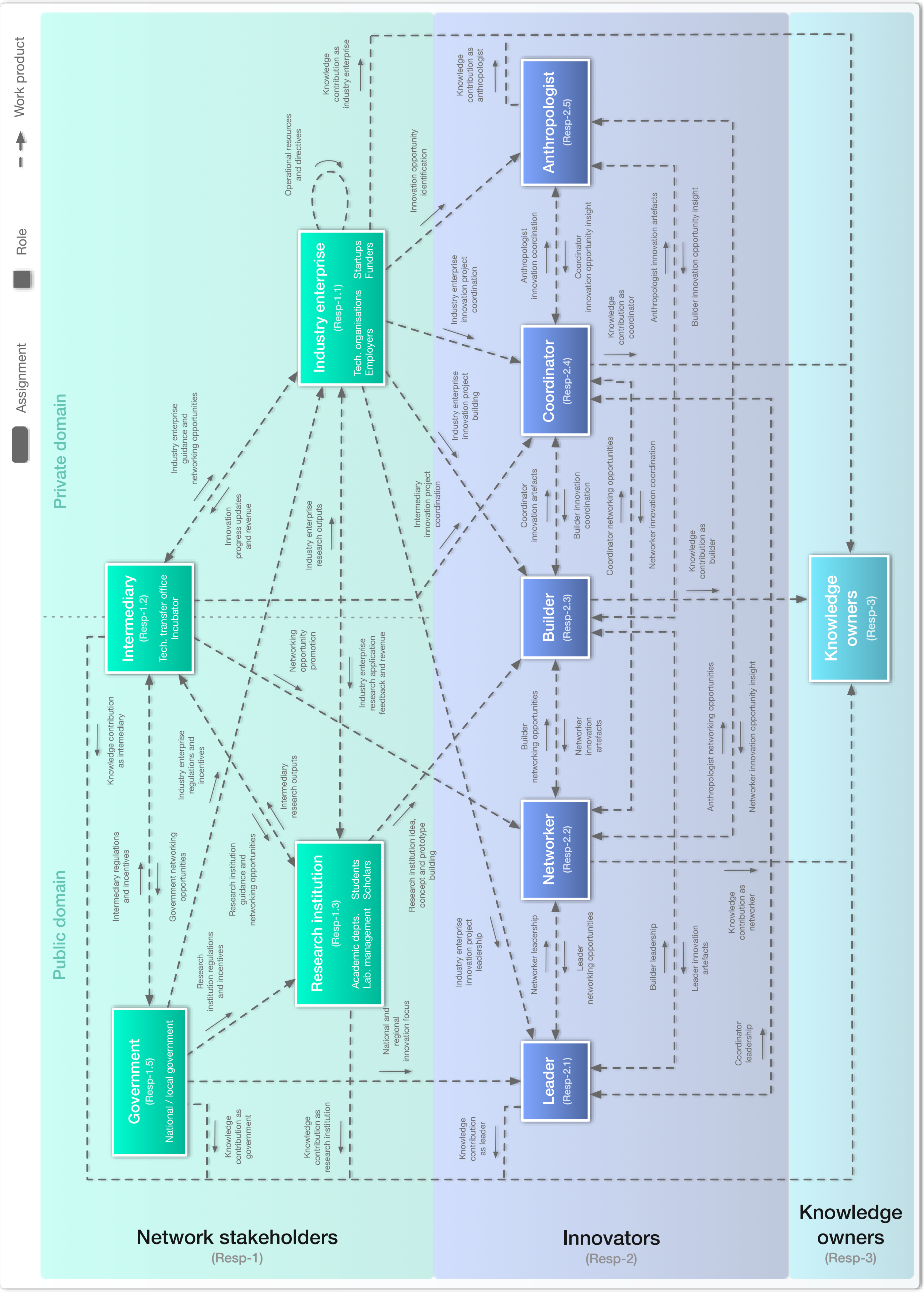


Figure 9.8: Extended responsibility definition model (MediaLab/MIH; C4, R2.3)

from the reference architecture showed that a different type of intermediary organisations might be missing from the operational network. From the work products and associations shared by roles and responsibilities, especially those between *network stakeholders* and *innovators* in the reference model, it was deduced that an *incubator* sub-role was required in the MediaLab/MIH network.

The model inherits the associations between *network stakeholders* and *innovators* (5), as well as between *network stakeholders* and *knowledge owners* (6), from the reference model.

The omission of the *market* role is also visible in the particular extended responsibility definition model (refer to Figure 9.8). As was the case with the particular extended inventory definition model, constituent sub-concepts are grouped into their parent concept (role) to maintain the readability of the model. A more detailed model indicating the deconstruction of all roles into sub-roles, along with their detailed sub-work products, would, however, be feasible. This model is also a future (to-be) model, as it includes the *incubator* sub-role as part of the deconstruction of the *intermediary* role.

9.5.1.2 Evaluation

*This section **interprets** evidence of the ability of the reference architecture for IKNs to provide inputs for the creation and assessment of primitive particular models.*

The first-row reference models included in the reference architecture for IKNs proved to be helpful in guiding the development of both current and future particular scope identification lists in the scenario workshops. These reference models enabled both the initial identification of particular network boundaries, as well as the structuring of identified detailed architectural boundaries into coherent types.

The close similarities between the reference and particular models in both the first and second rows at low levels of detail are understandable, given that they contain generalised architectural information regarding the construction of IKNs. It would also be expected that there is more differentiation between the reference and particular models at higher levels of detail, as the architectural information at these detail levels is more specific to the scenario network. These increases in detail and differentiation are found *in* both rows of reference models, and not *between* rows in which a change in perspective occurs rather than a growth in detail. It is therefore *not the case* that the similarities between reference models and particular models decreases from the first to the second row.

As the levels of detail increased in each cell, it was possible to inherit relevant architectural patterns from the reference models, while also adding and omitting elements as was required in order to accurately describe the scenario networks.

It is therefore possible to motivate that the reference architecture for IKNs succeeded in providing inputs for the creation and assessment of *primitive* particular models describing the construction of various scenario networks. The artefact therefore **functioned as a reference architecture in the engineering of IKNs through the implementation of a suitable architecture framework**.

9.5.2 Composites

The reference architecture for IKNs is required to provide inputs for the creation and assessment of composite particular models that describe individual instances of networks (refer to Figure 9.2) (2).

9.5.2.1 Demonstration

*This section **presents** evidence of the ability of the reference architecture for IKNs to provide inputs for the creation and assessment of composite particular models.*

Figure 9.9 shows a second-row particular composite model that was constructed from the primitive particular models discussed in section 9.5.1. This composite model describes elements of the MediaLab/MIH scenario network from a business perspective, and focuses on the segment of its architecture that includes the collaborative innovation process executed by the network stakeholders .

This is a current (as-is) multi-variable definition model with an intermediate level of detail and contains certain inventory entities, process transforms, distribution locations, responsibility roles, timing moments, motivation ends (refer to legend bottom right Figure 9.9), as well as some of the integrations that link them. These elements and integrations are deemed to be sufficient to describe the architectural information as intended by the model, although further elements and integrations could feasibly be indicated. The primitive links between these elements could also be indicated to add further detail to the model.

In order to generate this model, the particular primitive models developed during the illustrative scenarios, as well as the guidance provided by the reference architecture for IKNs were utilised as inputs. The model indicates the integrations between *innovation process* transforms and moments in the *innovation life cycle* (refer to Figure 9.9) (1). The production of *innovation artefacts* in the *innovation process* is also shown (2), along with some prominent *knowledge*-based inputs to the process (3). *Innovator* roles that are directly linked to the production of *innovation artefacts* are indicated (4), along with the allocation of these *innovator* roles to the various *network stakeholders* through *innovation role rules*. These *network stakeholders* are located at their respective *network stakeholder locations* (5) and are seen as *knowledge owners* (6). Examples of how *innovation process rules* can result in iterations of various process transforms and life cycle moments are also indicated (7).

The accuracy of this composite model as a description of the innovative activities of the scenario network was verified with representatives from the network. This model was assessed through comparison with the primitive reference models, including the integration directives provided by the reference architecture for IKNs. From this benchmarking exercise the need for an incubator role in the network was reiterated.

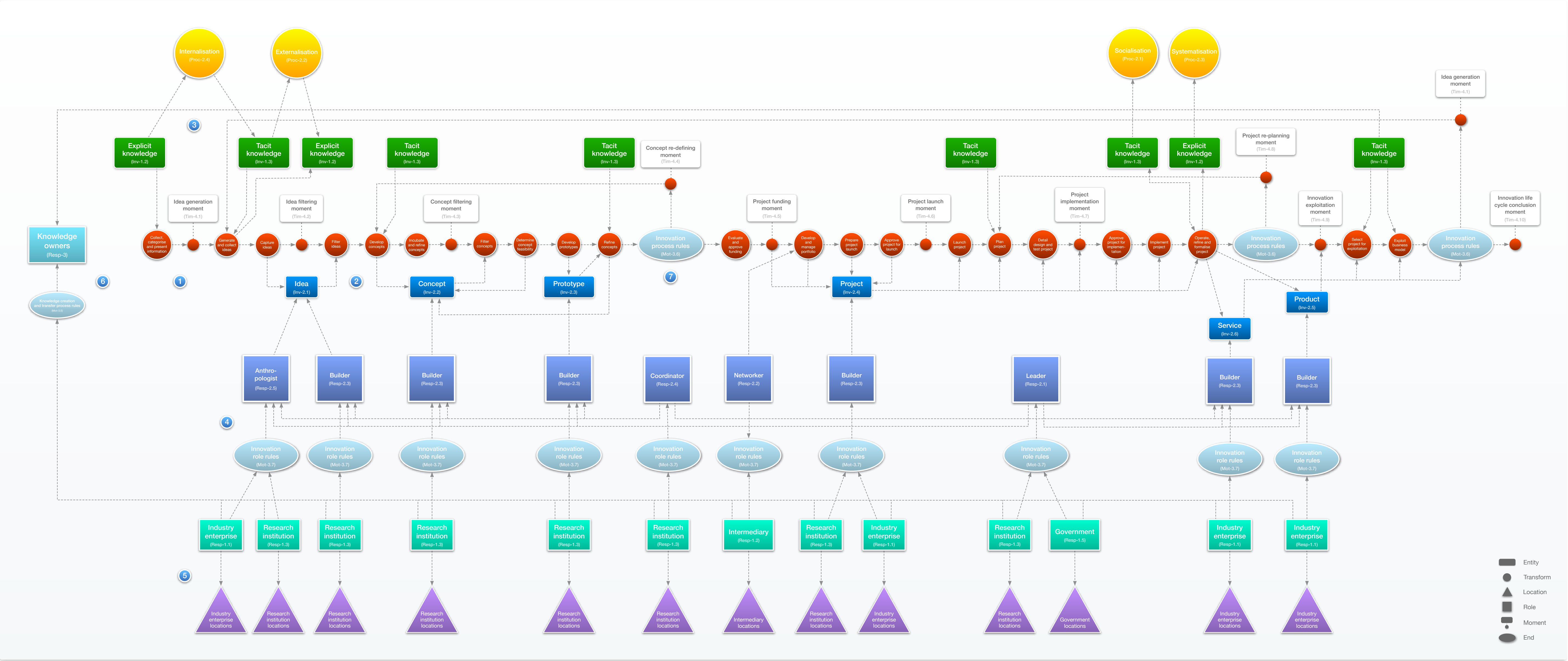


Figure 9.9: MediaLab/MIH composite model

9.5.2.2 Evaluation

*This section **interprets** evidence of the ability of the reference architecture for IKNs to provide inputs for the creation and assessment of composite particular models.*

The particular composite models demonstrated in the previous section describe the construction of the scenario networks from a business perspective as they are encountered in practice, i.e. as multi-variable constructs. The models, however, maintain an ontological view on the networks by separating their construction from their implementation. This is evident in various parts of the model demonstrated in Figure 9.9, including the fact that various references are made to *research institutions*, while in the implementation of the network only a single research institution is involved in the network, albeit in a number of different guises. The reusability of architectural parts from the primitive particular models is also evident from the construction of composite models.

The composite particular models provided insight into the construction of the scenario networks by indicating the links and dependencies between components from various abstractions. In composite models that describe an even more limited segment of the architecture of a network than the model presented in Figure 9.9, inclusion of the primitive links between components (i.e. relationships, inputs/outputs, connections, work products, intervals and means) would provide further insights into the construction of the network. These insights enable the engineering of IKNs, enabling them to respond to changes in their environment.

The ability to construct composite particular models provides further evidence for the ability of the reference architecture for IKNs to provide inputs for the development of accurate primitive particular models. The construction of the composite particular models were directed by the guidance of the reference architecture for IKNs on possibilities to integrate its reference models.

It is therefore possible to motivate that the reference architecture for IKNs succeeded in providing inputs for the creation and assessment of *composite* particular models that describe the construction of scenario networks. The artefact therefore **functions as a reference architecture in the engineering of IKNs through the implementation of a suitable architecture framework**.

9.5.3 Conclusion

Given the demonstration and evaluation of all the items in the checklist for this section, it is possible to motivate that the artefact **achieves SO4 by functioning as a reference architecture in the engineering of IKNs through the implementation of a suitable architecture framework.**

9.6 Artefact behaviour specification

This section presents the demonstration and evaluation of how the artefact attempts to achieve the green solution objective that was identified in section 4.4.1 and designed for in chapter 5. The checklist for this particular solution objective is discussed, after which each of its items are demonstrated and evaluated.

The solution objective in focus in this section of the demonstration and evaluation of the artefact is as follows:

Solution objective: To exhibit the desired behaviour of a reference architecture in the engineering of IKNs. (SO1)

The aim is therefore to show how the artefact attempts to simultaneously address its functional requirement and the scenarios in which it is expected to operate (refer to chapter 5), and to judge the success with which this is achieved.

The checklist used to demonstrate and evaluate the artefact's success in achieving the above solution objective is presented in Table 9.4. The content of this list is based on the identified desired behaviour of a reference architecture in the engineering of IKNs, and contains a number of features of the artefact to be demonstrated. Each of these items is accompanied by the related requirement that has to be met to achieve an element of the larger solution objective stated above.

Table 9.4: Checklist for demonstration and evaluation of architecture framework selection

Check	Requirement
1. Genericity	Partial reference architecture, aimed at engineering of IKNs.
2. Audience	Stakeholders involved in design, refinement and phase-out of IKNs.
3. Functional modes	Input for design of particular architectures, benchmarking tool.
4. Actions	Provide inputs for design of to-be architecture description when no as-is architecture description or instantiation is present.
	Provide inputs for design of operational as-is architecture description when only as-is instance is present.
	Assist in assessing performance of operational as-is architecture description by serving as benchmarking tool.
	Provide inputs for design of refined to-be architecture description when diagnosed, operational as-is architecture description is present.
	Provide inputs for design of phased-out to-be architecture description when operational as-is architecture description is present.

The items in this list were used to identify the desired behaviour of a reference architecture in the engineering of IKNs based on theory in chapter 5, and they are now employed to guide the practical demonstration and evaluation of this desired behaviour as exhibited by the final artefact. There is therefore a shift in focus from the identified desired behaviour to the success with which it is exhibited by the reference architecture for IKNs. Given that the purpose in this section is to evaluate the behaviour of a reference architecture in multiple illustrative scenarios, the primary source of data in the evaluation is the experience and insights of the architect, who was involved throughout the entire evaluation.

9.6.1 Genericity

The desired behaviour of the artefact includes that it should function as a partial reference architecture aimed at the engineering of IKNs. This means that the reference architecture should be free from implementation bias towards the engineering of any particular IKN, but also not so generic that it applies to the engineering of enterprises in general (refer to Figure 9.10) (1).

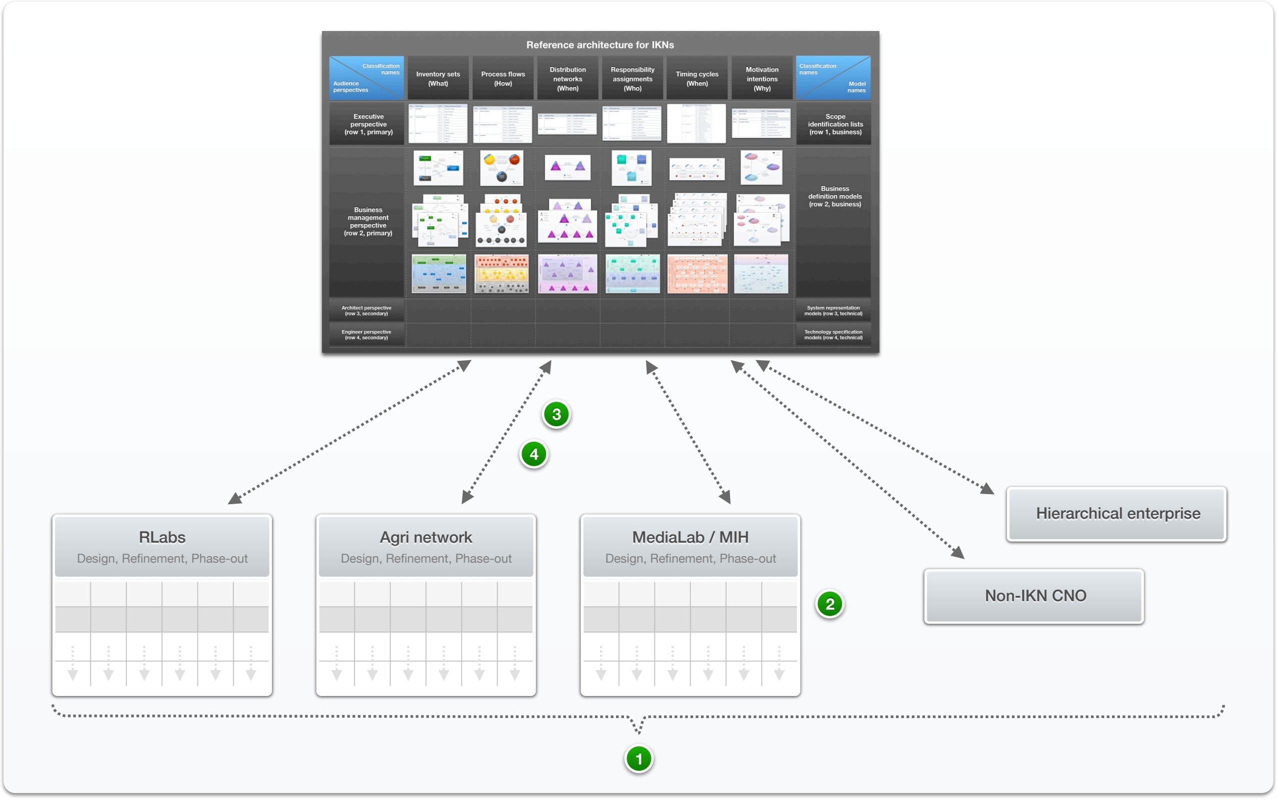


Figure 9.10: Demonstration and evaluation of artefact behaviour specification

9.6.1.1 Demonstration

*This section **presents** evidence of the artefact exhibiting the required level of genericity.*

This feature of the reference architecture for IKNs was demonstrated by implementing the artefact in multiple illustrative scenarios (refer to Figure 9.10) (1). Three of these scenarios featured the engineering of networks that can be regarded as representative instances of IKNs (refer to section 9.4).

The reference architecture for IKNs, however, was also demonstrated in principle in two further scenarios. The first of these featured the engineering of a CNO that did per definition *not* qualify to be regarded as an instance of an IKN, and the second featured the engineering of an enterprise that can be regarded as an instance of the engineering of a traditional, hierarchical enterprise. These additional scenarios were introduced as a control to verify the integrity of the demonstration results and to enable meaningful evaluation of the performance of the reference architecture. The entire group of scenarios therefore demonstrates both the lower and upper boundaries of the range of a partial reference architecture.

The reference architecture for IKN was found to be highly applicable to all three illustrative scenarios that can be regarded as instances of the engineering of IKNs. Evidence of this applicability was provided in the previous section where the overall functionality of the reference architecture was demonstrated and evaluated. Furthermore, no implementation bias to the engineering of any particular instance of an IKN was found. Moderate applicability was found for the scenario in which the engineering of a non-IKN CNO was performed in principle. Several elements of the reference architecture, e.g. the description of network stakeholders and mechanisms to circumvent geographical dispersion, was found to be applicable to a non-IKN networked organisation. Other elements, e.g. the focus on innovation as a network motivation and key process flow, did not apply. Only very limited applicability was found in the scenario featuring the engineering of a hierarchical enterprise, with especially elements such as network stakeholders not being applicable.

9.6.1.2 Evaluation

*This section **interprets** evidence of the artefact exhibiting the required level of genericity.*

A partial reference architecture is expected to be accurate for organisations that fall within the boundaries of its intended range. By being highly applicable to multiple illustrative scenarios where IKNs were engineered, the reference architecture for IKNs is shown to not be too particular and therefore free from implementation bias.

However, by only being moderately applicable to the engineering of an inter-organisational network that does not qualify as an IKN, it is also not too general. This is expected behaviour for a partial reference architecture, as it would have some degree of generalised applicability to subjects that are on the boundary of its range. A partial reference architecture, however, should only have very limited applicability on a generic level, i.e. significantly beyond the boundaries of its range, and this was found in the scenario in which the engineering of a hierarchical enterprise was attempted.

When comparing the results of the demonstration of the *genericity* of the artefact to this facet of the solution objective, it is therefore possible to motivate that the artefact **exhibited the desired behaviour of a reference architecture in the engineering of IKNs**.

9.6.2 Audience

The desired behaviour of the reference architecture for IKNs includes that it should provide functionality which is aimed at an audience that is involved in the design, refinement and phase-out phases of the IKN life cycle (refer to Figure 9.10) (2). In section 5.6.5.6 it was further elaborated that these users of the artefact will be engaging with the architecture of an IKN at a strategic and business level, although some secondary interaction should occur at a technical level.

9.6.2.1 Demonstration

*This section **presents** evidence of the artefact being aimed at the intended audiences.*

In the three illustrative scenarios that involve instances of IKNs, effort was made to involve representatives from the scenario networks that form part of the intended audience of the reference architecture in the data-gathering workshops (refer to Figure 9.10) (2). For the Agri network and MediaLab/MIH scenarios this resulted in workshop attendees from only the primary strategic audience, while for the RLabs network workshop attendees were distributed between the primary audience and the more technical secondary audience.

It was found that the reference architecture for IKNs was accurate in addressing its intended primary audience by providing functionality that is relevant on a strategic and business level. This was particularly evident from the fact that the artefact firstly assisted in identifying the boundaries of the particular IKN architectures, before assisting in the definition of business concepts. It was furthermore found that the artefact did not provide functionality that directly benefited the intended secondary audience. This was evident through the reference architecture not providing any direct inputs for the representation of system logic or the specification of the technologies.

All three scenario networks were in the operational phase of their life cycles when the workshops were conducted, and the focus was therefore on the use of the reference architecture in the refinement of the respective networks. In the RLabs and MediaLab/MIH networks there was, however, also interest in creating new instances of the networks elsewhere, which turned the attention to the use of the reference architecture in the design phase of an IKN. In all cases the use of the reference architecture in the phase-out of the networks was also discussed, although it was not an immediate priority for any of the networks.

9.6.2.2 Evaluation

*This section **interprets** evidence of the artefact being aimed at audiences that are involved in the design, refinement and phase-out of IKNs.*

The feedback obtained from the workshops conducted in each of the scenario networks indicated that the artefact does indeed provide functionality aimed at stakeholders involved with the design, refinement and phase-out of IKNs.

It is, however, understandable that the reference architecture does not provide functionality which is directly aimed at a technical audience that is involved with the implementation and operation of these networks. These stakeholders engage with the outputs produced by the primary audience, i.e. a business architecture description of their respective networks. Due to the time constraints of such an exercise, this description was not completed and formalised through multiple rounds of architectural work, as it would be in practice. The technical audience could see how the use of the artefact would indirectly benefit them, but struggled to engage with the artefact’s primary functionality. This secondary technical audience is predominantly involved in the implementation and operation phases of the IKN life cycle, and this further confirms that the reference architecture is accurate in the way in which it targets its audience.

When comparing the results of the demonstration of the *audience* of the artefact to this facet of the solution objective, it is therefore possible to motivate that the artefact **exhibited the desired behaviour of a reference architecture in the engineering of IKNs**.

9.6.3 Functional modes

The desired behaviour of the reference architecture for IKNs includes that it should be capable of two functional modes (refer to Figure 9.10) (3). The first is the provision of inputs for the design of new particular architecture descriptions, and the second is to function as a benchmarking tool for existing particular architecture descriptions.

9.6.3.1 Demonstration

*This section **presents** evidence of the artefact operating in the required functional modes.*

In the three illustrative scenarios that involve instances of IKNs, both functional modes were demonstrated for every network (refer to Figure 9.10) (3). No formal architecture descriptions existed for any of the networks and the artefact provided inputs for the design of new particular architecture descriptions for every network. These particular architecture descriptions were subsequently benchmarked against the reference architecture in order to identify opportunities for network refinement. Concept versions of refined particular architecture descriptions were also developed with the reference architecture once again providing inputs to these architecture designs. Evidence of these architectures and particular models was presented in section 9.5.

9.6.3.2 Evaluation

*This section **interprets** evidence of the artefact operating in the required functional modes.*

When comparing the results of the demonstration of the *functional modes* of the artefact to this facet of the solution objective, it is possible to motivate that the artefact **exhibited the desired behaviour of a reference architecture in the engineering of IKNs**.

The fact that none of the scenario networks had architecture descriptions available that accurately describe their operational architecture confirms the assertion of the research problem considered in this study (refer to section 1.3). All three the networks were emergent phenomena that, although they showed clear signs of intentional endeavour, were by no means engineered and, thus, highly vulnerable to changes in their environment.

9.6.4 Actions

The desired behaviour of the reference architecture for IKNs includes that it should be able to perform five actions (refer to Figure 9.10) (4). These are as follows:

1. Provide *inputs* for the *design* of a *to-be architecture* description when *no as-is architecture* description or *instantiation* is present.
2. Provide *inputs* for the *design* of an *operational as-is architecture* description when only an *as-is instance* is present.
3. Assist in *assessing* the performance of an *operational as-is architecture* description by functioning as a benchmarking tool.
4. Provide *inputs* for the *design* of a *refined to-be architecture* description when a diagnosed, *operational as-is architecture* description is present.
5. Provide *inputs* for the *design* of a *phased-out to-be architecture* description when an *operational as-is architecture* description is present.

9.6.4.1 Demonstration

*This section **presents** evidence of the artefact performing the required actions.*

As none of the scenario networks had particular architecture descriptions for their operational networks available, the second action was initially demonstrated for each network (refer to Figure 9.10) (4). This involved developing architecture descriptions for each of the networks with the reference architecture for

IKNs providing inputs for these descriptions. Evidence of these architectures and particular models was presented in section 9.5.

The third action was then demonstrated for each network, as their newly developed architecture descriptions were benchmarked against the reference architecture for IKNs. This action provided insight into possible options for refinement of the network. The fourth action was demonstrated for each scenario network through the design of initial refined architecture descriptions, while using the reference architecture for IKNs as a source of inputs. The fifth action was subsequently demonstrated in principle for every network by using the reference architecture as a source of inputs for hypothetical phased-out architecture descriptions for the networks.

As operational instances of IKNs already existed in all the illustrative scenarios, it was not possible to demonstrate the first action to the same degree in these contexts, even when the network representatives were interested in replicating their networks elsewhere. This action, however, was demonstrated in principle by developing a designed architecture description for a hypothetical new IKN.

9.6.4.2 Evaluation

*This section **interprets** evidence of the artefact performing the required actions.*

All five the specified required actions were demonstrated either in practice or in principle in the illustrative scenarios. When comparing the results of the demonstration of the *actions* of the artefact to this facet of the solution objective, it may be argued that the artefact **exhibited the desired behaviour of a reference architecture in the engineering of IKNs**.

The solution objective demonstrated and evaluated in this section does not address the architecture framework employed by the artefact. It should, however, be noted that the ease with which the third, fourth and fifth actions can be performed are highly reliant on the level of compatibility with the inherent architecture framework (if any) employed in the existing architecture descriptions and that employed by the reference architecture for IKNs. In cases in which this compatibility is very low, it may require that the second action be performed, even though an architecture description exists, in order to produce a new architecture description that better relates to the inputs received from the reference architecture for IKNs in subsequent actions.

9.6.5 Conclusion

Given the demonstration and evaluation of all the items in the checklist for this solution objective, it may be argued that the artefact **achieves SO1 by exhibiting the required behaviour of a reference architecture in the engineering of IKNs.**

9.7 Architecture framework selection

This section presents the demonstration and evaluation of how the artefact attempts to achieve the yellow solution objective that was identified in section 4.4.2 and designed for in chapter 6. The checklist for this particular solution objective is discussed, after which each of its items are demonstrated and evaluated.

The solution objective in focus in this section of the demonstration and evaluation of the artefact is as follows:

Solution objective: To employ an architecture framework that is suitable to the engineering of IKNs. (SO2)

The aim is therefore to show how the artefact attempts to simultaneously address its technical requirement and the scenarios in which it is expected to operate (refer to chapter 6), and to judge the success with which this is achieved.

The checklist used to demonstrate and evaluate the artefact's success in achieving the above solution objective is presented in Table 9.5. The content of this list is based on the criteria that were used to initially select the architecture framework. It contains a number of features of the artefact to be demonstrated, along with the related requirements that have to be met to achieve an element of the larger solution objective stated above.

Table 9.5: Checklist for demonstration and evaluation of architecture framework selection

Check	Requirement
1. Type of information	Business, organisation, technical
2. Scope	Enterprise, network
3. Quality attribute	Ontological view, flexibility, reusability
4. Stakeholders	Enterprise architects, strategists, concept owners, system architects, engineers, technicians
5. Transformations	Current, future
6. Nature	Models

This list was used to select a suitable architecture framework based on theory in chapter 6 and it is now employed to guide the practical demonstration and evaluation of this architecture framework selection as part of the final artefact. There is therefore a shift in focus from the selected architecture framework itself to the success with which it is employed within the reference architecture for IKNs. Given that the purpose in this section is to evaluate the architecture framework that is employed in the reference architecture for

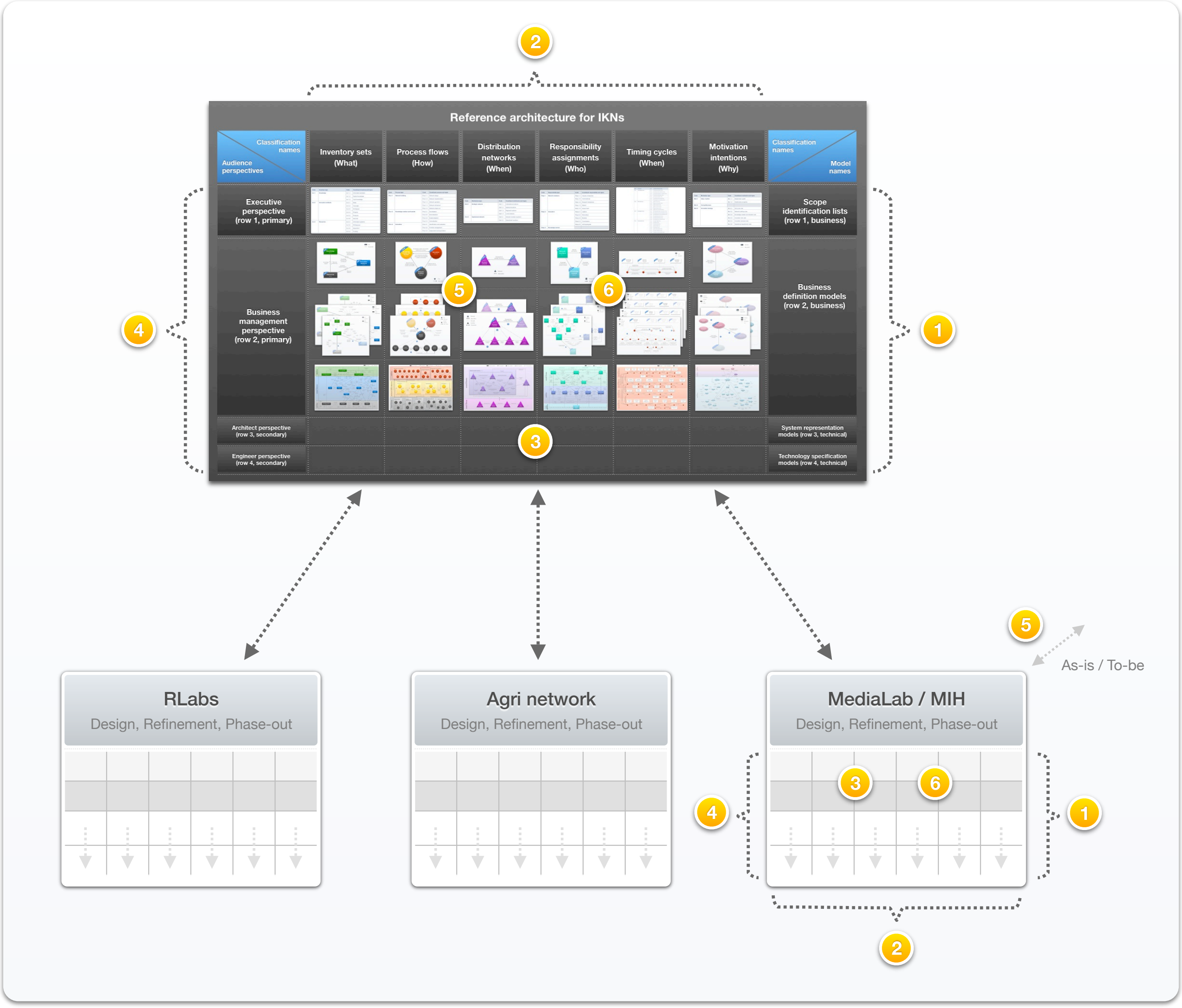


Figure 9.11: Demonstration and evaluation of architecture framework selection

IKNs, the primary source of data in the evaluation design is the experience and insights of the architect that has knowledge of architecture frameworks.

9.7.1 Type of information

The architecture framework employed by the reference architecture for IKNs should structure business, organisational and technical information in order to comprehensively describe the construction of these networks (refer to Figure 9.11) (1).

9.7.1.1 Demonstration

*This section **presents** evidence of the artefact structuring architectural information that is of a business, organisational and technical type.*

In all three illustrative scenarios presented in section 9.4 it was demonstrated that both the reference architecture for IKNs as well as the architecture descriptions it produces contain architectural information of a business and organisational type. This is evident in the emphasis on strategic and conceptual information in the first-row and second-row models that the reference architecture provides and also produces (refer to Figure 9.11 (1); section 9.5). The reference architecture furthermore contains directives for the development of further technical perspectives in the particular architecture descriptions based on the business and organisational perspectives they contain. The artefact specifies that the primitive business definition models (second row) of particular architecture descriptions be transformed into primitive system representation models (third row). These system representation models are in turn transformed into primitive technology specification models (fourth row).

9.7.1.2 Evaluation

*This section **interprets** evidence of the artefact structuring architectural information that is of a business, organisational and technical type*

The reference architecture for IKNs includes sufficient business and organisational information in order to describe the construction of IKNs through these perspectives, especially given the scope of this information (refer to section 9.7.2). Due to the feasibility restrictions on reference models for the technical perspectives, the nature of the technical information included in the artefact is limited to directives for the development of technical perspectives in particular architecture descriptions. The combination of structured business, organisational and technical information is a direct effect of the implementation of the Zachman framework.

The type of information that is structured by the artefact is, however, well suited to describe the construction of IKNs. It is therefore possible to motivate that the reference architecture **employs an architecture framework that is suitable to the engineering of IKNs.**

9.7.2 Scope

The architecture framework employed by the reference architecture for IKNs should have an enterprise-wide scope that enables it to describe the construction of networks (refer to Figure 9.11) (2).

9.7.2.1 Demonstration

*This section **presents** evidence of the artefact having the required scope.*

The scope of the Zachman framework, as employed by the reference architecture for IKNs, was demonstrated by considering all six included abstractions (columns) for each of the three scenario networks (refer to Figure 9.11) (2). The inventory sets, process flows, distribution networks, responsibility assignments, timing cycles and motivation intentions of each network were modelled in order to describe the construction of each network.

It was found that it was possible to model the scenario networks in all six abstractions, and that no additional abstractions could be identified to be included in the reference architecture. All suggestions for enterprise constructs that did not directly fit any of the abstractions were found to be composite (multi-variable) architectural constructions that could be deconstructed into primitive elements fitting the architecture framework scope. Furthermore, it was found that once the scenario networks were comprehensively modelled according to the architecture framework employed, all conceivable composite enterprise concepts could be constructed from the available primitive elements.

9.7.2.2 Evaluation

*This section **interprets** evidence of the artefact having the required scope.*

The six abstractions that comprise the reference architecture for IKNs provide a holistic, enterprise-wide view of the construction of these networks. The scope of the information included in the artefact is sufficient to describe the construction of IKNs comprehensively. It is therefore possible to motivate that the reference architecture **employs an architecture framework which is suitable to the engineering of IKNs.**

9.7.3 Quality attribute

The architecture framework employed by the reference architecture for IKNs should exhibit the qualities of an ontological view on the networks which is independent of their implementation, flexibility of construction and reusability of parts (refer to Figure 9.11) (3).

9.7.3.1 Demonstration

*This section **presents** evidence of the artefact exhibiting the required quality attribute.*

It was demonstrated that the Zachman framework, as employed by the reference architecture for IKNs, provides an ontological view on the scenario networks that is independent of their implementation (refer to Figure 9.11) (3). This is most evident in the separation between the architectural perspectives of the models comprising the top two rows of the particular architecture descriptions produced in the illustrative scenarios (refer to section 9.5), and the networks themselves. This is evident in various parts of the model demonstrated in Figure 9.9, including the fact that various references are made to *research institutions*, while in the implementation of the network only a single research institution is involved in the network, albeit in a number of different guises.

It was furthermore demonstrated that the reference architecture, and therefore also the architecture descriptions it produces, exhibits flexibility of construction as an architectural quality. The structure of primitive elements as prescribed by the Zachman framework indicates how these elements are integrated within as well as between abstractions. These integrations highlight the implications of changes in the architecture, leading to greater adaptability, which is a hallmark of architectural flexibility. Primitive elements in both the reference architecture and the architecture descriptions may also be combined in various ways to create composite constructs, leading to the reusability of architectural parts in multiple contexts (refer to section 9.5.2).

9.7.3.2 Evaluation

*This section **interprets** evidence of the artefact exhibiting the required quality attribute.*

The operational networks in the various illustrative scenarios represent the sixth-row instances of the particular architectures, i.e. the implementation of the network, while the architecture descriptions describe the construction of the network, independent of implementation. This ontological view that separates construction and implementation is a direct effect of employing the Zachman framework in the reference architecture for IKNs. The architectural flexibility and reusability of parts as demonstrated in the construction of composite particular models, may also be interpreted as an effect of employing the Zachman framework in the artefact.

The qualities of the reference architecture for IKNs and the architecture descriptions it produces include an ontological view on the networks, architectural flexibility and reusability of architectural parts. It is therefore possible to motivate that the reference architecture **employs an architecture framework that is suitable to the engineering of IKNs.**

9.7.4 Stakeholders

The architecture framework employed by the reference architecture for IKNs should address a wide array of stakeholders with interest in the construction of these networks, including enterprise architects, strategists, concept owners, system architects, engineers and technicians (refer to Figure 9.11) (4).

9.7.4.1 Demonstration

*This section **presents** evidence of the artefact addressing the required stakeholders.*

The architecture workshops that were conducted in the various illustrative scenarios were designed to include the intended stakeholders that the reference architecture for IKNs should address. It was demonstrated that the Zachman framework, as employed by the reference architecture for IKNs, directly addresses enterprise architects through its intention to describe the construction of the scenario networks (refer to Figure 9.11) (4). This intention falls within the domain and mandate of these stakeholders. Furthermore, employing the Zachman framework results in the reference models constituting the reference architecture for IKNs to be aimed at strategists (first row) and concepts owners (second row). The degree to which the Zachman framework addresses system architects, engineers and technicians was, however, not extensively demonstrated, given that these stakeholders fell outside the primary audience of the reference models included in the reference architecture.

It was found that it was important to match the appropriate stakeholders with the architectural information they could take ownership of. The information required to generate the particular first-row models reside with business context planners and the information for the particular second row models with business concept owners. It was therefore important to engage the correct representatives from the scenario networks in order to accurately gather data. Stakeholders with a more technical background, e.g. the system architects, engineers and technicians described above, had difficulty directly relating to the architectural content in the more business-oriented perspectives.

9.7.4.2 Evaluation

*This section **interprets** evidence of the artefact addressing the required stakeholders.*

The reference architecture for IKNs accurately addresses enterprise architects, strategists and concept owners, but lacks technical information that directly addresses stakeholders with a more technical background. This is, however, an effect of the feasibility of reference models at these technical levels of the construction of IKNs, and rather highlights the pitfalls of mismatching information owners with modelling intentions than a shortcoming of the Zachman framework.

The Zachman framework does indeed address technical stakeholders, but in the reference architecture for IKNs this is done only through directives for the development of technical perspectives in architectural descriptions based on the already developed business perspectives. The Zachman framework specifies that primitive business definition models be transformed into primitive system representation models, which are in turn transformed into primitive technology specification models.

It could therefore be reasonably expected that these technical stakeholders would also be addressed when particular architecture descriptions, that inherit the employed architecture framework from the reference architecture for IKNs, were developed up to this technical level. Given the range of stakeholders that are addressed to provide a multi-layered view of the construction of IKNs, it is possible to motivate that the reference architecture **employs an architecture framework that is suitable to the engineering of IKNs.**

9.7.5 Transformations

The architecture framework employed by the reference architecture for IKNs should enable the creation of both current (as-is) and future (to-be) architecture descriptions (refer to Figure 9.11) (5).

9.7.5.1 Demonstration

*This section **presents** evidence of the artefact enabling the required time-based transformations.*

In all three illustrative scenario it was shown that the architecture framework employed by the reference architecture for IKNs enabled the creation of both current and future architecture descriptions for each of the networks (refer to section 9.6.4.1). The structure of the produced architecture descriptions allowed for both as-is and to-be models in each cell, while the reference architecture provided inputs for the creation of both kinds of particular models (refer to Figure 9.11) (5). Evidence of these as-is and to-be models are presented in section 9.5.1.1.

9.7.5.2 Evaluation

*This section **interprets** evidence of the artefact enabling the required time-based transformations.*

The reference architecture for IKNs enables both current and future transformations in the architecture descriptions it produces. It is therefore possible to motivate that the artefact **employs an architecture framework that is suitable to the engineering of IKNs.**

9.7.6 Nature

The nature of the architectural information structured by the architecture framework employed by the reference architecture for IKNs should be model-based (refer to Figure 9.11) (6). This requirement enables the construction of “white-box” models that indicate the construction of these networks.

9.7.6.1 Demonstration

*This section **presents** evidence of the artefact containing architectural information in models.*

It was demonstrated that the nature of information required and structured by the Zachman framework in both the reference architecture and architecture descriptions was model-based (refer to Figure 9.11) (6). The reference architecture itself contains 36 reference models, while the architecture descriptions of the scenario network contain multiple particular models. The intention of both the reference models and particular models is to indicate the construction of its modelling subject, be it either the typical IKN or a specific IKN instance, thereby constituting a “white-box” model.

9.7.6.2 Evaluation

*This section **interprets** evidence of the artefact containing architectural information in models.*

The architectural information structured by the reference architecture for IKNs and the architecture descriptions it produces is contained in models. These models indicate the construction of their modelling subject, and it is therefore possible to motivate that the artefact **employs an architecture framework that is suitable to the engineering of IKNs.**

9.7.7 Conclusion

Given the demonstration and evaluation of all the items in the checklist for this solution objective, it may be argued that the artefact **achieves SO2 by employing an architecture framework that is suitable to the engineering of IKNs.**

9.8 Artefact development roadmap

This section presents the demonstration and evaluation of how the artefact attempts to achieve the red solution objective that was identified in section 4.4.3 and designed for in chapter 7. The checklist for this particular solution objective is discussed, after which each of its items are demonstrated and evaluated.

The solution objective focused on in this section of the demonstration and evaluation of the artefact is as follows:

Solution objective: To implement the selected architecture framework in a way that enables the artefact to function as a reference architecture. (SO3)

The aim is therefore to show how the artefact attempts to simultaneously address its technical and functional requirements (refer to chapter 7), and to judge the success with which this is achieved.

The checklist used to demonstrate and evaluate the artefact's success in achieving the above solution objective as presented in Table 9.6. The content of this list is based on the artefact development roadmap and contains a number of features to be demonstrated, along with the related requirement that has to be met to achieve an element of the larger solution objective stated above.

Table 9.6: Checklist for demonstration and evaluation of artefact development roadmap

Check	Requirement
1. Meta-level	Reference architecture.
2. Nature	Reference models.

Check	Requirement
3. Scope	Enterprise.
4. Stakeholders	Primary (business), secondary (technical).
5. Type	Business.
6. Transformations	Versioned (current, future).
7. Detail	Varying levels.
8. Representation	Informal, semi-formal.

This list was used to show how the selected architecture framework could be implemented to produce a reference architecture in chapter 7. It is now employed to guide the practical demonstration and evaluation of this implementation as part of the final artefact. For this reason a shift in focus from the designed implementation of the architecture framework to the success with which it enables the function of the reference architecture for IKNs occurs. Given that the purpose in this section is to evaluate the way in which the selected architecture framework is implemented in the reference architecture for IKNs, the primary source of data in the evaluation design is the architect that has knowledge of architecture frameworks.

9.8.1 Meta-level

The implementation of the selected architecture framework should enable the artefact to have the appropriate meta-level required of a reference architecture (refer to Figure 9.12) (1).

9.8.1.1 Demonstration

*This section **presents** evidence of the implementation of the Zachman framework enabling the artefact to have the meta-level of a reference architecture.*

It was demonstrated in all three illustrative scenarios that the implementation of the Zachman framework resulted in an artefact that has the meta-level of a reference architecture (refer to Figure 9.12) (1). This was done by showing that the artefact does not describe the architecture of any particular IKN, but rather contains generalised architectural patterns that describe a typical configuration of an IKN. The artefact was therefore able to provide inputs for the development of particular architecture descriptions (refer to section 9.5).

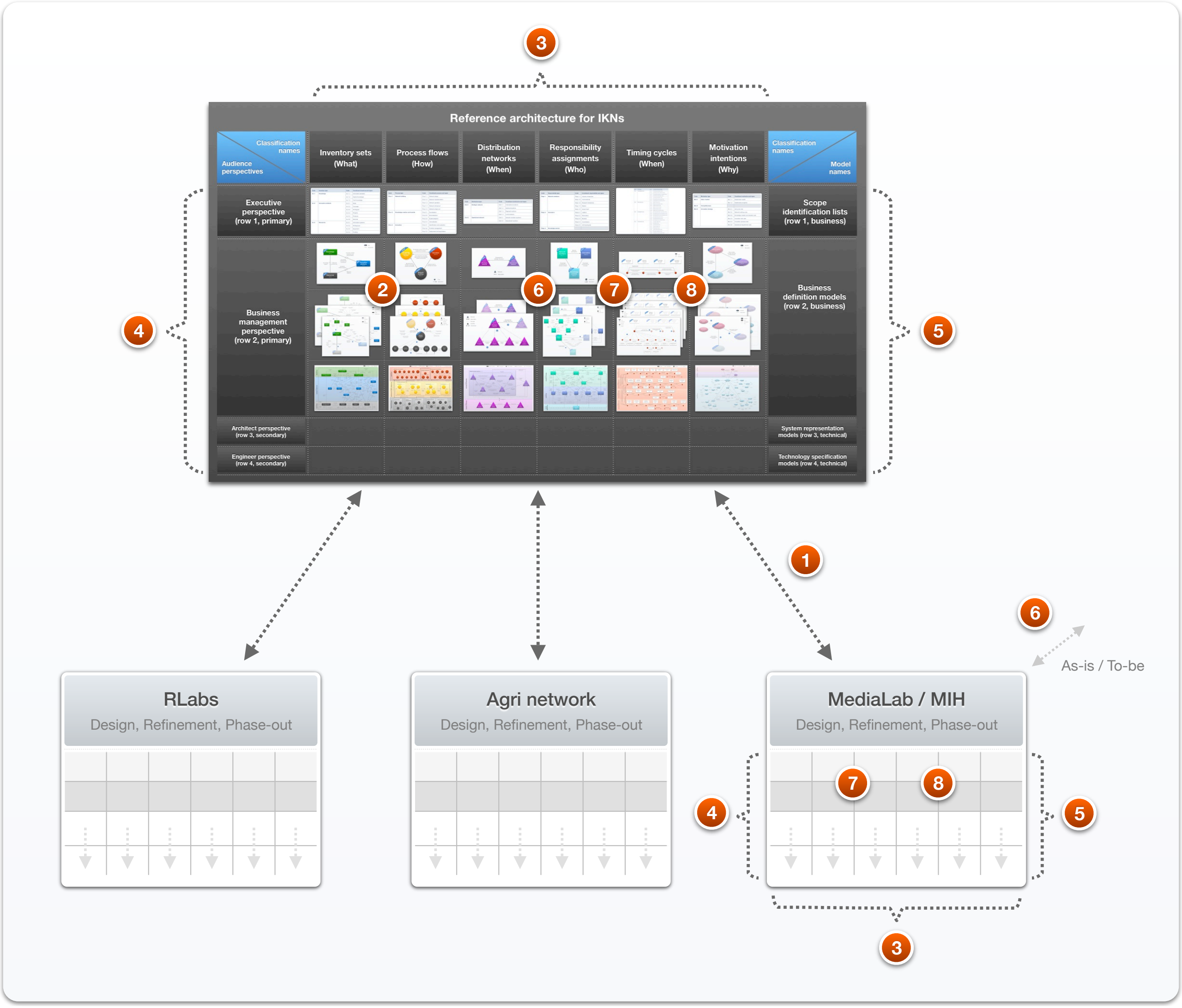


Figure 9.12: Demonstration and evaluation of artefact development roadmap

9.8.1.2 Evaluation

*This section **interprets** evidence of the implementation of the Zachman framework enabling the artefact to have the meta-level of a reference architecture.*

Although architectural artefacts that implement the Zachman framework regularly have the meta-level of a particular architecture, its implementation in the reference architecture for IKNs is explicitly specified to have a more generalised meta-level. It can therefore be motivated that the implementation of the architecture framework **enables the artefact to function as a reference architecture**.

9.8.2 Nature

The implementation of the selected architecture framework should enable the architectural information contained in the reference architecture to have the nature of reference models (refer to Figure 9.12) (2).

9.8.2.1 Demonstration

*This section **presents** evidence of the implementation of the Zachman framework enabling the artefact to contain information that has the nature of reference models.*

It was demonstrated that the implementation of the Zachman framework resulted in an artefact that contains reference models (refer to Figure 9.12) (2). These reference models do not describe any particular IKN, but rather represent generalised patterns found in the architectures of IKNs. It could therefore be said that the artefact describes a hypothetical typical configuration of an IKN. It was shown that for each illustrative scenario the reference models served as a starting point for the development of particular models that more accurately described the relevant scenario network (refer to section 9.5). Elements were therefore adopted from the reference models, in accordance with their relevance to the network construction, and then customised and extended to describe the particular network architecture.

9.8.2.2 Evaluation

*This section **interprets** evidence of the implementation of the Zachman framework enabling the artefact to contain information that has the nature of reference models.*

While architectural artefacts that implement the Zachman framework regularly contain particular models in the prescribed classification schema, the implementation of the framework in the reference architecture for IKNs contains reference models. This is in keeping with the demonstrated meta-level of the artefact (refer to section 9.8.1). It may therefore be argued that the implementation of the architecture framework therefore **enables the artefact to function as a reference architecture**.

9.8.3 Scope

The implementation of the selected architecture framework should enable the reference architecture to have an enterprise scope (refer to Figure 9.12) (3). A clearly defined and adequate scope is a key characteristic of an architectural artefact (Greefhorst et al. 2006; Schekkerman 2004).

9.8.3.1 Demonstration

*This section **presents** evidence of the implementation of the Zachman framework providing the artefact with an enterprise scope.*

It was demonstrated that the implementation of the Zachman framework enabled an enterprise-wide scope by modelling the three scenario networks in terms of six abstractions (refer to Figure 9.12) (3). The implementation of these abstractions prescribed by the Zachman framework resulted in the reference architecture for IKNs having six independent columns, each considering a different interrogative, i.e. "*What?*", "*How?*", "*Where?*", "*How?*", "*When?*" and "*Why?*".

It was found that no additional abstractions could be identified to be included in the reference architecture. All suggestions for enterprise constructs that did not directly fit any of the abstractions were found to be composite (multi-variable) architectural constructions that could be deconstructed into primitive elements fitting the architecture framework scope. Furthermore, it was found that once the scenario networks were comprehensively modelled according to the architecture framework employed, all conceivable composite enterprise concepts could be constructed from the available primitive elements.

9.8.3.2 Evaluation

*This section **interprets** evidence of the implementation of the Zachman framework providing the artefact with an enterprise scope.*

The complete set of abstractions included in the reference architecture for IKNs constitute an enterprise-wide scope as, once populated, they provide all the required information to model an IKN from a holistic point of view. The inclusion of these abstractions are a direct effect of the implementation of the Zachman framework in the artefact. It can therefore by motivated that the implementation of the architecture framework **enables the artefact to function as a reference architecture** with a clearly defined and adequate scope.

9.8.4 Stakeholders

The implementation of the selected architecture framework should enable the artefact to address an audience that is comprised of business-oriented primary stakeholders, as well as technical-oriented secondary stakeholders (refer to Figure 9.12) (4). A clearly defined group of stakeholders is a key characteristic of an architectural artefact (Greefhorst et al. 2006; Schekkerman 2004).

9.8.4.1 Demonstration

*This section **presents** evidence of the implementation of the Zachman framework enabling the artefact to address the required stakeholders.*

It was demonstrated in all three illustrative scenarios that the implementation of the Zachman framework primarily addressed stakeholders with a business perspective on IKNs (refer to Figure 9.12) (4). It was found that the representatives from the scenario networks that had a strategic or conceptual perspective on their networks could directly exploit the reference models provided by the reference architecture. Representatives with a more systems-based or technology-based point of view, however, could also appreciate the value of the reference architecture. Of particular note to these stakeholders was the primitive model structure that is prescribed by the architecture framework for the perspectives that they are involved in.

9.8.4.2 Evaluation

*This section **interprets** evidence of the implementation of the Zachman framework enabling the artefact to address the required stakeholders.*

Due to the way in which the Zachman framework is implemented in the reference architecture for IKNs, the artefact addresses stakeholders with both a business and technical perspective on IKNs. It can therefore be motivated that the implementation of the architecture framework **enables the artefact to function as a reference architecture** aimed at a focussed group of appropriate stakeholders.

9.8.5 Type

The implementation of the selected architecture framework should enable the artefact to contain business information (refer to Figure 9.12) (5). A key characteristic of an architectural artefact is the identification of the type of architectural information that it contains and structures (Greefhorst et al. 2006; Schekkerman 2004).

9.8.5.1 Demonstration

*This section **presents** evidence of the implementation of the Zachman framework enabling the artefact to contain business information.*

It was shown that the reference architecture for IKNs is populated with business information spanning the top two perspectives (rows) of the Zachman framework (refer to Figure 9.12) (5). The reference models in the top row provide inputs for the identification of the boundaries of particular IKN architecture descriptions. The reference models in the second row provide inputs for the definition of the business concepts that fall within these boundaries. It was furthermore demonstrated that the reference architecture does contain directives for the development of technical perspectives, but that no reference models are supplied for these perspectives (rows).

9.8.5.2 Evaluation

*This section **interprets** evidence of the implementation of the Zachman framework enabling the artefact to contain business information.*

The implementation of the Zachman framework in the artefact is limited to generalised business information in the top two rows, complemented by directives for the development of the third and fourth rows in architecture descriptions. This results in the contents of the artefact being limited to a focussed type of architectural information for which reference models are feasible. It may therefore be argued that the implementation of the Zachman framework **enables the artefact to function as a reference architecture**.

9.8.6 Transformations

The implementation of the selected architecture framework should enable the artefact to produce versioned architecture descriptions with both current (as-is) and future (to-be) transformations (refer to Figure 9.12) (6).

9.8.6.1 Demonstration

*This section **presents** evidence of the implementation of the Zachman framework enabling the artefact to support time-based transformations.*

It was demonstrated in the illustrative scenarios that the reference models specified by the implementation of the Zachman framework enable the creation of multiple versions of derivative particular models (refer to Figure 9.12) (6). These particular models could represent elements of either current (as-is) or future (to-be) scenario architectures (refer to section 9.5.1). The reference models themselves, however, are agnostic of transformations, and therefore have the ability to serve as starting points for the development of either version.

9.8.6.2 Evaluation

*This section **interprets** evidence of the implementation of the Zachman framework enabling the artefact to support time-based transformations.*

The reference architecture for IKNs supports the development of both current and future versions of particular models. It can therefore be motivated that the implementation of the selected architecture framework **enables the artefact to function as a reference architecture**.

9.8.7 Detail

The implementation of the selected architecture framework should enable the artefact to support varying levels of detail in its architectural information (refer to Figure 9.12) (7).

9.8.7.1 Demonstration

*This section **presents** evidence of the implementation of the Zachman framework enabling the artefact to support varying levels of detail.*

It was demonstrated in all three illustrative scenarios that the reference architecture for IKNs contains reference models, and produces particular models, that feature varying levels of detail (refer to Figure 9.12) (7). It was shown that the implementation of the Zachman framework did not result in a growth in detail between rows, but that this growth in detail was possible within each row without transforming to a different perspective.

In the first-row reference models indented lists are employed to introduce a variance in detail. It was found to be helpful to introduce workshop attendees to the intention of the models via the lowest level of detail, and subsequently explore the higher detail levels of the reference models and their own particular models. The variance in detail that exists between the second-row reference models contained in the artefact was also found to be helpful in structuring the development of particular definition models. Particular models with a lower level of detail were also easier to convey to network representatives when feedback was presented on their particular architectures. Once the lowest level of detail was assimilated, it was possible to explore the higher level of detail of a particular concept, as well as of an entire abstraction.

9.8.7.2 Evaluation

*This section **interprets** evidence of the implementation of the Zachman framework enabling the artefact to support varying levels of detail.*

The variance in detail in the implementation of the Zachman framework made the reference architecture for IKNs more accessible to scenario network representatives. It was necessary to communicate to representatives that the transition between rows did not imply a growth in detail, but rather a change in perspective, and that various levels of detail were possible in each cell. Given the demonstration results, it can be motivated that the varying levels of detail in the implementation of the Zachman framework **enable the artefact to function as a reference architecture**.

9.8.8 Representation

The implementation of the selected architecture framework should enable the artefact to represent architectural information in both informal and semi-formal formats (refer to Figure 9.12) (8).

9.8.8.1 Demonstration

*This section **presents** evidence of the implementation of the Zachman framework enabling the artefact to represent architectural information.*

It was demonstrated that the reference models constituting the artefact utilised both informal and semi-formal representation formats (refer to Figure 9.12) (8). An informal representation format was implemented for the top-row reference models through language-based indented lists. A semi-formal representation format was implemented in the second-row reference models through both visual and textual formats. It was shown that the visual versions of the reference models adhere to meta-models for their specific abstractions, while the textual versions of these models also follow a uniform structure in their presentation of information.

9.8.8.2 Evaluation

*This section **interprets** evidence of the implementation of the Zachman framework enabling the artefact to represent architectural information.*

The representation formats utilised by the implementation of the Zachman framework (particularly those of the second-row models) are not linked to any existing or proprietary formats, which both have their advantages and disadvantages. The freedom from existing formats avoided any compatibility issues with modelling approaches already in use in the scenario networks, and therefore enhanced the artefact's ability to function as a reference architecture in multiple network contexts.

A drawback, however, was that network representatives were not familiar with the representational language, and had to become accustomed to it before accelerated progress could be made in modelling efforts. It was especially clear that the primitive (single-variable) modelling approach was one that few workshop attendees had come across before, and this took a while to grasp. In cases with very low compatibility between existing modelling approaches and the representational language implemented by the reference architecture for IKNs, it could also result in a remodelling effort before the inputs from the artefact could be effectively employed (refer to section 9.6.4.2).

Despite the learning curve involved, it can be motivated that the representational language employed in the implementation of the Zachman framework **enabled the artefact to function as a reference architecture**.

9.8.9 Conclusion

Given the demonstration and evaluation of all the items in the checklist for this section, it may be argued that the artefact **achieves SO3 by implementing the selected architecture framework in a way that enables the artefact to function as a reference architecture.**

9.9 Conclusion

This chapter presented the fourth activity in the research design for this study, and demonstrated and evaluated the artefact that was designed and developed in the third activity of the research design (refer to Part 3).

The demonstration phase of this activity showed how the artefact aims to achieve the solution objectives identified in Part 2 of the document, as implemented in a number of illustrative scenarios. The evaluation phase then reflected on the performance of the artefact and the extent to which it achieved these solution objectives. This combination of the demonstration and evaluation phases tested the artefact's utility, i.e. to act as a solution that achieves its specified objectives.

In order to demonstrate the artefact, the cooperation of a number of operational IKNs was enlisted to serve as illustrative scenarios. For each of these scenario networks, the reference architecture for IKNs was implemented in as close an approximation of its expected working environment as was possible. Workshops aimed at describing the architecture of the operational networks were conducted with representatives from the organisations that participate in each network, with the artefact playing its role as reference architecture in each case. Checklists were used to direct the demonstration of the artefact from various perspectives in order to ensure that sufficient data was gathered for meaningful evaluation of its performance.

Firstly, the success of the artefact in functioning as a reference architecture in the engineering of IKNs through the implementation of a suitable architecture framework was demonstrated and evaluated. This simultaneously considered the scenarios, technical and functional requirements of the artefact (SO4, blue, refer to chapter 8). The way in which the reference architecture for IKNs provides inputs for the creation and assessment of primitive particular models was demonstrated and evaluated first, followed by its inputs to composite particular models. The performance of the artefact in the illustrative scenarios leads to the conclusion that it can be motivated that the artefact *achieves SO4 by functioning as a reference architecture in the engineering of IKNs through the implementation of a suitable architecture framework*.

The degree to which the artefact exhibits the desired behaviour of a reference architecture in the engineering of IKNs was demonstrated and evaluated next. This placed the emphasis on the combination of functional requirements and scenarios as expressed in the artefact behaviour specification (SO1, green, refer to chapter 5). Attention was given to the validity of the reference architecture for multiple networks, as well as the stakeholders that it addressed. Its functional modes and the actions in which it is useful were furthermore demonstrated and evaluated. By considering these various perspectives of the artefact's performance, it can be motivated that the artefact *achieved SO1 by exhibiting the desired behaviour of a reference architecture in the engineering of IKNs*.

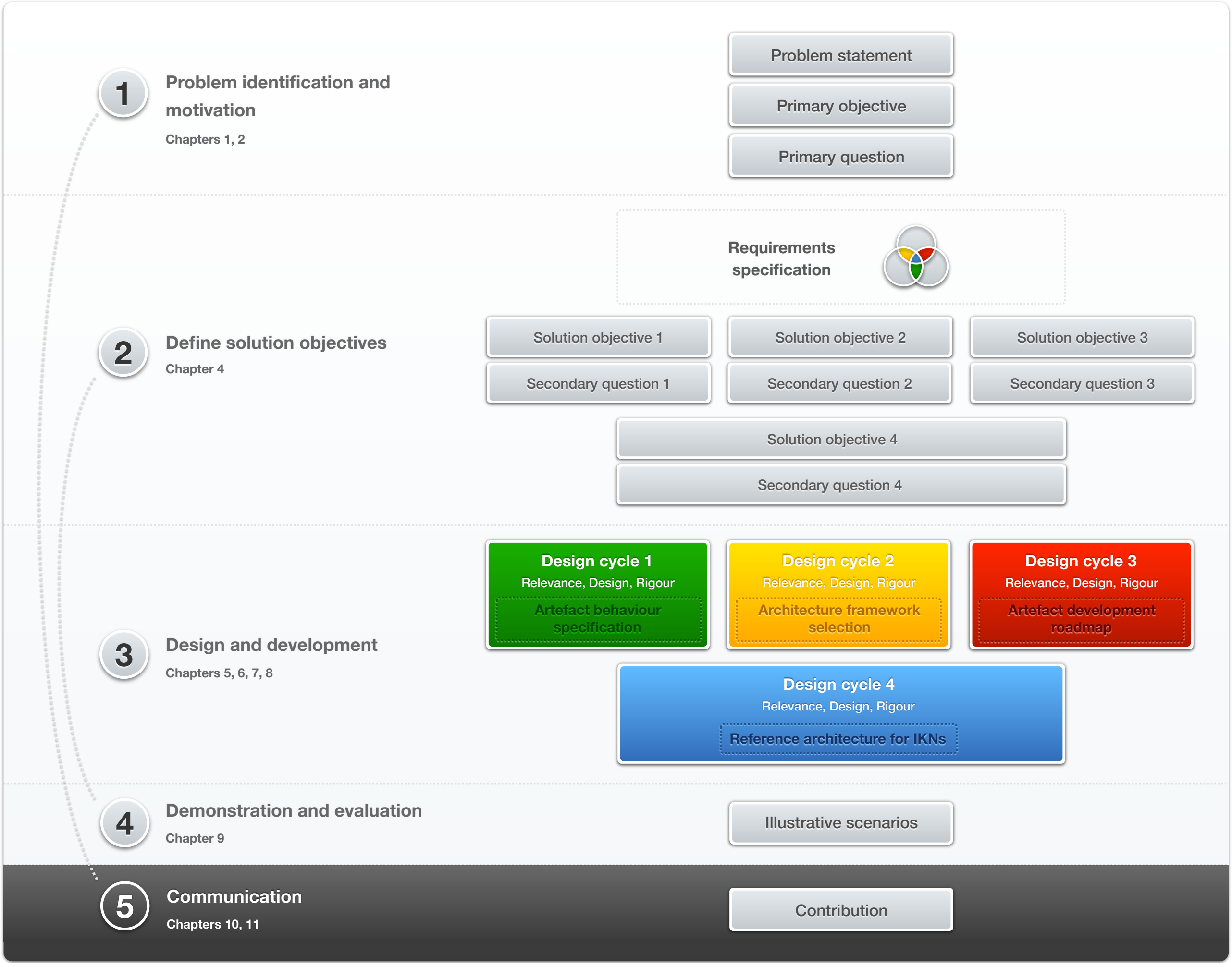
The suitability of the employed architecture framework to the engineering of IKNs was demonstrated and evaluated next. This placed the focus on the selection of the Zachman framework as a way of satisfying both the scenarios and technical requirements for the artefact (SO2, yellow, refer to chapter 6). Perspectives on this solution objective included the varied types of architectural information used to indicate the construction of IKNs, and the enterprise scope of this information. The ontological view on IKNs, as well as flexibility and reusability of parts were considered as qualities imparted by the Zachman framework. The range of stakeholders addressed the approach to current and future transformations and the model-based nature of the information included by the architecture framework was also considered. Consideration of these various aspects of the artefact's performance led to the conclusion that it can be motivated that the artefact *achieved SO2 by employing an architecture framework which is suitable to the engineering of IKNs*.

Lastly, the way in which the artefact implemented the selected architecture framework to enable the artefact to function as a reference architecture was demonstrated and evaluated. The emphasis was therefore on the way the Zachman framework was implemented in the artefact through execution of the artefact development roadmap to address the functional and technical requirements (SO3, red, refer to chapter 7). The success of the implementation of the Zachman framework was considered in terms of the meta-level, nature, scope and stakeholders of the reference architecture for IKNs. The type of information provided, approach to transformations, detail and representation were also demonstrated and evaluated. It was motivated that the artefact *achieved SO3 by implementing the selected architecture framework in a way that enables it to function as a reference architecture*.

The demonstration and evaluation of the reference architecture for IKNs through multiple illustrative scenarios have shown that it may be argued that the artefact achieves all four its solution objectives, as identified in Part 2 of the document. This version of the artefact is therefore seen as a candidate solution to the research problem, and no further iteration of the third activity in the research design (design and development) is required. The utility of the reference architecture for IKNs in solving the research problem, as well as its novelty and the rigour of its design, will be communicated in Part 5 of the document.

Part 5 – Communication

The fifth and final part of the document contains two chapters and presents the fifth activity of the research design employed in this study. It aims at communicating the artefact discussed in Parts 2, 3 and 4 as a solution to the problem that was identified and motivated in Part 1. The research contributions made by the study are highlighted in chapter 10, while chapter 11 concludes the document by providing an overview of the study. In this part of the document, particular attention is paid to the effective presentation of the research to both technology-oriented and management-oriented audiences.



10. Contribution

10.1 Introduction

This chapter discusses the research contributions made through the execution of the study. In the context of the pragmatic underpinning of the study, the chapter therefore highlights the newly generated knowledge contained in these research contributions, and which is now available to stimulate action and induce change (Goldkuhl 2012). The aim of this chapter is to argue the usefulness of this newly generated knowledge as a solution to the research problem which initiated the study, as well as the various research questions that were considered along the way.

One of the guidelines for the use of DSR is that studies that employ the method should produce clear and verifiable contributions in the areas of the design artefact, design foundations and/or the design methodologies (Hevner et al. 2004). The research contributions made by the study are therefore discussed on these grounds by considering the artefacts constructed in each design cycle, and the research design that was employed (refer to Figure 10.1). This is consistent with the notion of “learning through building’ that features prominently in the rationale for DSR” (refer to section 3.3.1).

The primary contribution of the study is a reference architecture for IKNs, which enhances the current state of IKN engineering and development. This artefact extends the knowledge base, but also applies existing knowledge through extensive use of existing constructs and models in the development of the reference models contained in the reference architecture. Both the practical and scientific implications of this primary contribution are discussed in section 10.2.

Various secondary contributions are highlighted in section 10.3. Section 10.3.1 discusses the contribution made by the artefact behaviour specification developed in the first (green) design cycle, with the architecture framework selection developed in the second (yellow) design cycle being considered in section 10.3.2. Section 10.3.3 discusses the contribution made by the artefact development roadmap from the third (red) design cycle, while section 10.3.4 discusses a visual representation format for the Zachman framework that also originated in the third cycle. Section 10.3.5 considers the contribution made in the form of a research design for DSR that was developed for and implemented in the study. After this section 10.4 concludes this chapter.

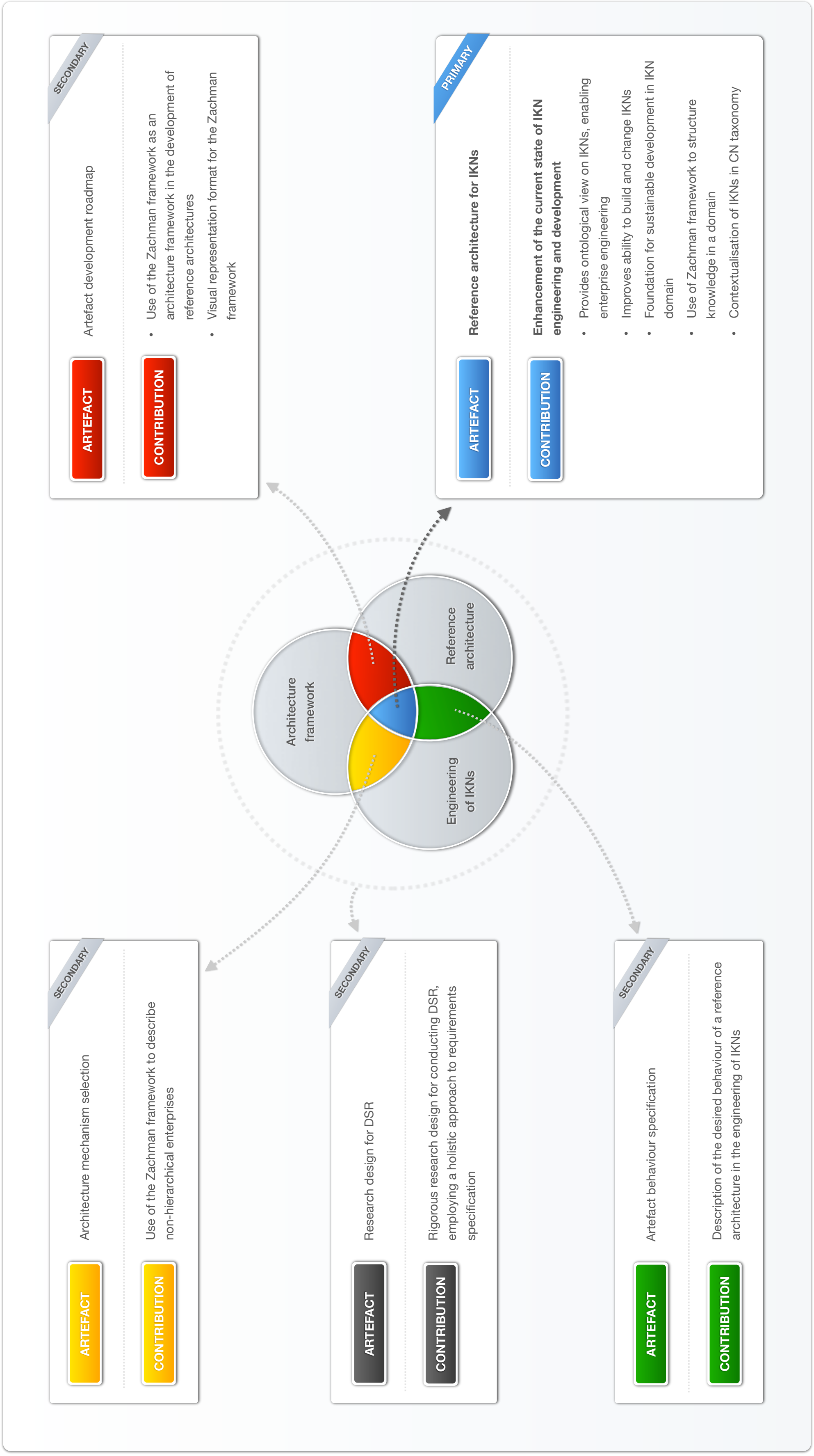


Figure 10.1: Overview of contributions

10.2 Primary contribution

The research problem that initiated this study is articulated as follows (refer to section 1.3):

No reference architecture exists for use in the engineering of IKNs.

The primary contribution of the study is a solution to this problem in the form of a **reference architecture for IKNs** that enhances the current state of IKN engineering and development (refer to Figure 10.2).

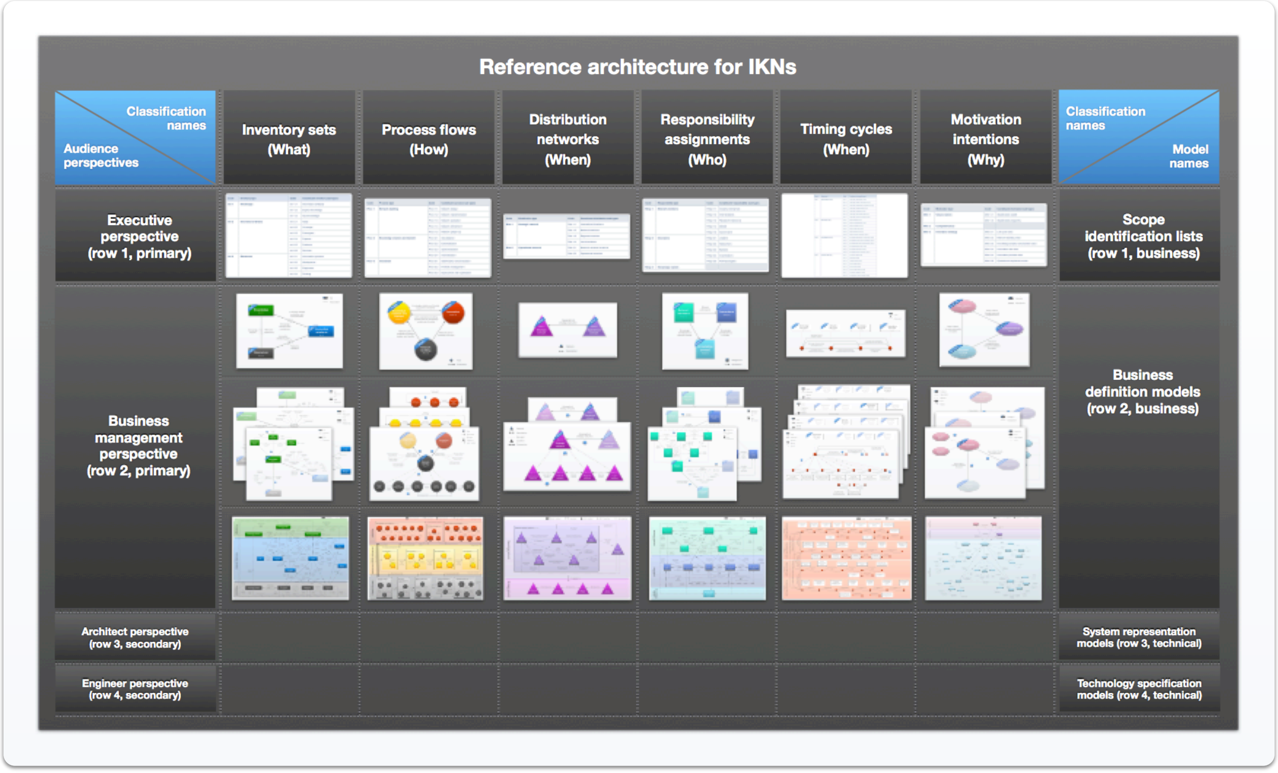


Figure 10.2: Primary contribution - Reference architecture for IKNs

The reference architecture for IKNs consists of a collection of reference models that provide inputs for the creation of particular models that describe specific IKNs. The reference models are structured according to the top two rows of the Zachman framework and describe IKNs from a strategic and executive perspective. This enables an ontological view on IKNs and supports the engineering of these networks. The way in which this artefact solves the research problem from both a practical and scientific point of view is accordingly discussed.

10.2.1 Practical implications

The development of the reference architecture for IKNs contributes a valuable resource to the enterprise engineering toolset of practitioners in the domain. By providing inputs for describing the construction of networks independent of their implementation, the reference architecture for IKNs facilitates the ontological view on these networks that had previously been lacking. This ontological view corresponds to a “white-box” model of IKNs, which improves the ability of practitioners to build and change networks (Dietz 2006). These models compliment the “black-box” models of IKNs that are currently facilitated by existing resources like the methodology for IKNs (Schutte 2010) and which focus on the operation and control of networks. A more detailed discussion of the scientific implications of the interactions between “white-box” and “black-box” models during the design and operation of IKNs is presented in section 10.2.2.

Networks that are *built* based on the inputs provided by the reference architecture for IKNs have a higher likelihood of successfully achieving their objectives, as their construction is based on sound constructional principles. Since the reference architecture provides starting points for the building of networks, practitioners also no longer need to construct IKNs without any building blocks to start with. The roll-out time of networks is therefore shortened, which increases the ability of IKNs to react to innovative opportunities in a timely manner. The mechanism through which the reference architecture for IKNs provides input in the building of new networks results in the development of explicit architecture descriptions, which in turn improves the replicability of successful networks.

In terms of *changing* networks, the reference architecture for IKNs provides reference models against which practitioners may benchmark the construction of existing networks. From there the reference architecture may be used to provide input in the design of a refined “to-be” network architecture description. The ontological view on IKNs that the reference architecture facilitates improves the ability of practitioners to comprehend the complex web of relationships between architecture components. This enables change management when designing and rolling out refined versions of the network architecture. The presence of descriptions of the network architecture is also a valuable resource in knowledge transfer regarding network activities to new network stakeholders during both network implementation and refinement.

The input that the reference architecture for IKNs provides for the development of particular network architecture descriptions are flexible and practitioners may customise particular models to more accurately describe a particular implementation.

In combination with the functional (“black-box”) views on IKNs, these networks may now be designed and operated as systems. This equates to managing the knowledge-creating organisation rather than knowledge itself, which is the correct approach given the humanistic aspects of knowledge (refer to section 2.3.1).

10.2.2 Theoretical implications

Moving toward extending the concept of EE to IKNs (and CNOs in general), this project takes a first step by contributing a reference architecture for IKNs that features strong constructional principles. This reference architecture complements existing resources that focus on contributing functional principles to the domain, e.g. the methodology for IKNs (Schutte 2010). Having both constructional (“white-box”) and functional (“black-box”) perspectives available, IKNs may now be comprehensively designed, operated, and, indeed, studied as systems.

The reference architecture for IKNs provides the foundation for more sustainable development in the IKN domain, since it completes a comprehensive systems view of this class of networks. It contributes an ontological view on IKNs that is independent of implementation; and since the behaviour of a system is brought about through its construction, it allows for the explanation and modelling of various complex behavioural traits of these networks. One example of this is that the management of IKNs to evolve from an explorative to an exploitative mode through the innovation life cycle (Chesbrough & Prencipe 2008) may now be described in constructional terms. It would be possible to create a composite model of an IKN based on the inputs from the reference architecture that shows the construction of a network which is managed in this way. This model would indicate the evolution of the innovation strategy, stakeholders, knowledge and innovation process focus of the network as the innovation life cycle progresses.

The modular approach of the Zachman framework allows for changes in a particular cell of the reference architecture for IKNs without other cells necessarily being impacted. Where relationships, i.e. horizontal integrations and/or vertical transformations, between the updated cell and other cells do exist, these relationships are clearly indicated. This allows for alternate or improved views of certain topics, e.g. knowledge, that form part of the reference architecture for IKNs, to be included in the description without invalidating the entire artefact. This enables further development of the reference architecture for IKNs as the understanding of these networks as social phenomena deepens.

A further theoretical implication of the contribution of the reference architecture for IKNs is that it is an example of the use of the Zachman framework to structure the body of knowledge in a certain domain. With its comprehensive abstractions and perspectives, and no bias toward any particular subject, the framework is a useful taxonomy within which to classify existing knowledge in a domain. This classification illuminates areas where gaps exist in the understanding of a domain or where the knowledge that does exist is not yet mature. Relationships between topics in a domain are also highlighted by the Zachman framework classification scheme, which helps to identify areas that warrant further investigation.

A final theoretical implication of the contribution of the reference architecture for IKNs, is an improved contextualisation of IKNs against the larger taxonomy of CNs. Simultaneous developments in the CN-domain in the time during which this study was conducted allowed for a much better understanding of the wider landscape of networks (Camarinha-Matos & Afsarmanesh 2012). This has enabled the classification of IKNs in the taxonomy of CNs as a hybrid network form that includes elements from both CNOs and ad-hoc collaborations. As such the reference architecture for IKNs utilised specialised elements from the ARCON reference model for CNOs (Camarinha-Matos & Afsarmanesh 2008a) as inputs for the reference models it contains.

10.3 Secondary contributions

In the first three design cycles included in the study, three interim versions of the reference architecture for IKNs were developed as part of the incremental development toward the final artefact. These interim versions were all packaged as artefacts in themselves and contain newly generated knowledge that contributes to the domain of IKNs. These artefacts, along with a visual representation format for the Zachman framework and a distillation of the methodology adopted, form the secondary contributions made by the study.

10.3.1 Artefact behaviour specification

The secondary research question considered in chapter 5 (SRQ1) highlighted a gap in the current understanding of IKNs by asking what the desired behaviour of a reference architecture in the engineering of IKNs is. As part of the design reasoning in the first design cycle, which addressed this question, the awareness was articulated that the behaviour of a reference architecture for IKNs should address the functional requirements of the network engineering process for such an artefact.

This awareness initiated an investigation into the relationships between the meta-relationships and IKN life cycle dimensions of the modelling framework for the reference architecture for IKNs. This investigation yielded an artefact behaviour specification as a design artefact that addresses the knowledge gap mentioned above (refer to Figure 10.3).

The artefact behaviour specification describes the role of an artefact that conveys constructional principles in the life cycle of IKNs. Therefore, it functionally describes the landscape of IKN engineering (refer to Figure 10.3). Since it does not include any specification of a preferred technical approach, the artefact behaviour specification is generic and may be exploited by artefacts other than the reference architecture for IKNs (refer to section 10.2).

The functional description of the IKN engineering landscape provided by the behaviour specification includes that the artefact should be free from implementation bias toward any particular network. It should furthermore be aimed at stakeholders that are involved in the design, refinement and phase-out phases of the IKN life cycle. A range of desired actions are specified and summarised into two functional modes:

- The provision of inputs for the design of particular architecture descriptions, and
- To assist in assessing an existing architecture description by serving as a benchmarking tool.

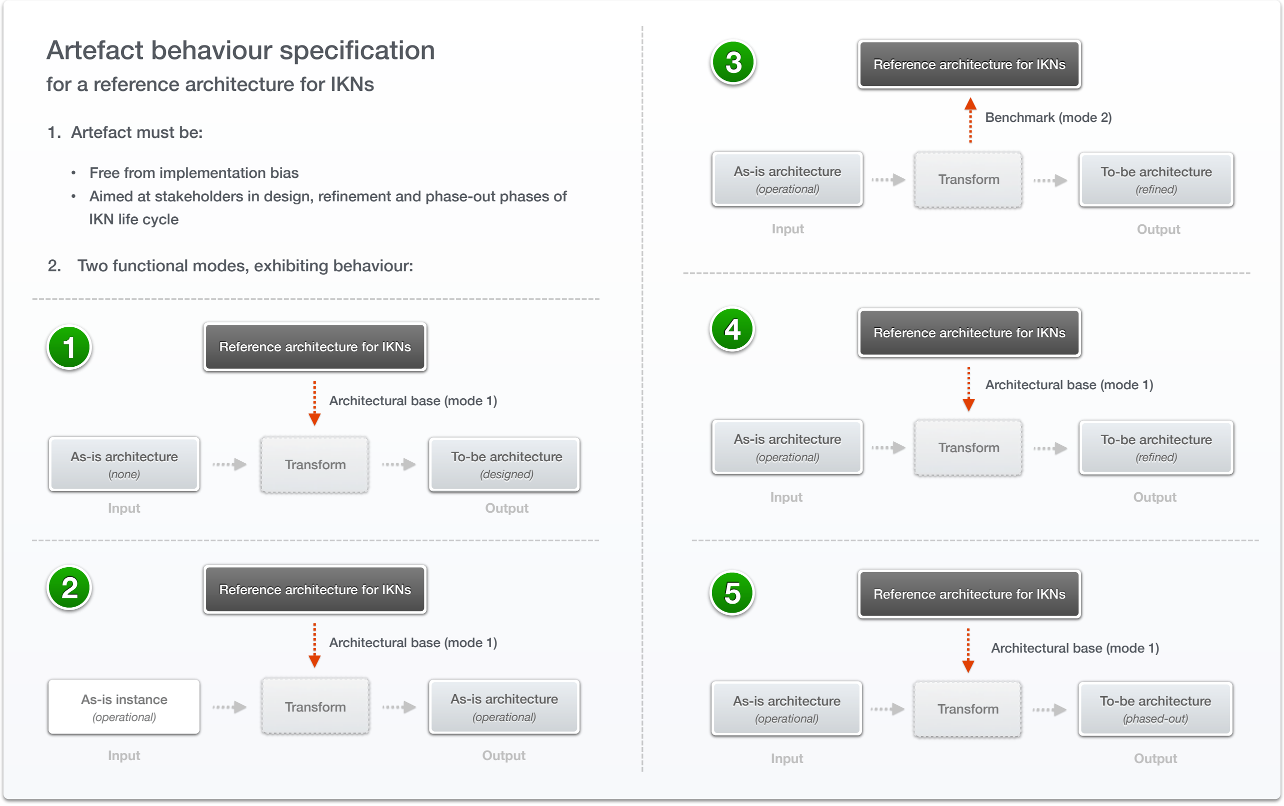


Figure 10.3: Secondary contribution – Artefact behaviour specification

The accuracy of the artefact behaviour specification in functionally describing the IKN engineering landscape was extensively demonstrated and evaluated in section 9.6.

10.3.2 Architecture framework selection

The secondary research question considered in chapter 6 (SRQ2) highlighted a further gap in the current understanding of IKNs by asking whether an architecture framework that is suitable to the engineering of IKNs is available. As part of the design reasoning in the second design cycle, which addressed this question, the awareness was articulated that not all architecture frameworks are suitable to the engineering of IKNs. It was furthermore noted that the requirements for suitability could be expressed in terms of architecture framework dimensions.

This awareness initiated an investigation into the relationships between the views axis of the modelling framework for the reference architecture for IKNs and the axis indicating the engineering of IKNs throughout their life cycle. This investigation yielded the selection of the Zachman framework as an architecture framework that is suitable to the engineering of IKNs and therefore addresses the knowledge gap mentioned above.

The Zachman framework is traditionally associated with hierarchical enterprises and a perception exists that it is not equipped to describe modern organisational forms such as collaborative networks. As part of the investigation into the requirements for suitability to the engineering of IKNs, a rigorous set of criteria was developed. Contrary to what may intuitively be expected by practitioners in the network architecture domain, the Zachman framework satisfied all these criteria.

The selection of the Zachman framework as the architecture framework employed in the reference architecture for IKNs was extensively demonstrated and evaluated in section 9.7.

10.3.3 Artefact development roadmap

The secondary research question considered in chapter 7 (SRQ3) highlighted a gap in the current understanding of reference architectures by asking how the Zachman framework can be implemented in a way that enables the artefact to function as a reference architecture. The Zachman framework is generally implemented in order to structure particular architecture descriptions rather than reference architectures, and a novel implementation of the framework therefore had to be developed.

The awareness of this knowledge gap initiated an investigation into the relationships between the views axis of the modelling framework and the axis indicating the meta-relationships between the artefact and its environment. This investigation yielded an artefact development roadmap that addresses the knowledge gap mentioned above (refer to Table 10.1).

Table 10.1: Secondary contribution – Artefact development roadmap

Dimension	Value	Zachman framework implementation
Initialisation phase		
1. Meta-level	Reference architecture	Basis for particular architecture descriptions
2. Nature	Reference models	Primitive reference models
3. Scope	Enterprise	Complete set (six columns)
4. Stakeholders	Primary (business) Secondary (technical)	Primary (rows 1 and 2) Secondary (rows 3 and 4)
5. Type	Business	Rows 1 and 2
6. Transformations	Versioned (current, future)	Multiple relevancies, transformation-agnostic
Population phase		
7. Detail	Varying levels	Multiple models in cells
8. Representation	Informal Semi-formal	Lists (row 1) Meta-models with semantic relationships (row 2)

This artefact development roadmap contributes a generalised approach to implementing the Zachman framework in order to create reference architectures. The roadmap specifies values for various architecture framework dimensions, as well as the interpretation of these values in the context of the Zachman framework.

The specification of values for the meta-level and nature dimensions fundamentally enables the production of a reference architecture. The specification of a value for the type dimension also warrants special mention, since it shows that potentially only the top two rows of the Zachman framework may be populated when creating a reference architecture. Although a case can be made for the population of the third row with reference models, the perspectives of the fourth and fifth rows are in principle too implementation-specific to accommodate reference models.

The ability of the artefact development roadmap to produce an artefact that functions as a reference architecture by implementing the Zachman framework was extensively demonstrated and evaluated in section 9.8.

10.3.4 Visual representation format

The seventh and eight steps of the artefact development roadmap discussed in the previous section (refer to Table 10.1) imply, amongst other things, the use of a semi-formal visual representation format for the Zachman framework that accommodates various levels of detail.

Apart from specifying the principles of primitive modelling and supplying basic meta-models for each abstraction, the Zachman framework is agnostic of any specific visual representation format. A common approach is therefore to attempt a strict application of the rules of primitive modelling while employing existing notations such as UML (Fatolahi & Shams 2006; Gerber et al. 2013). The success of this approach, however, is greatly dependent on the rigour with which the practitioner that develops the models adheres to the principles of primitive modelling, since these notations are in principle intended for composite (multi-variable) modelling. Since the study did not intend to inherit the uncertainty of such an approach, a novel visual representation format that adheres to the meta-models of the Zachman framework, while accommodating various levels of detail, was therefore required.

In order to achieve the representation format illustrated in Figure 10.4, the meta-models supplied by the Zachman framework were extended through the concepts of hyponomy, meronomy and holonomy from the domains of knowledge representation and linguistics. This representation format allows for primitive modelling and may be customised for the concept types present in each abstraction of the framework (inventory sets, process flows, distribution networks, responsibility assignments, timing cycles and motivation intentions). The representation format furthermore allows for the illustration of primitive associations between concepts, as well as the deconstruction of concepts into elements with detailed

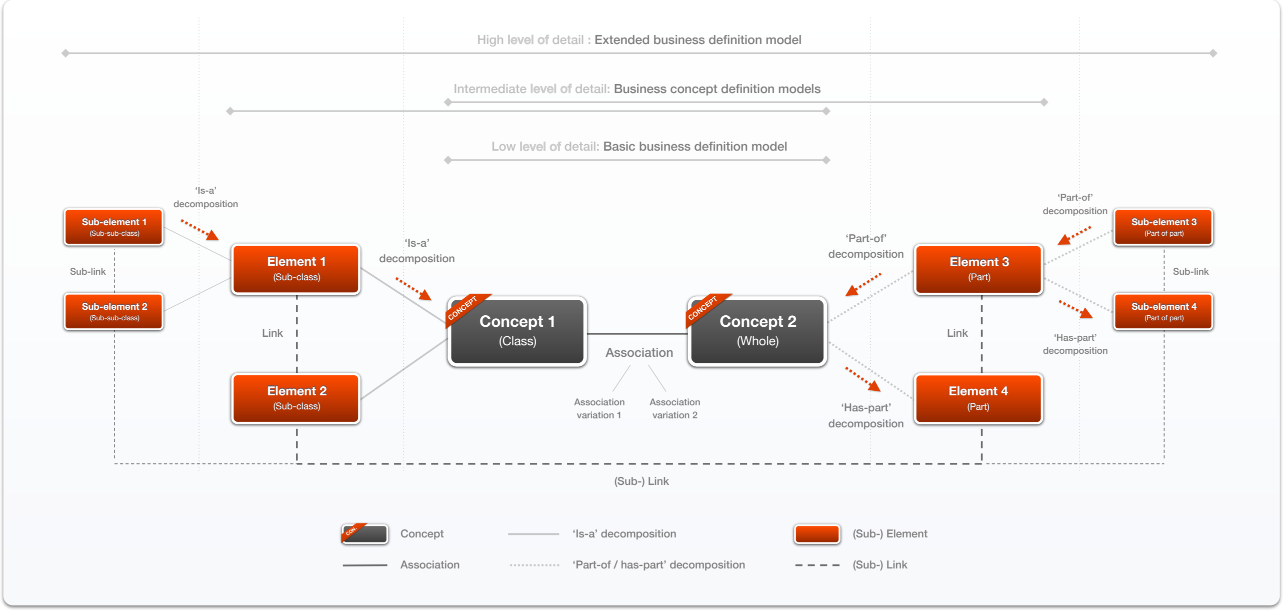


Figure 10.4: Secondary contribution – Visual representation format for the second row of the Zachman framework

primitive links amongst them. Three detail levels are enabled by the deconstructions indicated in the format, although it may be extended to include as many detail levels as are required.

The use of this visual representation format in the second row of the Zachman framework was extensively demonstrated and evaluated during the population of the reference architecture for IKNs with reference models, as well as in section 9.8. There is however no reason to believe that this representation format would not be suitable to other perspectives of the framework when adjusted to adhere to the respective meta-models. This visual representation format complements the contribution of the Zachman framework, as a tool to describe the knowledge available in a given domain (refer to section 10.2).

10.3.5 Research design for DSR

The domain of enterprise engineering is a multi-paradigmatic one that is characterised by both quantitative and qualitative research. It is therefore imperative that researchers in this domain are explicit about the philosophical paradigms adopted in their research. Most studies within the enterprise engineering domain, however, are initiated by problems, which makes pragmatism a likely candidate for a suitable paradigmatic choice.

Among the qualitative studies in enterprise engineering, the vast majority attempt to solve the problems that initiated the studies through the design of some kind of artificial artefact (Simon 1996) that solves the problem. It is furthermore often the case that incomplete information about the problem and the nature of the envisaged solution is available at the inception of the study. This results in DSR being an apt choice as research method in EE. During the course of this study it has indeed been observed that several studies conducted in this field in the Enterprise Engineering research group at the Department of Industrial Engineering, Stellenbosch University, employ DSR, although mostly implicitly. Researchers involved in these studies often attempt to implement hypothesis-driven research methodologies, while a DSR-based methodology based in pragmatism would have resulted in more rigorous, communicable and elegant research. Furthermore, in a recent issue of the journal Business and Systems Engineering, it is noted that the number of DSR papers presented at doctoral consortia and information systems conferences is comparatively small (Bichler 2014). We argue that this lack of DSR being explicitly conducted and published is in part due to difficulties in coming up with practical research designs for conducting DSR.

In this study the DSRM presented by Peffers et al. (2008) was employed as the basis of a research design that produced an artefact that solves the problem that initiated the study. A minor alteration was made to the DSRM by combining the fourth and fifth activities, namely Demonstration and Evaluation, to arrive at a five-step research design (refer to Figure 10.5). In this research design the fifth step communicates the solution to the problem that was identified and motivated in the first step. The fourth step demonstrates how the artefact attempts to achieve the solution objectives that were defined in the second step, and evaluates its success in doing so.

The major methodological contribution of this study occurs in the second activity of the research design in which solution objectives for the artefact are defined based on the problem description. In this study, an approach to requirements specification that originated in the domain of industrial design was employed in order to derive rational and complete solution objectives for the artefact. This approach describes a requirements specification frame of reference that consists of functional requirements, technical requirements and scenarios (Miedema et al. 2007). By considering the various intersections between these different requirement types, it is possible to derive comprehensive solution objectives for the artefact.

The sequential consideration of requirements intersections relates to the generic system development process (not to be confused with the use of the same process to describe the research problem addressed in this study) (refer to Figure 10.6). In this sense, the artefact relates to the “object system” described in the generic system development process.

By considering the intersection of functional requirements and scenarios, a “black-box” solution objective for the desired artefact is defined that describes its desired behaviour (refer to Figure 10.6) (1). This is followed by considering the intersection of technical requirements and scenarios (2). This results in the definition of a “white-box” solution objective for the artefact that indicates how its construction is applicable to the scenarios in which it will be required to operate, i.e. its “using system”. This constructional solution objective is extended in the intersection of functional and technical requirements, where a solution objective is defined that requires the construction of the artefact to bring about its desired behaviour (3). All three requirement specification types are then considered simultaneously to define a solution objective that

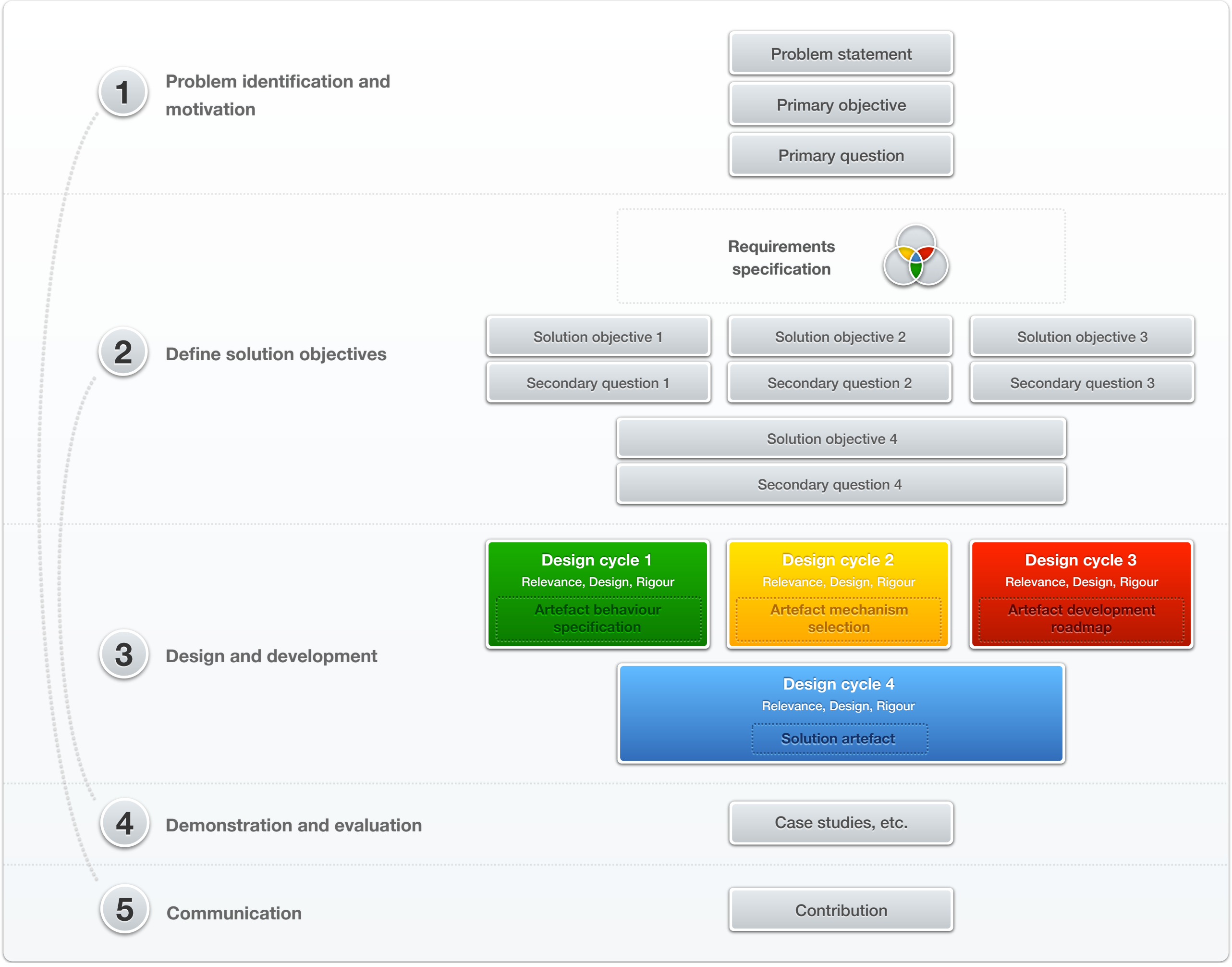


Figure 10.5: Secondary contribution – Research design for DSR

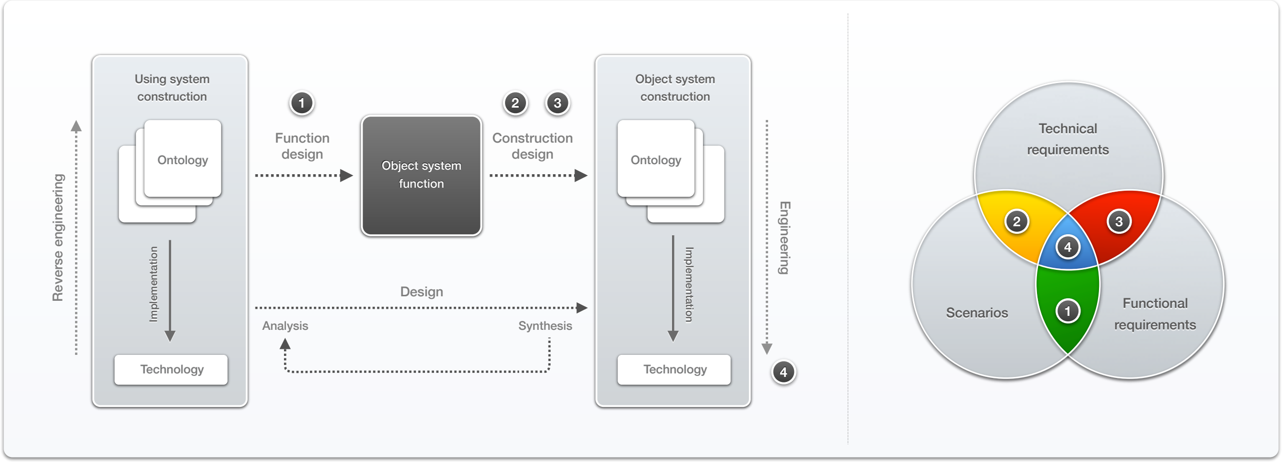


Figure 10.6: Perspectives on requirements specification

combines the other three solution objectives (4). Since it has a holistic view on the requirements of the artefact that is to be designed, this final solution objective is an operationalised version of the primary research objective and leads to an instantiation of the solution artefact. Finally, four secondary research questions are defined that each addresses one of the defined solution objective.

In the third step of the research design (refer to Figure 10.5), these four solution objectives with their accompanying secondary research questions are employed as the drivers of four sequential design research cycles that iteratively design and develop the required solution artefact (refer to Figure 10.7). The first three cycles represent the design of the artefact in order to achieve its solution objectives (artefact versions 1.1, 1.2 and 1.3), with the fourth cycle representing the development and instantiation of the eventual solution (artefact version 2).

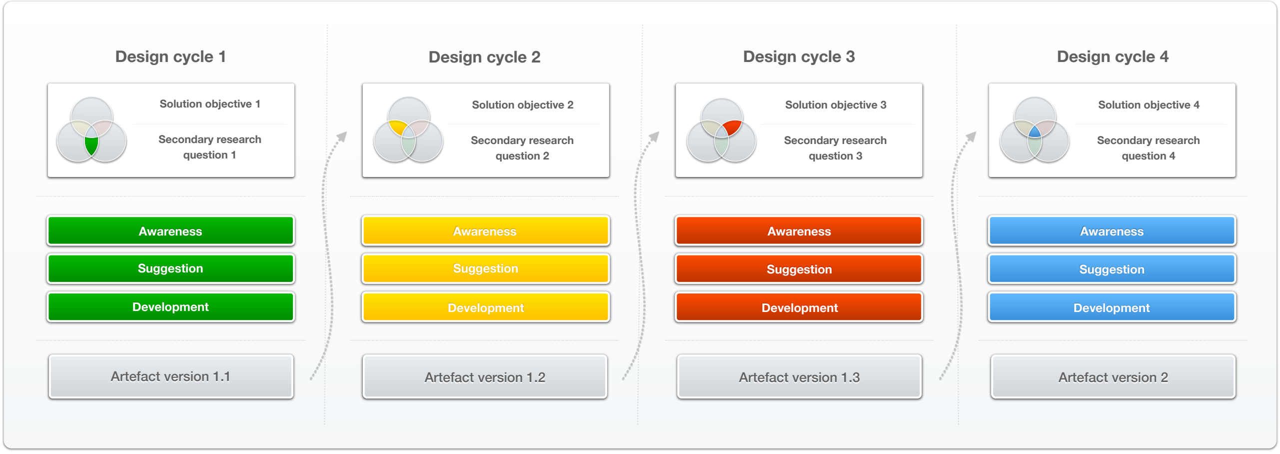


Figure 10.7: Drivers and reasoning in design cycles

In accordance with the drivers of each design cycle, the first cycle (green) produces a “black-box” model of the artefact that researches and designs its desired behaviour. This version of the artefact may generically be named the *artefact behaviour specification*. The artefact behaviour specification is complemented in the second cycle (yellow) by a “white-box” model that researches and designs the construction of the artefact in basic terms by essentially selecting technologies for the construction of the artefact. This version of the artefact may therefore be referred to as the *artefact mechanism selection*. The third cycle (red) extends this “white-box” model by designing the artefact in detail in order for its construction, using the selected technologies, to bring about its required behaviour. This detailed design may be packaged in terms of an *artefact development roadmap*. This iterative design work culminates in the fourth cycle (blue) where the *solution artefact* is instantiated according to the design work done in the previous cycles.

In each of these cycles design reasoning is employed, including an awareness, suggestion, development and potentially an evaluation step. These evaluation activities, however, may also be combined and performed in the demonstration and evaluation activity of the research design (refer to Figure 10.5). In each cycle the development of an iterative version of the artefact leads to the generation of knowledge regarding both the problem and the required solution. This knowledge that is generated through “learning through building” is used to update the requirements specification frame of reference. This results in a better understanding of how to progress in subsequent cycles to eventually arrive at an artefact that achieves its defined solution objectives.

This approach is not specific to this study and is therefore seen as a generic contribution to the use of DSR in enterprise engineering. The research design was extensively demonstrated and evaluated during its implementation to design and develop the reference architecture for IKNs in this study.

10.4 Conclusion

This chapter presented an overview of newly generated knowledge contained in the research contributions made through the course of the study. Throughout the discussion, the case was made for the usefulness of this newly generated knowledge to solve the research problem and various research questions. The study makes clear and verifiable contributions in the form of primary and secondary artefacts, as well as a methodology. These contributions extend the body of knowledge in the domains of IKNs and enterprise engineering as described in chapter 2.

The primary contribution of the study is a reference architecture for IKNs that solves the research problem that initiated the project (refer to Part 1) by greatly enhancing the current state of IKN engineering and development. This artefact extends the knowledge base and applies existing knowledge through extensive use of existing constructs and models. Both the practical and scientific implications of this primary contribution were discussed. The various secondary contributions of the study were also discussed. These included the artefact behaviour specification, architecture framework selection and artefact development roadmap that were developed during the first three design cycles. A visual representation format for the Zachman framework was discussed, as well as a methodological contribution to the design and development of an artefact in enterprise engineering.

Chapter 11 completes the communication of the research contribution by providing an overview of the study to conclude this document.

11. Conclusion

11.1 Introduction

The aim of this chapter is to provide an overview of the findings of the study. Section 11.2 presents a summary of the findings of the study, structured according to the research design that was employed. Section 11.3 presents opportunities for further research, and the document concludes with final reflections in section 11.4.

11.2 Summary of findings

11.2.1 Problem identification and motivation

This section provides a summary of the problem identification and motivation discussed in Part 1 of the document. A summary of the research methodology that was adopted by the study is also provided.

Three *emerging disciplines* currently contribute towards enterprise design and alignment with the aim of enabling enterprises to adapt to the challenges of the Information Age. These are EE (enterprise engineering), EA (enterprise architecture) and EO (enterprise ontology). Although a number of publications exist for EE and EA, there is a lack of shared meaning in terms of the *theoretical foundations*, *definitions* and *business benefits* of these approaches (Kappelman 2011; Lapalme 2012). This creates significant challenges in searching for relevant literature and advancing the EE and EA disciplines. This study has however adopted the definitions for EE and EA that are stated in sections 2.6 and 2.7. In short, EA is regarded as the set of descriptive representations of the enterprise, while EE relates to the intentional design and evolution of the enterprise between various desired states. Both of these disciplines are seen as crucial enablers of growth in modern enterprises, allowing them to adapt to their dynamic environment (Kappelman 2011; Lapalme 2012).

It is furthermore widely recognised that the ability to innovate and generate new knowledge through the development of new products, services and processes is another key factor in the survival of enterprises in the Information Age (Krogh et al. 2001; Drucker 2007). The latest trends in innovation management however show that the innovation process is no longer one that is executed inside a single enterprise, giving rise to the development of inter-organisational innovation networks (Powell et al. 1996). The importance of knowledge as a dynamic enabler of this networked innovation approach is furthermore highlighted (Seufert et al. 1999). This has lead to the emergence of IKNs in which knowledge is created and shared between network stakeholders in order to foster sustainable innovation (Du Preez et al. 2008a).

The increasing rate of change associated with the modern economy means that enterprises, including IKNs, have progressively less time to react to market changes and opportunities (Kappelman & Zachman 2013). The focus therefore shifts to the potential of EE as a tool to adapt to the dynamic landscape of the Information Age (Dietz et al. 2013). The focus of this study is on extending the discipline of EE to IKNs through the development of a reference architecture.

When EE is considered from a systems theory point of view, both functional and constructional principles are required. The engineering of IKNs is however currently hampered by the lack of a reference architecture that provides constructional principles. This led to the *identification and motivation* of the following research problem that was considered in the study:

No reference architecture exists for use in the engineering of IKNs.

The study is rooted in pragmatism, with its mission to bring about action and change (Goldkuhl 2012). The research problem stated above was therefore addressed by a research objective (RO) that aims to change reality surrounding the research problem (refer to Table 11.1). It is accompanied by a primary research question (PRQ) that directs the focussed investigation in the study.

Table 11.1: Research objective and primary research question

Code	Research objective	Code	Research question
RO	To develop a reference architecture for use in the engineering of IKNs.	PRQ	How can a reference architecture for use in the engineering of IKNs be developed?

A constructivist research methodology was adopted for the production of a reference architecture for IKNs in order to answer the primary research question and thereby achieve the research objective (refer to Figure 11.1). This research methodology employs *design science research* (DSR) and is based on the DSRM presented by Peffers et al. (2004). The aim of the methodology was to produce an artefact in the form of a reference architecture for IKNs, as that would solve the identified research problem.

This first activity in the research design employed in the study was indeed to identify and motivate the research problem, as was summarised in this section. It corresponds to the fifth activity in the research design (refer to section 11.2.5) in which the success, utility and value of the solution to the problem was communicated.

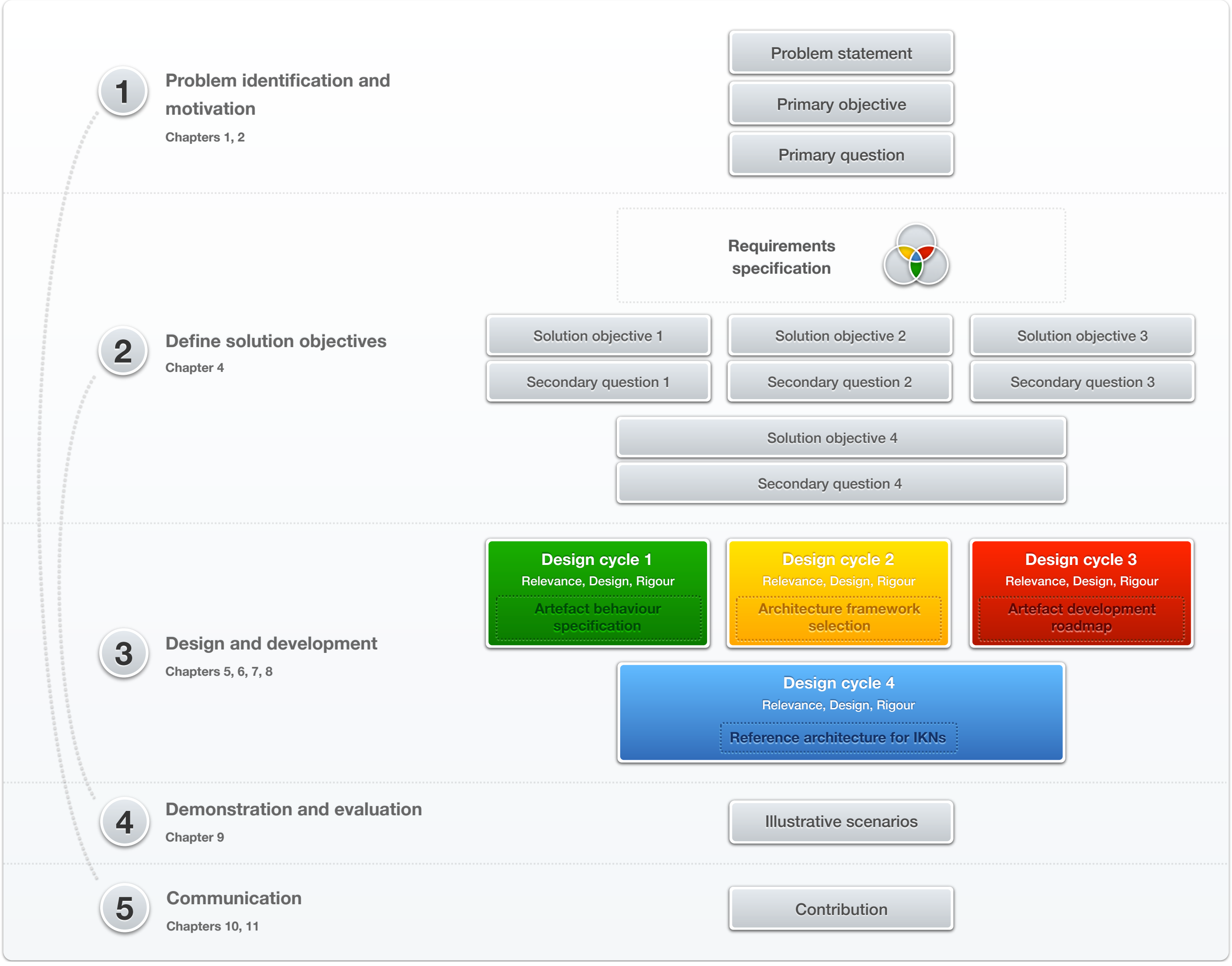
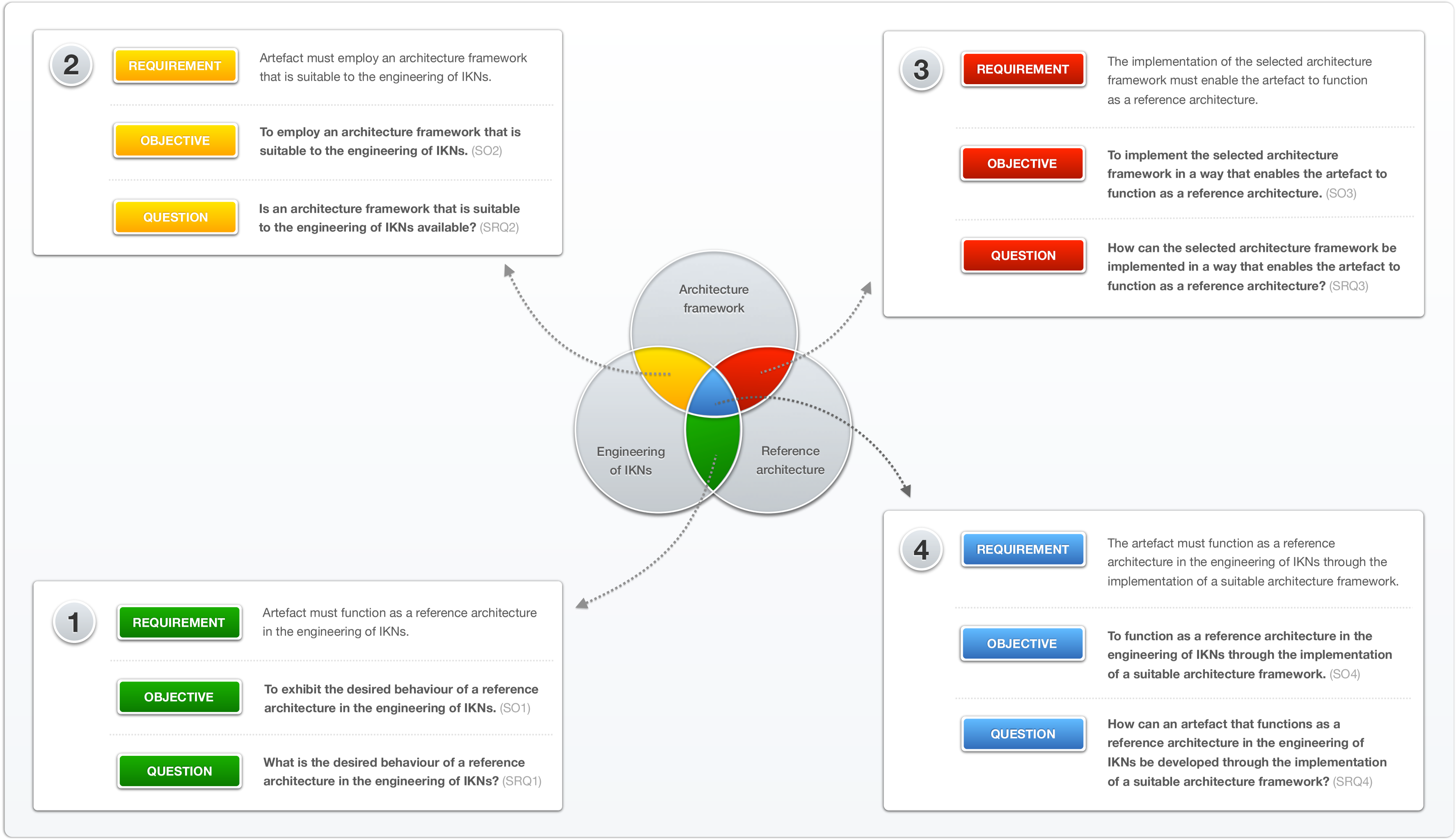


Figure 11.1: Research design

11.2.2 Definition of solution objectives

This section provides a summary of the solution objectives that were defined in Part 2 of the document.

In the second activity of the research design employed in this study, solution objectives were defined for an artefact that solves the identified research problem. These solution objectives were inferred rationally through a structured approach to requirements specification. This approach considered functional requirements, technical requirements and implementation scenarios for the solution artefact (refer to Figure 11.2). These different requirement types were combined to form compound requirement perspectives, which were in turn translated into solution objectives with accompanying secondary research questions. These secondary research questions directed the search process toward the design of an artefact that achieves the solution objectives. This second activity corresponds to the fourth activity in the research design (refer to section 11.2.4) in which the solution's ability to achieve the objectives from this activity is demonstrated and evaluated.



1

REQUIREMENT: Artefact must function as a reference architecture in the engineering of IKNs.

OBJECTIVE: **To exhibit the desired behaviour of a reference architecture in the engineering of IKNs.** (SO1)

QUESTION: **What is the desired behaviour of a reference architecture in the engineering of IKNs?** (SRQ1)

2

REQUIREMENT: Artefact must employ an architecture framework that is suitable to the engineering of IKNs.

OBJECTIVE: **To employ an architecture framework that is suitable to the engineering of IKNs.** (SO2)

QUESTION: **Is an architecture framework that is suitable to the engineering of IKNs available?** (SRQ2)

3

REQUIREMENT: The implementation of the selected architecture framework must enable the artefact to function as a reference architecture.

OBJECTIVE: **To implement the selected architecture framework in a way that enables the artefact to function as a reference architecture.** (SO3)

QUESTION: **How can the selected architecture framework be implemented in a way that enables the artefact to function as a reference architecture?** (SRQ3)

4

REQUIREMENT: The artefact must function as a reference architecture in the engineering of IKNs through the implementation of a suitable architecture framework.

OBJECTIVE: **To function as a reference architecture in the engineering of IKNs through the implementation of a suitable architecture framework.** (SO4)

QUESTION: **How can an artefact that functions as a reference architecture in the engineering of IKNs be developed through the implementation of a suitable architecture framework?** (SRQ4)

Figure 11.2: Compound requirements, solution objectives and secondary research questions

11.2.3 Design and development

This section provides a summary of the design and development of the reference architecture for IKNs, as discussed in Part 3 of the document.

The third activity in the research design employed in the study discussed the design and development of the reference architecture for IKNs. This design and development process incrementally develops the artefact in four design cycles. Each design cycle focuses on a single solution objective, along with its associated secondary research question, as defined in the second activity (refer to section 11.2.2).

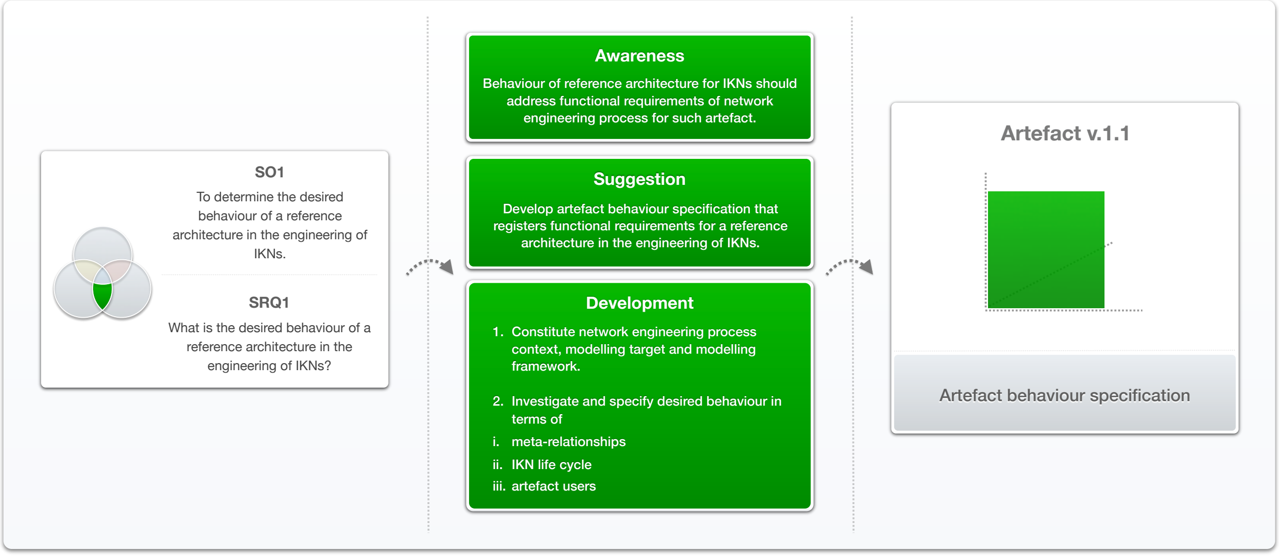
11.2.3.1 First design cycle

The first design cycle was presented in chapter 5, and considered the compound requirement perspective that combines the scenarios and the functional requirement (refer to Figure 11.3). The design cycle therefore addressed the first solution objective (SO1):

> **To determine the desired behaviour of a reference architecture in the engineering of IKNs.**

The search process in the cycle was directed by the first secondary research question (SRQ1):

> **What is the desired behaviour of a reference architecture in the engineering of IKNs?**



SO1
To determine the desired behaviour of a reference architecture in the engineering of IKNs.

SRQ1
What is the desired behaviour of a reference architecture in the engineering of IKNs?

Awareness
Behaviour of reference architecture for IKNs should address functional requirements of network engineering process for such artefact.

Suggestion
Develop artefact behaviour specification that registers functional requirements for a reference architecture in the engineering of IKNs.

Development
1. Constitute network engineering process context, modelling target and modelling framework.
2. Investigate and specify desired behaviour in terms of
   - i. meta-relationships
   - ii. IKN life cycle
   - iii. artefact users

Artefact v.1.1
Artefact behaviour specification

Figure 11.3: Summary of first design cycle

The *awareness* of the problem for the cycle outlined that the behaviour of a reference architecture for IKNs should address the functional requirements of the network engineering process for such an artefact. Subsequently, the *suggestion* was made that an artefact behaviour specification should be developed to register the functional requirements for a reference architecture in the engineering of IKNs. The *development* of this artefact behaviour specification was initiated through the identification of the enterprise engineering process as a point of departure, specifying the typical IKN as the modelling target. A modelling framework for the reference architecture for IKNs was furthermore constituted, including axes that describe the meta-relationships between architectural artefacts, the IKN life cycle, and various architectural views. The artefact behaviour specification was then populated through the investigation and specification of the desired behaviour of the reference architecture for IKNs in terms of meta-relationships between architectural artefacts throughout the life cycle of IKNs. The likely users of the artefact were also taken into account in the final artefact behaviour specification.

The artefact behaviour specification contributed by this cycle represents the *relationships between the meta-relationships and IKN life cycle dimensions of the modelling framework*, as illustrated by the green plane to the right of Figure 11.3. This artefact behaviour specification is seen as the first version of the reference architecture for IKNs (v.1.1). It answers SRQ1 and achieves SO1 by determining the *desired behaviour of a reference architecture in the engineering of IKNs*, and the first design cycle was therefore concluded.

The generated knowledge about the final artefact, gained through design decisions and the development of a first version, could now be used to update the requirement frame of reference and led to the initiation of a next design cycle.

11.2.3.2 Second design cycle

The second design cycle was presented in chapter 6, and considered the compound requirement perspective that combines the technical requirement and scenarios (refer to Figure 11.4). The design cycle addressed the second solution objective (SO2):

> **To employ an architecture framework that is suitable to the engineering of IKNs.**

The investigation in this cycle was directed by the second secondary research question (SRQ2):

> **Is an architecture framework that is suitable to the engineering of IKNs available?**

The *awareness* of the problem for the cycle outlined that not all architecture frameworks are suitable to engineering of IKNs, and that the requirements for suitability could be expressed in terms of architecture framework dimensions. The *suggestion* was subsequently made that the Zachman framework should be investigated and evaluated for suitability to the engineering of IKNs, as expressed earlier as part of the problem awareness.

The *development* of the architecture framework selection was done through the investigation and evaluation of the Zachman framework in terms of the dimensions used to describe the requirements for

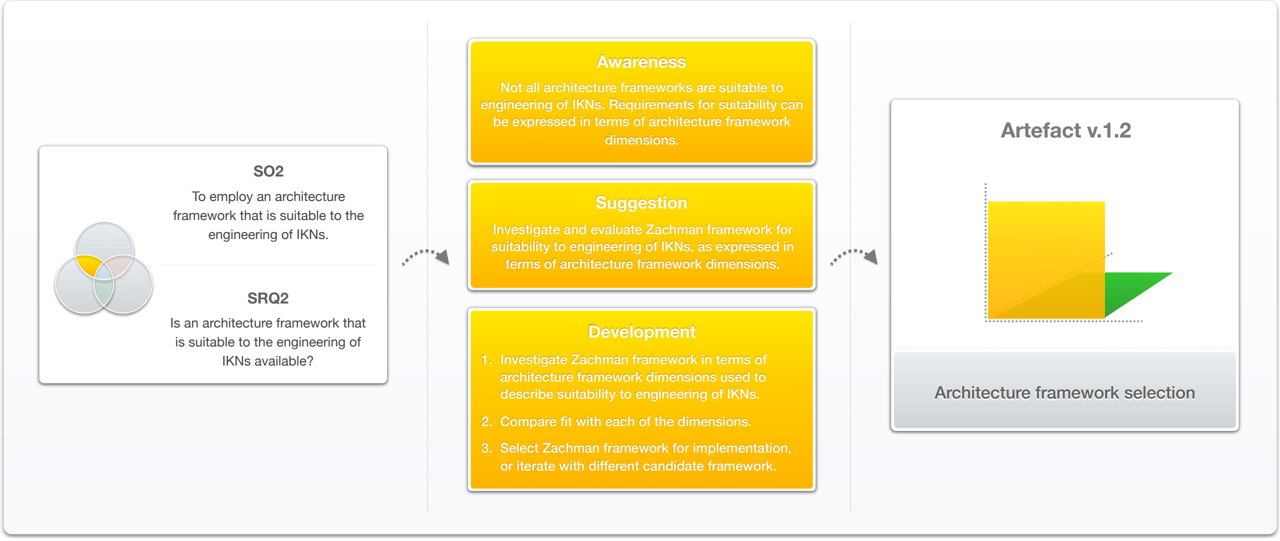


Figure 11.4: Summary of second design cycle

suitability to the engineering of IKNs. The Zachman framework and the architectural information it describes were therefore investigated in terms of type and scope of information, quality attributes, stakeholders, transformations and nature. For each of these dimensions, the level to which the framework satisfies the requirements was evaluated. As the Zachman framework satisfied all the requirements, it was selected as the architecture framework to be employed by the reference architecture for IKNs.

This implied that the architectural views provided by the Zachman framework, as manifested through a number of reified perspectives, should *populate the architectural views axis of the modelling framework embedded in the reference architecture for IKNs*. The search for an architecture framework that satisfies the requirements for the engineering of IKNs represented the investigation of the *relationships between the views axis and the axis indicating the engineering of IKNs throughout their life cycle* (refer to the yellow plane to the right of Figure 11.4).

The architecture framework selection contributed by this chapter is seen as the second version of the reference architecture for IKNs (v.1.2). It answers SRQ2 and achieves SO2 *by identifying an architecture framework that is suitable to the engineering of IKNs*, and the second design cycle was therefore concluded.

The generated knowledge about the final artefact, gained through design decisions and the development of a second version, could now be used to update the requirement frame of reference and led to the initiation of a next design cycle.

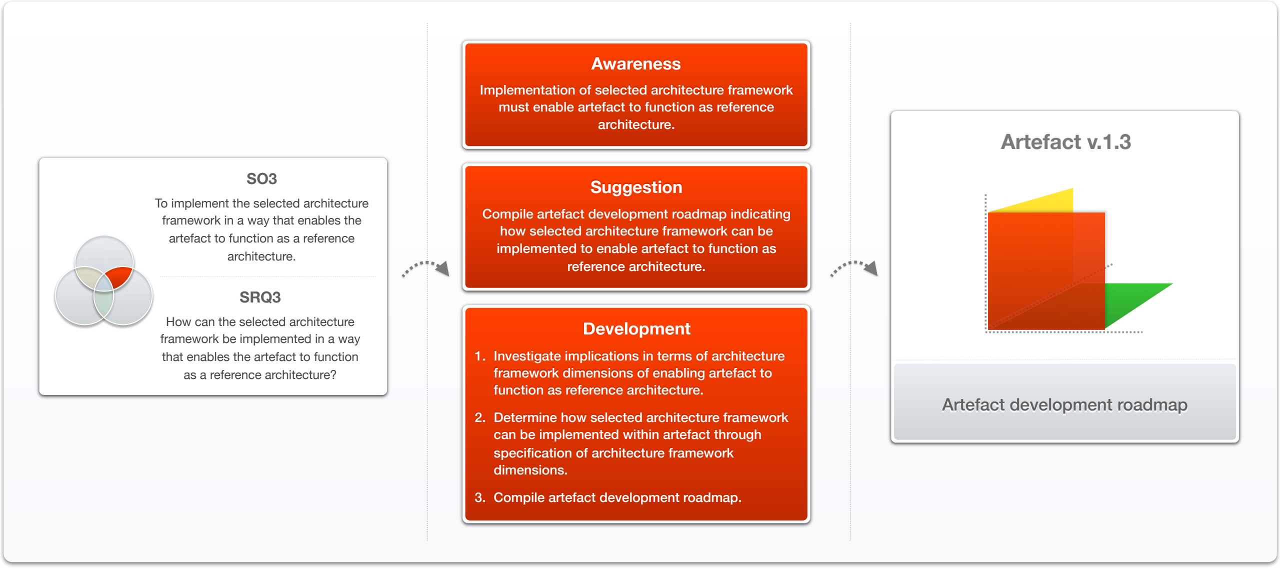
11.2.3.3 Third design cycle

The third cycle was presented in chapter 7, and considered the compound requirement perspective that combines the technical and functional requirements (refer to Figure 11.5). The design cycle addressed the third solution objective (SO3):

> **To implement the selected architecture framework in a way that enables the artefact to function as a reference architecture.**

The investigation in the cycle was directed by the third secondary research question (SRQ3):

> **How can the selected architecture framework be implemented in a way that enables the artefact to function as a reference architecture?**



SO3
To implement the selected architecture framework in a way that enables the artefact to function as a reference architecture.

SRQ3
How can the selected architecture framework be implemented in a way that enables the artefact to function as a reference architecture?

Awareness
Implementation of selected architecture framework must enable artefact to function as reference architecture.

Suggestion
Compile artefact development roadmap indicating how selected architecture framework can be implemented to enable artefact to function as reference architecture.

Development
1. Investigate implications in terms of architecture framework dimensions of enabling artefact to function as reference architecture.
2. Determine how selected architecture framework can be implemented within artefact through specification of architecture framework dimensions.
3. Compile artefact development roadmap.

Figure 11.5: Summary of third design cycle

The *awareness* of the problem for the cycle echoed the solution objective and secondary research question above by stating that the implementation of the Zachman framework must enable the artefact to function as a reference architecture. The *suggestion* was subsequently made to compile an artefact development roadmap that documents how this required implementation of the Zachman framework can be achieved. The *development* of the artefact development roadmap was done through the assignment of appropriate values to relevant architecture framework dimensions. The dimensions were that of *meta-level*, *nature*, *scope*, *stakeholders*, *type*, *transformations*, *detail* and *representation*. These values combine to articulate the required implementation of the Zachman framework from different points of view. The roadmap is divided into two phases, namely an initialisation phase in which an artefact skeleton is constructed, and a population phase during which this skeleton is populated with content.

The search for an implementation of the Zachman framework that enables the artefact to function as a reference architecture represented the investigation of the *relationships between the views axis of the modelling framework and the axis indicating the meta-relationships between the artefact and its environment.* This is illustrated with the red plane of the modelling framework to the right of Figure 11.5.

The artefact development roadmap contributed by this cycle answers SRQ3 and achieves SO3 by describing *an implementation of the Zachman framework that enables the artefact to function as a reference architecture*, and the second cycle was therefore concluded. The artefact development roadmap represents a third version of the reference architecture for IKNs (v.1.3).

The generated knowledge about the final artefact, gained through design decisions and packaged as an artefact development roadmap could now be used to update the requirement frame of reference and led to the initiation of a final design cycle.

11.2.3.4 Fourth design cycle

The fourth design cycle was presented in chapter 8, and considered the compound requirement perspective that combines the technical and functional requirements, as well as scenarios (refer to Figure 11.6). The cycle addressed the fourth solution objective (SO4):

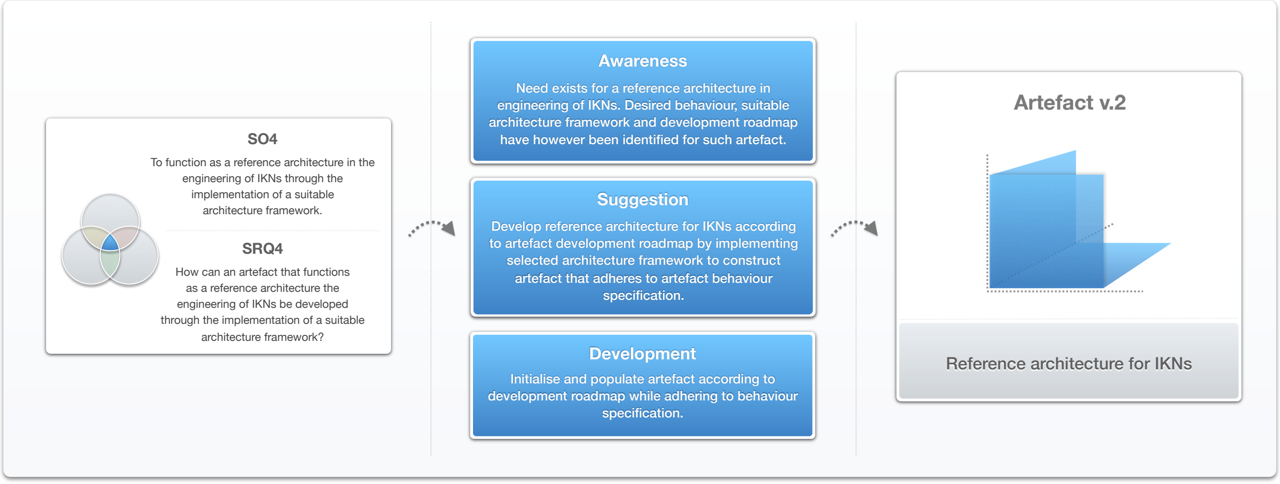
> **To function as a reference architecture in the engineering of IKNs through the implementation of a suitable architecture framework.**

The search process in this cycle was directed by the fourth secondary research question (SRQ4):

> **How can an artefact that functions as a reference architecture in the engineering of IKNs be developed through the implementation of a suitable architecture framework?**

The *awareness* of the problem for the cycle stated that a need exists for a reference architecture in the engineering of IKNs. The desired behaviour of such an artefact (refer to chapter 5), a suitable architecture framework to be employed (refer to chapter 6) and a roadmap for its development (refer to chapter 7) had however previously been identified. The *suggestion* was subsequently made to develop the reference architecture for IKNs according to the artefact development roadmap by implementing the selected architecture framework to construct an artefact that adheres to the behaviour specification.

The *development* of the reference architecture for IKNs was therefore done through initialising and populating the artefact according to the development roadmap, while adhering to the behaviour specification.



SO4

To function as a reference architecture in the engineering of IKNs through the implementation of a suitable architecture framework.

SRQ4

How can an artefact that functions as a reference architecture the engineering of IKNs be developed through the implementation of a suitable architecture framework?

Awareness

Need exists for a reference architecture in engineering of IKNs. Desired behaviour, suitable architecture framework and development roadmap have however been identified for such artefact.

Suggestion

Develop reference architecture for IKNs according to artefact development roadmap by implementing selected architecture framework to construct artefact that adheres to artefact behaviour specification.

Development

Initialise and populate artefact according to development roadmap while adhering to behaviour specification.

Artefact v.2

Reference architecture for IKNs

Figure 11.6: Summary of fourth design cycle

According to the development roadmap, the construction of the reference architecture for IKNs was divided into two phases, namely an *initialisation* phase in which the artefact skeleton was constructed, and a *population* phase during which this skeleton was populated with content. In both of these phases, the artefact was constructed in adherence to the artefact behaviour specification.

In the initialisation phase, the artefact skeleton was specified to function as *reference architecture* containing *primitive reference models*. The artefact skeleton was subsequently set up to view IKNs as *enterprises*, by including all six abstractions (columns) provided by the Zachman framework. Primary and secondary *audiences* were specified for the artefact, along with a delineation of the artefact population to the top two rows according to the *type* of information included. The primitive reference models were specified to be *transformation-agnostic,* therefore serving as the basis for the development of versioned particular models that describe either *current* or *future* transformations.

In the artefact population phase of the development roadmap, the artefact skeleton was populated with primitive (single-variable) reference models for all six abstractions included in the artefact scope. A manual literature analysis technique was employed to extract generalised, primitive (single-variable) inputs for the reference models from the composite (multi-variable) knowledge base on IKNs. These reference models feature varying levels of detail, with appropriate representation formats or meta-models for each abstraction.

In employing the artefact development roadmap, architecture framework selection and artefact behaviour specification, the construction of the reference architecture for IKNs represented the simultaneous investigation of the *relationships between the views, meta-relationships and IKN life cycle axes of the modelling framework*. This is illustrated with the blue planes of the modelling framework to the right of Figure 11.6.

The cycle showed how an artefact that functions as a reference architecture in the engineering of IKNs could be developed through the implementation of a suitable architecture framework. The construction of the reference architecture for IKNs therefore answers SRQ4 and achieves SO4, and concluded the activities of the fourth design cycle.

The reference architecture for IKNs was designed and developed by achieving and answering, respectively, SO1 and SRQ1 (refer to chapter 5), SO2 and SRQ2 (refer to chapter 6), SO3 and SRQ3 (refer to chapter 7), as well as SO4 and SRQ4 (refer to chapter 8). The reference architecture for IKNs was therefore seen as a candidate answer to the primary research question (PRQ), potentially achieving the research objective (RO). The design and development activity of the research design was therefore concluded, leading to the demonstration and evaluation of the reference architecture in the next activity.

11.2.4 Demonstration and evaluation

This section provides a summary of the demonstration and evaluation of the reference architecture for IKNs, as documented in Part 4 of the document.

The fourth activity in the research design employed in this study was presented in chapter 9. The activity demonstrated and evaluated the artefact that was designed and developed in the third activity (refer to section 11.2.3). The demonstration phase of this activity indicated how the artefact achieves the solution objectives identified in Part 2 of the document, through the execution of a number of illustrative scenarios. The evaluation phase then reflected on the performance of the artefact and the extent to which it did in fact achieve these solution objectives. This combination of the demonstration and evaluation phases tested the artefact's ability to function, i.e. to act as a solution that achieves its specified objectives.

In order to demonstrate the artefact, the cooperation of a number of operational IKNs was enlisted to serve as illustrative scenarios. For each of these scenario networks, the reference architecture for IKNs was implemented in as close an approximation of its expected working environment as was possible. The success of the artefact in achieving its various solution objectives was then evaluated in a structured way.

The demonstration and evaluation of the reference architecture for IKNs through multiple illustrative scenarios showed that the artefact achieves all four its solution objectives, as identified in Part 2 of the document. This version of the artefact is therefore seen as a candidate solution to the research problem, and no further iteration of the third activity in the research design for the study (design and development) was required.

11.2.5 Contributions

This section provides a summary of the contributions made by the study, as communicated in Part 5 of the document.

Chapter 10 presented an overview of newly generated knowledge contained in the research contributions made through the course of the study (refer to Figure 11.7). The case was made for the usefulness of this newly generated knowledge to solve the research problem and various research questions. These contributions extend the body of knowledge in the domains of IKNs and enterprise engineering as described in chapter 2.

The contributions made by the study may be summarised as follows:

1. The *primary contribution* of the study is a *reference architecture for IKNs* that solves the research problem that initiated the project (refer to Part 1) by enhancing the state of IKN engineering and development.

2. Various *secondary contributions* also resulted from the study:
    - 2.1. *Artefact behaviour specification*, indicating the desired behaviour of reference architectures in the engineering of IKNs.
    - 2.2. *Architecture framework selection*, indicating the use of the Zachman framework to describe non-hierarchical enterprises.
    - 2.3. *Artefact development roadmap*, indicating the use of the Zachman framework in the development of reference architectures.
    - 2.4. *Visual representation format* for the Zachman framework.
    - 2.5. *Research design* for design science research.

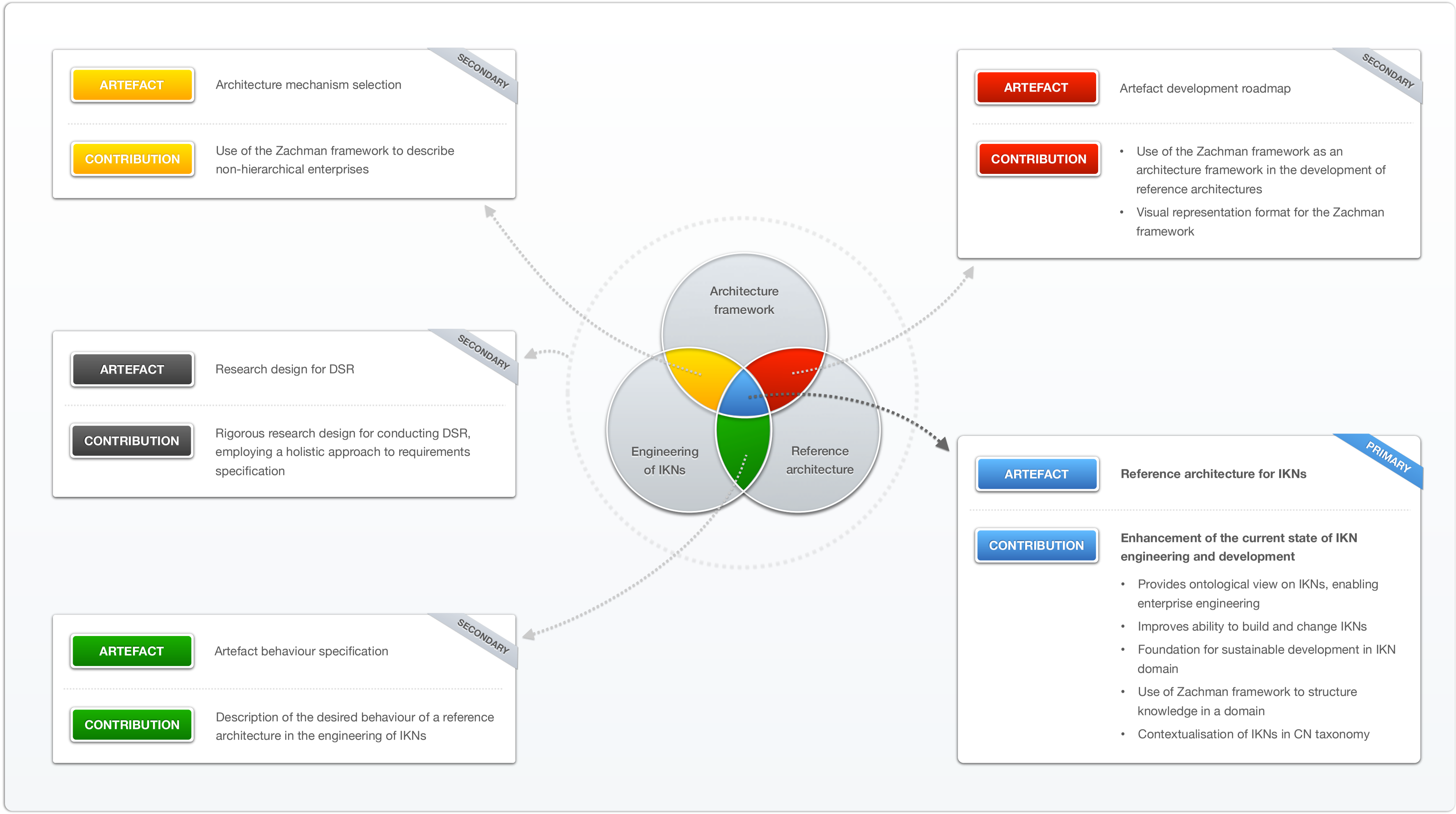


Figure 11.7: Summary of contributions

11.3 Further research

This section suggests opportunities further research, based on the results obtained in the study.

11.3.1 Reference architecture for IKNs

Opportunities exist for extending the reference architecture for IKNs in terms of three facets: field testing, component improvement and extension, and scope extension.

The execution of the entire Relevance cycle of the DSR method falls outside the scope of this study due to the timeframes afforded. The evaluation of the reference architecture for IKNs in this study is therefore limited to illustrative scenarios that illustrate the utility of the artefact in solving the research problem. The effectiveness of the reference architecture for IKNs, i.e. its effect on organisations in which it is implemented, however, remains to be evaluated. The opportunity exists to conduct multiple *case studies* in which the artefact is employed as a reference architecture in the design, refinement and phase-out of IKNs, and for the effectiveness of the artefact to be observed in these environments.

The modular nature of the Zachman framework enables the *extension and improvement of components* of the reference architecture for IKNs without invalidating the entire artefact. It would therefore be possible to for instance, extend the inventory set definition model for *knowledge* (refer to Figure 8.9) to include a concept such as implicit knowledge (Newman & Conrad 2000). In similar vein, it would be possible to devise and include an improved process flow definition model for *innovation* (refer to Figure 8.17) as the understanding of the collaborative innovation process continues to deepen.

Regarding *scope extension*, reference was made in section 7.6.5 to the fact that a case could be made for the feasibility of reference models for the third row of the reference architecture for IKNs. Such reference models would describe patterns that are commonly found in the system logic of IKNs. The decision to exclude the development of these models from the study, while solely focusing on the business information in the top two rows, was made with the study's research scope in mind. Such third-row reference models would directly address members of the secondary audience of the artefact and would therefore be valuable to improve the alignment of business and technology in IKNs at a system level.

The creation of third-row reference models from common patterns in the system logic of IKNs, however, is a non-trivial exercise. The perspectives (rows) of the Zachman framework each requires a different way of thinking about the enterprise. A system-level approach to the body of knowledge on IKNs would be required to generate reference models for the third row of the reference architecture. This is different, and yet complementary, to the strategic and conceptual approach employed in chapter 8 to generate reference models for the top two rows of the reference architecture for IKNs. The generation of these third-row reference models would however be facilitated by the existence of reference models for the top two rows.

11.3.2 Framework for IKNs

In order to engineer IKNs as systems, both functional and constructional principles are required (refer to Figure 1.2). Through the study of Schutte (2010), a methodology that provides *functional* principles for IKNs was developed, while this study contributed a reference architecture for IKNs that focuses on *constructional* principles. The opportunity therefore exists to combine these two artefacts into a *framework for IKNs*. Such a framework for IKNs will provide holistic support for the engineering of IKNs. The framework could possibly be packaged in a web-based platform that allows network stakeholders to gain remote access to particular architecture descriptions of their IKNs, as well as the reference models and methodological content.

11.3.3 Reference architectures for collaborative networks

An artefact behaviour specification, the selection of the Zachman framework, and an artefact development roadmap were used as building blocks in order to generate the reference architecture for IKNs in this study. Since the scenarios that were included in the requirements frame of reference in the study specifically refer to the engineering *of IKNs*, the artefact behaviour specification and selection of the Zachman framework are currently focussed on this class of collaborative networks.

By extending the concept of EE to IKNs, this study has however shown that an engineering approach may be applied to organisational forms that include ad-hoc collaborative elements (refer to Figure 1.1). The opportunity therefore exists to investigate the feasibility of a *reference architecture for collaborative networks*. In such an investigation it would be required to alter the abovementioned scenario description to the “engineering of collaborative networks”. This would result in an artefact behaviour specification and architecture framework selection to be used as building blocks for the development of a reference architecture for the generalised class of collaborative networks. From the experience of this study it could however be reasonably expected that the Zachman framework would still be an appropriate architecture framework selection. If the Zachman framework was indeed deemed suitable for application to collaborative networks in general, the artefact development roadmap developed in this study would remain intact.

It would be expected that reference models in such a reference architecture would have less of a focus on the innovation process than is the case in the reference architecture for IKNs.

11.4 Reflections

11.4.1 Methodological reflection

This section reflects on the methodology that was adopted to achieve the research objective. Certain generic elements of the methodological approach discussed in this section are seen as a secondary contribution of the study, and are discussed in greater detail in 10.3.5.

The study was conducted within a *pragmatic* paradigm, and was therefore initiated by an observed problem in the state of engineering of IKNs. Pragmatism aims at contributing constructive knowledge that is useful for action and change in order to solve research problems (Goldkuhl 2012). Research objectives were therefore the primary driving forces in the study, even though they were always associated with research questions to guide the process of inquiry. Unlike a positivistic paradigm where a single stable reality is constructed through experimentation and hypothesis testing, pragmatism links meaning and value to practical consequences. This means that various possible solutions to a research problem may exist, and that if an artefact has the desired practical consequences, it is deemed to be scientifically sound. Therefore, while the reference architecture for IKNs was demonstrated and evaluated to be a satisfactory solution to the research problem (refer to Part 4), alternate solutions to the research problem could also be conceived.

Within the pragmatic paradigm, a *qualitative* approach primarily concerned with interpretation was employed in the study. An interpretive understanding assumes that meaning is context-specific and constructive, and there is therefore no single “correct” meaning. Thus, there is a possibility that two different researchers may apply sound, but similar research methods, yet arrive at different answers or solutions. Qualitative research requires a different concept of reliability than quantitative research. In making qualitative research reliable, Steinke (2004) suggests a systematic and transparent research process, which includes motivations for every conclusion and every step in the research process.

The study furthermore adopted the *design science research* method that aims to solve observed problems and make scientific contributions by designing artefacts. Since design has an inherent creative element and employs abductive reasoning, a natural variation may be expected in solutions produced through DSR for the same research problem. The methodological reflection conducted in the following sections is therefore especially pertinent in a study that is rooted in pragmatism and qualitative research, and which employs DSR. The above discussion furthermore served to highlight why a large part of this document is dedicated to the systematic and transparent presentation of the arguments that form the basis of the reference architecture for IKNs.

The following sections will examine the research design employed in the study, which was based on the DSRM presented by Peffers et al. (2008), and reflect on the appropriateness of steps taken in the study. Possible alternate steps that could conceivably have been taken are also identified.

11.4.1.1 Problem identification and motivation

The problem that was identified and motivated in this activity reads as follows (refer to Part 1):

> **No reference architecture exists for use in the engineering of IKNs.**

The need to engineer IKNs and the lack of an artefact that provides constructional principles to the design of these networks as systems were motivated from literature (refer to chapter 2). However, it could be argued that the term “reference architecture” need not have been embedded in the problem description. Reference architectures have various connotations stemming from their history of use within information systems and hierarchical enterprises, and may not necessarily be associated with collaborative networks. It would therefore also have been possible to describe the problem as:

> *The engineering of IKNs requires an artefact that provides constructional principles.*

From this alternate problem statement, the exact nature of the artefact would only be determined during the design and development activity (refer to section 11.4.1.3). However, by embedding both the terms “architecture” and “engineering” in the problem statement identification and motivation in Part 1, the study gains focus by acknowledging the connection between the EA and EE disciplines.

11.4.1.2 Definition of solution objectives

In the second activity (refer to Part 2) solution objectives for the design artefact were inferred rationally from the problem statement by employing a qualitative approach to requirements specification. This approach considered technical and functional requirements, as well as scenarios in which the designed artefact was expected to function. The definition of solution objectives were based on the sequential consideration of intersections between these various types of requirements. Achieving each of these solution objectives contributed new knowledge on the design of the artefact, leading to a holistic design. This generation of constructive knowledge is in keeping with the DSR mantra of “learning through building” (Kuechler & Vaishnavi 2011). The combination of requirements specification and DSR, including the alignment of design cycles with solution objectives, is highlighted as one of the secondary contributions of the study (refer to section 10.3.5).

A possible variation on the solution objectives that were defined in chapter 4 would occur if the alternate problem statement presented in the previous section was indeed followed. In this case, the functional requirement for the artefact would not have been to “function as a reference architecture”, but perhaps simply to “provide constructional principles”. The technical requirement of the artefact would then be to employ some “constructional mechanism”. This could have resulted in the following set of solution objectives being defined as follows:

1. *To provide constructional principles to the engineering of IKNs.*
2. *To employ a constructional mechanism that is suitable to the engineering of IKNs.*
3. *To implement the selected constructional mechanism in a way that enables the artefact to provide constructional principles.*
4. *To provide constructional principles to the engineering of IKNs through the implementation of a suitable constructional mechanism.*

11.4.1.3 Design and development

Part 3 documented the design and development activity of the reference architecture for IKNs through four iterative design cycles. These design cycles were aligned with, and driven by, the solution objectives and associated secondary research questions defined in Part 2. This imparted the systematic and transparent structure to the design and development of the reference architecture for IKNs that is required when documenting a study that adopts qualitative research and DSR.

It should be noted that it would be possible to swap the order of the cycles in which the second and third solution objectives from the second activity are pursued. The order in which it was performed in Part 3 resulted in an architecture framework that is suitable to the engineering of IKNs being selected before it was determined how this architecture framework could bring about the required functionality. This stresses the importance of aligning functional and technical requirements when constituting a requirements frame of reference. An inverted cycle sequence would imply that it is first determined which architecture framework(s) can bring about the required functionality, before determining one which is suitable to the engineering of IKNs is selected. In practice, both argument sequences were employed in the study, although the former was selected as the most beneficial way to present the research in this document. This decision was made in order to discuss the motivations for selecting the Zachman framework to describe collaborative networks before the implementation of the framework was discussed in greater detail.

In the case where the alternate problem statement and solution objectives from the previous sections were employed, the design of the artefact would have included determining an appropriate artefact type. The design and development process, however, benefits from the clarity provided by the problem statement and solution objectives in Part 1 and Part 2 of the study.

The qualitative systematic review that was conducted as part of the fourth design cycle allowed the flexibility of data collection that was required in order to illicit primitive (single-variable) data from composite (multi-variable) descriptions of IKNs. Alternate approaches, for example topic modelling, may have resulted in shorter data collection time, but would have lacked the required flexibility to collect and interpret primitive data.

11.4.1.4 Demonstration and evaluation

The demonstration and evaluation activity of the research design employed in the study was documented in Part 4. The activity demonstrated how the reference architecture for IKNs attempted to achieve each of its solution objectives, and evaluated its success in doing so in order to investigate the utility of the artefact. The structure provided by the requirements frame of reference adopted in Part 2 allowed for a structured and transparent approach to both these actions. The demonstration and evaluation of the fourth solution objective verified the utility of the artefact from a holistic point of view. By demonstrating and evaluating the first three solution objectives, the internal design logic of the reference architecture for IKNs was also verified.

Illustrative scenarios may be motivated as an appropriate method to demonstrate and evaluate the performance of socio-technical artefacts (Venable et al. 2012; Peffers et al. 2012), including the reference architecture for IKNs. The primary motivation for this perception is that the method exposes the artefact to an environment that closely resembles the scenarios in which it is required to operate, while being able to be completed in acceptable timeframes. Larger-scale case study implementations of the artefact in the field could potentially take several years to complete, and would have exceeded the timeframe afforded to this particular doctoral study. The involvement of the researcher as a participant observer in the illustrative scenarios does allow for the possible biasing of demonstration and evaluation data. An attempt to mediate this position was made by defining a range of predefined questions that were considered during the architecture workshops. Given the complex nature of the reference architecture for IKNs, however, an improved evaluation design that met the practical constraints of the study could not be envisaged.

Various operational IKNs from within the Western Cape region of South Africa could be identified and enlisted to serve as illustrative scenarios in the demonstration of the reference architecture. This serves as evidence for the pervasiveness of collaborative networks, also in developing countries. During the workshops that were conducted with representatives from these scenario networks, it was clear that the identification of target audiences for the reference architecture for IKNs was accurate. Individuals with a strategic and/or conceptual perspective on their network had very little difficulty in relating to the information contained in the reference architecture, as well as the value and utility of the artefact. Those individuals with a more technical background, however, had some difficulty in relating to the reference models, although they could comprehend the functionality of the artefact.

It was further noted that, while the abstractions (columns) of the Zachman framework have no prescribed order, it was beneficial to follow a certain sequence in discussing the network architectures with workshop participants. By beginning with the motivation abstraction (“Why?”), the boundaries of the network could be identified by virtue of the purpose of the network to the organisations that the participants represented. From there it was possible to describe the responsibility (“Who?”) and the process (“How?”) abstractions.

The inventory (“What?”), timing (“When?”) and distributions (“Where?”) abstractions could then be completed with relative ease.

11.4.1.5 Communication

The contributions made by this study are communicated in Part 5. This discussion highlighted the novelty and utility of the reference architecture for IKNs, as well as the rigour of its design and its effectiveness. Various secondary research contributions in the form of artefacts and a methodological approach were also identified and discussed. This activity however also relates to the communication of the entire study through the compilation of this document.

In the entire document, including Part 5, the methodology adopted for the study was used to structure the communication of the research. This approach imparted the systematic and transparent structure which is required when documenting and presenting qualitative research that adopts DSR, and ensures the reliability of the study (Steinke 2004).

The approach to the design of EE artefacts through a combination of DSR and requirements specification which was discussed in this section is not specific to this study. It is therefore seen as a generic contribution to the use of DSR in EE (refer to section 10.3.5).

11.4.2 Scientific reflection

The purpose of this section is to reflect on the various scientific contributions made by the study within a broader context, i.e. relating the contributions to the body of scientific knowledge.

As summarised in section 11.2.5, the contributions made by the study are as follows:

1. The *primary contribution* of the study is devising a *reference architecture for IKNs* that solves the research problem that initiated the project (refer to Part 1) by enhancing the state of IKN engineering and development.

2. Various *secondary contributions* also result from the study:
    - 2.1. *Artefact behaviour specification*, indicating the desired behaviour of reference architectures in the engineering of IKNs.
    - 2.2. *Architecture framework selection*, indicating the use of the Zachman framework to describe non-hierarchical enterprises.
    - 2.3. *Artefact development roadmap*, indicating the use of the Zachman framework in the development of reference architectures.
    - 2.4. *Visual representation format* for the Zachman framework.
    - 2.5. *Research design* for design science research.

The following sections reflect on the impact of each of these contributions on the scientific body of knowledge. Note that the methodological contribution in 2.5 above, however, is reflected upon in detail in section 11.4.1.

11.4.2.1 Reference architecture for IKNs

This primary contribution of the study is discussed in greater detail in section 10.2.

In Part 1 of the study the lack of a reference architecture to be used in the engineering of IKNs was identified and motivated as being the research problem that this study would pursue (refer to section 1.3). The reference architecture for IKNs which was designed and developed in the study solves this research problem and contributes to the body of scientific knowledge in various ways.

From a practical point of view, the reference architecture for IKNs is a valuable resource to the EE toolset of practitioners in the domain. By providing inputs for describing the construction of networks independent of their implementation, the reference architecture for IKNs facilitates the ontological view on these networks that had been lacking previously. This ontological view corresponds to a “white-box” model of IKNs, which improves the ability of practitioners to build and change networks (Dietz 2006). In keeping with the pragmatic paradigm of the study, the knowledge contributed through the development of the reference architecture for IKNs may therefore be used for action and change. These “white-box” models of IKNs that are now possible complement the “black-box” models of IKNs that are currently facilitated by existing resources like the methodology for IKNs (Schutte 2010) and which focus on the operation and control of networks. IKNs may now be comprehensively designed, operated, and, indeed, studied as systems. This is the first step toward extending the concept of EE to IKNs, and collaborative networks in general.

The reference architecture for IKNs provides the foundation for more sustainable development in the IKN domain, since it completes a comprehensive systems view of this class of networks. It contributes an ontological view on IKNs which is independent of implementation, and since the behaviour of a system is brought about through its construction, it allows for the explanation and modelling of various complex behavioural traits of these networks. One example of this is that the management of IKNs to evolve from an explorative to an exploitative mode through the innovation life cycle (Chesbrough & Prencipe 2008) may now be described in constructional terms.

A further scientific implication of the contribution of the reference architecture for IKNs is that it is an example of the use of the Zachman framework to structure the body of knowledge in a certain domain. With its comprehensive abstractions and perspectives, and no bias toward any particular subject, the framework is a useful taxonomy within which to classify existing knowledge in a domain. This classification illuminates areas where gaps exist in the understanding of a domain or where the knowledge that does exist is not yet mature. Relationships between topics in a domain are also highlighted by the Zachman framework classification scheme, which helps to identify areas that warrant further investigation.

A final scientific implication of the contribution of the reference architecture for IKNs is an improved contextualisation of IKNs against the larger taxonomy of CNs. Simultaneous developments in the CN domain during the time when this study was conducted allowed for a much better understanding of the wider landscape of networks (Camarinha-Matos & Afsarmanesh 2012). This has enabled the classification of IKNs in the taxonomy of CNs as a hybrid network form that includes elements from both CNOs and ad-hoc collaborations. As such, the reference architecture for IKNs utilised specialised elements from the ARCON reference model for CNOs (Camarinha-Matos & Afsarmanesh 2008a) as inputs for the reference models it contains.

11.4.2.2 Artefact behaviour specification

This secondary contribution of the study is discussed in greater detail in section 10.3.1.

The first secondary research question (SRQ1) highlighted a gap in the current understanding of IKNs by asking what the desired behaviour of a reference architecture in the engineering of IKNs is. The artefact behaviour specification developed in chapter 5 addresses this knowledge gap by describing the role of an artefact that conveys constructional principles in the life cycle of IKNs. The artefact behaviour specification therefore functionally describes the landscape of IKN engineering. Since it does not include any specification of a preferred technical approach, the artefact behaviour specification is generic and may be exploited by artefacts other than the reference architecture for IKNs.

11.4.2.3 Architecture framework selection

This secondary contribution of the study is discussed in greater detail in section 10.3.2.

The second secondary research question (SRQ2) indicated that a gap exists in the current understanding of IKNs by asking whether an architecture framework which is suitable in the engineering of IKNs is available. The investigation conducted in chapter 6 yielded the Zachman framework as a viable option, and therefore addresses this particular knowledge gap.

The Zachman framework is traditionally associated with hierarchical enterprises and a perception exists that it is not equipped to describe modern organisational forms such as collaborative networks. As part of the investigation into the requirements for its suitability in the engineering of IKNs, a rigorous set of criteria was developed. Contrary to what may intuitively be expected by practitioners in the network architecture domain, the Zachman framework satisfied all these criteria.

11.4.2.4 Artefact development roadmap

This secondary contribution of the study is discussed in greater detail in section 10.3.3.

The third secondary research question (SRQ3) highlighted a gap in the current understanding of reference architectures by asking how the Zachman framework can be implemented in a way that enables the artefact which is produced to function as a reference architecture. The Zachman framework is generally implemented in order to structure particular architecture descriptions rather than reference architectures, and a novel implementation of the framework therefore had to be developed.

An artefact development roadmap was compiled in order to address this knowledge gap. The roadmap contributes a generalised approach to implementing the Zachman framework in order to create reference architectures. The roadmap specifies values for various architecture framework dimensions, as well as the interpretation of these values in the context of the Zachman framework.

11.4.2.5 Visual representation format

This secondary contribution of the study is discussed in greater detail in section 10.3.4.

The seventh and eighth steps of the artefact development roadmap discussed in the previous section imply the use of a semi-formal visual representation format for the Zachman framework which accommodates various levels of detail. Apart from specifying the principles of primitive modelling and supplying basic meta-models for each abstraction, the Zachman framework, however, is agnostic of any specific visual representation format.

A common approach is therefore to attempt a strict application of the rules of primitive modelling while employing existing notations such as UML (Fatolahi & Shams 2006; Gerber et al. 2013). The success of this approach, however, is greatly dependent on the rigour with which the practitioner that develops the models adheres to the principles of primitive modelling, since these notations are in principle intended for composite (multi-variable) modelling. Since the study did not intend to inherit the uncertainty of such an approach, a novel visual representation format that adheres to the meta-models of the Zachman framework, while accommodating various levels of detail, was therefore developed.

The contribution of this visual representation format complements the contribution of the Zachman framework to structure the knowledge in a domain (refer to section 11.4.2.1). The combination of the taxonomy scheme provided by the Zachman framework, along with the discussed visual representation format, allows for the structured visual representation of knowledge in a domain.

11.5 Conclusion

This study has developed a reference architecture for IKNs by performing design science research within a pragmatic paradigm. The reference architecture provides constructional principles in the engineering of IKNs, thus enabling the design, operation and research of this class of collaborative networks. The study takes a first step toward extending the concept of EE to IKNs, and collaborative networks in general. This enables the greater adaptability of these networks to the dynamic environment of the Information Age.

References

Abernathy, W.J. & Utterback, J.M., 1978. Patterns of industrial innovation. *Journal Title: Technology review. Ariel*, 80(7), pp.41–47.

Acha, V. & Cusmano, L., 2005. Governance and co-ordination of distributed innovation processes: patterns of R&D co-operation in the upstream petroleum industry. *Economics of Innovation and New Technology*, 14(1-2), pp.1–21.

Adamides, E.D. & Karacapilidis, N., 2006. Information technology support for the knowledge and social processes of innovation management. *Technovation*, 26(1), pp.50–59.

Adebesin, F., Kotzé, P. & Gelderblom, H., 2011. Design research as a framework to evaluate the usability and accessibility of the Digital Doorway. In Proceedings of Design, Development and Research. Cape Town.

Alexander, C., 1964. *Notes on the Synthesis of Form*, Harvard University Press.

Anklam, P., 2007. *Net Work*, Burlington, MA, USA: Butterworth–Heinemann.

Asheim, B.T., Ebersberger, B. & Herstad, S.J., 2011. MNCs between the local and the global: knowledge bases, proximity and distributed knowledge networks. In DRUID Summer Conference. London, pp. 1–30.

Álvares-Ribeiro, N., Martins, Â. & Pinto Ferreira, J., 2004. Introducing Time Horizons to Enterprise Networking Architecture. *Virtual Enterprises and Collaborative Networks*, pp.63–70.

Back, A., Krogh, von, G. & Seufert, A., 2005. *Putting Knowledge Networks Into Action*, Springer.

Badaracco, J., 1991. *The Knowledge Link*, Harvard Business Press.

Badii, A. & Sharif, A., 2003. Information management and knowledge integration for enterprise innovation. *Logistics Information Management*, 16(2), pp.145–155.

Baker, K.A., 2002. Innovation. In *Management Benchmark Study*. The Air University.

Barger, R.N., 2001. Philosophical Belief Systems. *www3.nd.edu*. Available at: http://www3.nd.edu/~rbarger/philblfs-with-epis.html [Accessed September 13, 2013].

Barthélemy, J., 2003. The seven deadly sins of outsourcing. *Academy of Management Executive*, 17(2), pp.87–98.

Baskerville, R.L. & Wood-Harper, A.T., 1996. A critical perspective on action research as a method for information systems research. *Journal of Information Technology*, 11(3), pp.235–246.

Becker, M., Rullani, F. & Zirpoli, F., 2009. Coordinating distributed innovation processes: The case of the automotive and open source software industries. In DRUID Summer Conference. Copenhagen, pp. 1–26.

Bentley, L.D., Whitten, J.L. & Randolph, G., 2007. *Systems analysis and design for the global enterprise*, Irwin/McGraw-Hill.

Berasategi, L., Arana, J. & Castellano, E., 2011. A comprehensive framework for collaborative networked innovation. *Production Planning & Control*, 22(5-6), pp.581–593.

Bernard, S.A., 2012. *An Introduction to Enterprise Architecture* Third, AuthorHouse.

Bernus, P. & Nemes, L., 1996. A framework to define a generic enterprise reference architecture and methodology. *Computer Integrated Manufacturing Systems*, 9(3), pp.179–191.

Bernus, P., Nemes, L. & Williams, T.J., 1996. *Architectures for Enterprise Integration*, London: Chapman & Hall.

Bertalanffy, von, L., 1969. *General System Theory: Foundations, Development, Applications*, New York: George

Braziller.

Bevan, J., 1987. What is Co-Makership. *International Journal of Quality & Reliability Management*, 4(3), pp.47–56.

Bichler, M., 2014. Reflections on the State of Design Science Research. *Business & Information Systems Engineering*, 6(2), pp.71–72.

Biolchini, J., Mian, P.G. & Natali, A., 2005. *Systematic review in software engineering*, Rio de Janeiro: Systems Engineering and Computer Science Department, COPPE / UFRJ.

Bogers, M. & West, J., 2012. Managing Distributed Innovation: Strategic Utilization of Open and User Innovation. *Creativity and Innovation Management*, 21(1), pp.61–75.

Booth, A., Papaioannou, D. & Sutton, A., 2012. *Systematic Approaches to a Successful Literature Review*, SAGE.

Bornemann, M. et al., 2003. *An Illustrated Guide to Knowledge Management*, Wissensmanagement Forum.

Brennan, A. & Dooley, L., 2005. Networked creativity: a structured management framework for stimulating innovation. *Technovation*, 25(12), pp.1388–1399.

Brinton, L.J., 2000. *The Structure of Modern English: A Linguistic Introduction*, John Benjamins B.V.

Buchanan, M., 2003. *Nexus: Small Worlds and the Groundbreaking Theory of Networks*, W. W. Norton & Company.

Bullinger, H.J., Auernhammer, K. & Gomeringer, A., 2004. Managing innovation networks in the knowledge-driven economy. *International Journal of Production Research*, 42(17), pp.3337–3353.

Bunge, M., 1979. *Treatise on Basic Philosophy: Volume 4*, Springer.

Burgelman, R.A., Christensen, C.M. & Wheelwright, S.C., 2009. *Strategic Management of Technology and Innovation*, McGraw-Hill.

Camarinha-Matos, L.M., 2009. Collaborative networked organizations: Status and trends in manufacturing. *Annual Reviews in Control*, 33(2), pp.199–208.

Camarinha-Matos, L.M. & Afsarmanesh, H., 2007. A comprehensive modeling framework for collaborative networked organizations. *Journal of Intelligent Manufacturing*, 18(5), pp.529–542.

Camarinha-Matos, L.M. & Afsarmanesh, H., 2011. Behavioral aspects in collaborative enterprise networks. In IEEE 9th International Conference on Industrial Informatics. Lisbon, Portugal: IEEE, pp. 12–19.

Camarinha-Matos, L.M. & Afsarmanesh, H., 2005. Collaborative networks: a new scientific discipline. *Journal of Intelligent Manufacturing*, 16(4-5), pp.439–452.

Camarinha-Matos, L.M. & Afsarmanesh, H., 2008a. *Collaborative Networks: Reference Modeling*, Springer.

Camarinha-Matos, L.M. & Afsarmanesh, H., 2006. Collaborative networks. In PROLAMAT. Shanghai, China: Springer, pp. 26–40.

Camarinha-Matos, L.M. & Afsarmanesh, H., 2008b. Concept of Collaboration. In IGI Global, pp. 311–315.

Camarinha-Matos, L.M. & Afsarmanesh, H., 2008c. On reference models for collaborative networked organizations. *International Journal of Production Research*, 46(9), pp.2453–2469.

Camarinha-Matos, L.M. & Afsarmanesh, H., 2012. Taxonomy of collaborative networks forms. *Future Internet Enterprise Systems (FInES), Task Force on Collaborative Networks*.

Camarinha-Matos, L.M. et al., 2009. Collaborative networked organizations – Concepts and practice in manufacturing enterprises. *COMPUTERS & INDUSTRIAL ENGINEERING*, pp.1–15.

Camarinha-Matos, L.M., Afsarmanesh, H. & Koelmel, B., 2011. Collaborative Networks in Support of Service-Enhanced Products. In PRO-VE'11, FIP AICT Series 362/2011, Springer. Sao Paulo, Brazil: Springer, pp. 95–104.

Carayannis, E.G. & Campbell, D.F.J., 2006. *Knowledge Creation, Diffusion, and Use in Innovation Networks and Knowledge Clusters*, Greenwood Publishing Group.

Carneiro, A., 2000. How does knowledge management influence innovation and competitiveness? *Journal of Knowledge Management*, 4(2), pp.87–98.

Castells, M., 2010. *The Rise of the Network Society* 2nd ed., John Wiley & Sons.

Cavusgil, S.T., Calantone, R.J. & Zhao, Y., 2003. Tacit knowledge transfer and firm innovation capability. *Journal of Business & Industrial Marketing*, 18(1), pp.6–21.

Chesbrough, H., 2003. *Open innovation: The new imperative for creating and profiting from technology*, Harvard Business Press.

Chesbrough, H. & Prencipe, A., 2008. Networks of innovation and modularity: a dynamic perspective. *International Journal of Technology Management*, 42(4), p.414.

Chesbrough, H., Vanhaverbeke, W. & West, J., 2006a. Open innovation: a new paradigm for understanding industrial innovation. In *Open Innovation: Researching a New Paradigm*. Oxford University Press, pp. 1–12.

Chesbrough, H., Vanhaverbeke, W. & West, J., 2006b. *Open Innovation: Researching a New Paradigm*, Oxford University Press.

Choudhary, A.K. et al., 2013. Knowledge management and supporting tools for collaborative networks. *International Journal of Production Research*, 51(7), pp.1953–1957.

Clark, K.B. & Fujimoto, T., 1991. *Product development performance: Strategy, organization, and management in the world auto industry*, Boston: Harvard Business School Press.

Cloutier, R. et al., 2010. The Concept of Reference Architectures. *Systems Engineering*, 13(1), pp.14–27.

Contractor, F.J. & Lorange, P., 2002. *Cooperative strategies and alliances*, Elsevier Science Boston, MA.

Coombs, R., Harvey, M. & Tether, B., 2001. Analysing distributed innovation processes: a CRIC position paper.

Cooper, J.R., 1998. A multidimensional approach to the adoption of innovation. *Management Decision*, 36(8), pp.493–502.

Cowan, R. & Jonard, N., 2009. Knowledge portfolios and the organization of innovation networks. *Academy of management review*, 34(2), pp.320–342.

Cowan, R., Jonard, N. & Zimmermann, J.-B., 2007. Bilateral Collaboration and the Emergence of Innovation Networks. *Management science*, 53(7), pp.1051–1067.

Cox, H. & Mowatt, S., 2004. Consumer-driven innovation networks and e-business management systems. *Qualitative Market Research: An International Journal*, 7(1), pp.9–19.

Cruse, D.A., 1986. *Lexical Semantics*, Cambridge University Press.

Daft, R.L., 2012. *Organization Theory and Design* 11 ed., Cengage Learning.

Damschroder, L.J. et al., 2009. Fostering implementation of health services research findings into practice: a consolidated framework for advancing implementation science. *Implement Science*, 4(50).

De Villiers, M.R., 2005. Three approaches as pillars for interpretive information systems research: development research, action research and grounded theory. In SAICSIT. White River.

De Vries, M., 2012. *A Process Reuse Identification Framework Using an Alignment Model*. University of Pretoria.

DeBoever, L.R., Paras, G.S. & Westbrock, T., 2010. A pragmatic approach to a highly effective enterprise architecture program. In L. Kappelman, ed. *The SIM guide to enterprise architecture*. Boca Raton: CRC Press, pp. 156–161.

Dekkers, R., 2010. A Co-evolutionary Perspective on Distributed Manufacturing. In *Distributed Manufacturing*. London: Springer London, pp. 29–50.

Denise, L., 1999. Collaboration vs. C-Three (Cooperation, Coordination, and Communication). *Innovating*, 7(3).

Department of Industrial Engineering, Stellenbosch University, 2013. Enterprise Engineering E. E. Group, ed., pp.1–220.

Desouza, K.C. et al., 2009. Crafting organizational innovation processes. *Innovation: Management, policy & practice*, 11(1), pp.6–33.

Dhanaraj, C. & Parkhe, A., 2006. Orchestrating innovation networks. *Academy of management review*, 31(3), pp.659–669.

Dietz, J.L. & Hoogervorst, J.A., 2008. Enterprise ontology in enterprise engineering. In ACM symposium on Applied computing. Fortaleza, Brazil: ACM, pp. 572–579.

Dietz, J. et al., 2013. The discipline of enterprise engineering. *International Journal of Organisational Design and Engineering*, 3(1), p.86-114.

Dietz, J.L. & Hoogervorst, J.A., 2012. The Principles of Enterprise Engineering. *Advances in Enterprise Engineering VI*, 110, pp.15–30.

Dietz, J.L.G., 2006. *Enterprise Ontology*, Springer.

Dilk, C. et al., 2008. State and development of innovation networks: Evidence from the European vehicle sector. *Management Decision*, 46(5), pp.691–701.

Docherty, M., 2006. Primer on open innovation: Principles and practice. *PDMA Visions Magazine*, 30(2), pp.13–17.

Dooley, L. & O'Sullivan, D., 2007. Managing within distributed innovation networks. *International Journal of Innovation Management*, 11(03), pp.397–416.

Dooley, L. & O'Sullivan, D., 2000. Systems innovation manager. *Production Planning & Control*, 11(4), pp.369–379.

Douthwaite, B., 2006. Enabling innovation: Technology-and system-level approaches that capitalize on complexity. *innovations*, 1(4), pp.93–110.

Drucker, P., 2007. *Innovation and Entrepreneurship*, Butterworth–Heinemann.

Du Preez, N.D. et al., 2006. *A Roadmapping and Conceptual Framework Based Approach For Efficient Knowledge and Innovation Management*.

Du Preez, N.D. & Louw, L., 2008. A framework for managing the innovation process. In PICMET. Cape Town: IEEE, pp. 546–558.

Du Preez, N.D. & Louw, L., 2007. Managing the Knowledge Supply Chain to Support Innovation. *COMA'07*.

Du Preez, N.D., Louw, L. & Lutters, E., 2008a. A Knowledge Network Approach Supporting the Value Chain. In *Methods and Tools for Effective Knowledge Life-Cycle-Management*. Berlin, Heidelberg: Springer, pp. 159–168.

Du Preez, N.D., Louw, L., Essmann, H., et al., 2008b. Reference architectures as knowledge management tools guiding and supporting enterprise engineering. In A. Bernard & S. Tichkiewitz, eds. *Methods and Tools for Effective Knowledge Life-Cycle-Management*. Berlin, Heidelberg: Springer, pp. 391–414.

Eisenhardt, K.M. & Martin, J.A., 2000. Dynamic capabilities: what are they? *Strategic management journal*, 21(10-11), pp.1105–1121.

Erickson, C.L. & Jacoby, S.M., 2003. The Effect of Employer Networks on Workplace Innovation and Training. *Industrial and Labour Relations Review*, 56(2), pp.203–223.

Ermilova, E. & Afsarmanesh, H., 2007. Modeling and management of profiles and competencies in VBEs. *Journal of Intelligent Manufacturing*, 18(5), pp.561–586.

Eschenbächer, J., Seifert, M. & Thoben, K.-D., 2011. Improving distributed innovation processes in virtual organisations through the evaluation of collaboration intensities. *Production Planning & Control*, 22(5-6), pp.473–487.

Essmann, H.E., 2009. *Toward innovation capability maturity*. Stellenbosch University, South Africa.

Esterhuizen, D., Schutte, C.S.L. & Toit, du, A.S.A., 2012. Knowledge creation processes as critical enablers for innovation. *International Journal of Information Management*, 32(4), pp.354–364.

Etzkowitz, H. & Leydesdorff, L., 2000. The dynamics of innovation: from National Systems and “Mode 2” to a Triple Helix of university–industry–government relations. *Research Policy*, 29(2), pp.109–123.

Etzkowitz, H. et al., 2000. The future of the university and the university of the future: evolution of ivory tower to entrepreneurial paradigm. *Research Policy*, 29(2), pp.313–330.

Fatolahi, A. & Shams, F., 2006. An investigation into applying UML to the Zachman framework. *Information Systems Frontiers*, 8(2), pp.133–143.

Finkelstein, C., 2006. *Enterprise Architecture for Integration*, Artech House.

Fischer, M.M. & Varga, A., 2002. Technological innovation and interfirm cooperation: an exploratory analysis using survey data from manufacturing firms in the metropolitan region of Vienna. *International Journal of Technology Management*, 24(7-8), pp.724–742.

Flynn, M. et al., 2003. Idea management for organisational innovation. *International Journal of Innovation Management*, 7(04), pp.417–442.

Forfás, 2004. *Innovation Networks*, Dublin: Forfás. Available at: http://www.forfas.ie/publication/search.jsp?ft=/publications/2004/Title,821,en.php.

Freeman, C., 1991. Networks of innovators: a synthesis of research issues. *Research Policy*, 20(5), pp.499–514.

Galanakis, K., 2006. Innovation process. Make sense using systems thinking. *Technovation*, 26(11), pp.1222–1232.

Gallupe, R.B., 2007. The tyranny of methodologies in information systems research. *The Database for Advances in Information Systems*, 38(3), pp.20–28.

Geels, F.W., 2004. From sectoral systems of innovation to socio-technical systems. *Research Policy*, 33(6-7), pp.897–920.

Gemünden, H.G., Heydebreck, P. & Herden, R., 1992. Technological interweavement: a means of achieving innovation success. *R&D Management*, 22(4), pp.359–376.

Gemünden, H.G., Ritter, T. & Heydebreck, P., 1996. Network configuration and innovation success: An empirical analysis in German high-tech industries. *International Journal of Research in Marketing*, 13(5), pp.449–462.

Gerber, A., van der Merwe, A. & Bayes, K., 2013. An investigation into UML case tool support for the Zachman Framework. In 2013 Enterprise Systems Conference (ES). IEEE, pp. 1–9.

Gerybadze, A., 2004. Knowledge management, cognitive coherence, and equivocality in distributed innovation processes in MNCs. *MIR: Management International Review*, pp.103–128.

Gharajedaghi, J., 2011. *Systems Thinking* Third, Elsevier.

Giachetti, R.E., 2010. *Design of Enterprise Systems*, Boca Raton: CRC Press.

Goldkuhl, G., 2012. Design Research in search for a Paradigm: Pragmatism is the answer. *Communications in Computer and Information Science*, 286, pp.84–95.

Goldkuhl, G., 2011. Pragmatism vs interpretivism in qualitative information systems research. *European Journal of Information Systems*, 21(2), pp.135–146.

Gous, H. et al., 2011. Business architecture for inter-organisational innovation networks: A case study comparison from South Africa and Germany. In 17th International Conference on Concurrent Enterprising. Aachen, Germany: IEEE.

Grant, R.M., 1996. Prospering in Dynamically-Competitive Environments: Organizational Capability as Knowledge Integration. *Organization science*, 7(4), pp.375–387.

Greefhorst, D., Koning, H. & Van Vliet, H., 2006. The many faces of architectural descriptions. *Information Systems Frontiers*, 8(2), pp.103–113.

Gregor, S., Hart, D. & Martin, N., 2007. Enterprise architectures: enablers of business strategy and IS/IT alignment in government. *Information Technology & People*, 20(2), pp.96–120.

Grosz, B.J. & Kraus, S., 1996. Collaborative plans for complex group action. *Artificial Intelligence*, 86 (2), pp.269-357.

Gulati, R., Nohria, N. & Zaheer, A., 2000. Strategic networks. *Strategic management journal*, 21(3), pp.203–215.

Hagedoorn, J., 2002. Inter-firm R&D partnerships: an overview of major trends and patterns since 1960. *Research Policy*, 31(4), pp.477–492.

Hamel, G., 1996. *Strategy as Revolution*, Harvard Business Review.

Hamel, G., 2006. The why, what, and how of management innovation. *Harvard Business Review*.

Hamel, G. & Prahalad, C.K., 1994. *Competing for the Future*, Harvard Business Press.

Hanschke, I., 2009. *Strategic IT Management*, Springer.

Harris, K., 2011. Use Innovation Network Design to Unleash Open Innovation. *Gartner Research*, pp.1–18.

Hayek, F.A., 1945. The use of knowledge in society. *The American economic review*, 35(4), pp.519–530.

Henderson, R. & Cockburn, I., 1996. Scale, scope, and spillovers: the determinants of research productivity in drug discovery. *The RAND Journal of Economics*, 27(1), pp.32–59.

Hesse-Biber, S.N. & Leavy, P., 2010. *The Practice of Qualitative Research*, SAGE.

Hevner, A.R., 2007. A Three Cycle View of Design Science Research. *Scandinavian Journal of Information Systems*, 19(2), pp.87–92.

Hevner, A.R., 2012. Design Science Research Workshop.

Hevner, A.R. & Chatterjee, S., 2010a. Design Science Research Frameworks. In *Design Research in Information Systems*. Integrated Series in Information Systems. Boston, MA: Springer US, pp. 23–31.

Hevner, A.R. & Chatterjee, S., 2010b. Design Science Research in Information Systems. In *Design Research in Information Systems*. Integrated Series in Information Systems. Boston, MA: Springer US, pp. 9–22.

Hevner, A.R. et al., 2004. Design science in information systems research. *MIS quarterly*, 28(1), pp.75–105.

Himmelman, A.T., 2001. On Coalitions and the Transformation of Power Relations: Collaborative Betterment and Collaborative Empowerment - Springer. *American Journal of Community Psychology*, 29(2), pp.277–284.

Hitchins, D.K., 2003. *Advanced Systems Thinking, Engineering, and Management*, Artech House.

Hofstee, E., 2006. *Constructing a Good Dissertation*, Sandton, South Africa: EPE.

Hoogervorst, J.A.P., 2009. *Enterprise Governance and Enterprise Engineering*, Springer.

Howells, J., 2006. Intermediation and the role of intermediaries in innovation. *Research Policy*, 35(5), pp.715–728.

Howells, J., James, A. & Malik, K., 2003. The sourcing of technological knowledge: distributed innovation processes and dynamic change. *R&D Management*, 33(4), pp.395–409.

Huizingh, E.K.R.E., 2010. Open innovation: State of the art and future perspectives. *Technovation*, 3(1), pp.2–9.

Hunter, M.G., 2004. Qualitative Research in Information Systems: an Exploration of Methods. In M. E. Witman & A. B. Woszyczynski, eds. *The Handbook of Information Systems Research*. London: Idea Group Publishing.

Huston, L. & Sakkab, N., 2006. Connect and develop. *Harvard Business Review*, 84(3), pp.58–66.

Iansiti, M. & Clark, K.B., 1994. Integration and dynamic capability: evidence from product development in automobiles and mainframe computers. *Industrial and Corporate Change*, 3(3), pp.557–605.

IFIP-IFAC Task Force, 2003. GERAM: The Generalised Enterprise Reference Architecture and Methodology. In P. Bernus, L. Nemes, & G. Schmidt, eds. *Handbook on Enterprise Architecture*. Berlin Heidelberg: Springer, pp. 21–63.

Iivari, J., 2007. A paradigmatic analysis of information systems as a design science. *Scandinavian Journal of Information Systems*, 19(2), p.39-64.

Iivari, J. & Venable, J.R., 2009. Action research and design science research - Seemingly similar but decisively dissimilar. In 17th European Conference on Information Systems (ECIS). Verona.

Järvinen, P., 2007. Action research is similar to design science. *Quality & Quantity*, (41), pp.37–54.

Johannessen, J.-A., Olsen, B. & Olaisen, J., 1999. Aspects of innovation theory based on knowledge-management. *International Journal of Information Management*, 19(2), pp.121–139.

Jonkers, H. et al., 2006. Enterprise architecture: Management tool and blueprint for the organisation. *Information Systems Frontiers*, 8(2), pp.63–66.

Kaisler, S.H., Armour, F. & Valivullah, M., 2005. Enterprise Architecting: Critical Problems. In 38th Annual Hawaii International Conference on System Sciences. IEEE, pp. 224b–224b.

Kalthoff, O., Nonaka, I. & Nueno, P., 1998. *La luz y la sombra*, Bilbao: Deusto.

Kang, D. et al., 2010. An ontology-based Enterprise Architecture. *Expert Systems With Applications*, 37(2), pp.1456–1464.

Kappelman, L., 2011. *The SIM guide to enterprise architecture*, Boca Raton: CRC Press.

Kappelman, L. & Zachman, J.A., 2013. The Enterprise and Its Architecture: Ontology & Challenges. *Journal of Computer Information Systems*, 53(4), pp.87–95.

Katz, B., 2005. Enterprise Engineering, Greenfields Design & Enterprise Change. *Post Graduate course in Enterprise Engineering*.

Katzy, B., Zhang, C. & Löh, H., 2005. Reference Models for Virtual Organisations. In Boston: Kluwer Academic Publishers, pp. 45–58.

Khan, K.S. et al., 2003. Five steps to conducting a systematic review. *Journal of The Royal Society of Medicine*, 96, pp.118–121.

Kosanke, K., Vernadat, F. & Zelm, M., 1999. CIMOSA: enterprise engineering and integration. *Computers in Industry*, 40(2-3), pp.83–97.

Kosanke, K., Vernadat, F.B. & Zelm, M., 2000. Enterprise Engineering and Integration in the Global Environment. In *Advances in Networked Enterprises*. Boston, MA: Springer US, pp. 61–70.

Koskinen, K.U. & Vanharanta, H., 2002. The role of tacit knowledge in innovation processes of small technology companies. *International Journal of Production Economics*, 80(1), pp.57–64.

Kothari, C.R., 2004. *Research Methodology*, New Age International.

Kotlarsky, J., 2008. *Knowledge Processes in Globally Distributed Contexts*, Palgrave Macmillan.

Krogh, von, G., Nonaka, I. & Aben, M., 2001. Making the Most of Your Company's Knowledge: A Strategic Framework. *Long range planning*, 34, pp.421–439.

Kruss, G. et al., 2006. *Creating knowledge networks*, Cape Town: HSRC Press.

Kuczynski, A., Stokic, D. & Kirchhoff, U., 2005. Set-up and maintenance of ontologies for innovation support in extended enterprises. *The International Journal of Advanced Manufacturing Technology*, 29(3-4), pp.398–407.

Kuechler, B. & Vaishnavi, V., 2011. Extending prior research with design science research: two patterns for DSRIS project generation. In *Service-Oriented Perspectives in Design Science Research*. Springer, pp. 166–175.

Kühne, T., 2006. Matters of (meta-) modeling. *Software & Systems Modeling*, 5(4), pp.369–385.

Kvale, S., 2008. *Doing Interviews*, SAGE.

Lakhani, K.R. & Panetta, J.A., 2007. The principles of distributed innovation. *innovations*, 2(3), pp.97–112.

Lapalme, J., 2012. Three schools of thought on enterprise architecture. *IT Professional*, 14(6), pp.37–43.

Lavie, D., 2006. The competitive advantage of interconnected firms: An extension of the resource-based view. *Academy of management review*, 31(3), pp.638–658.

Lee, S. et al., 2010. Open innovation in SME's - An intermediated network model. *Research Policy*, 39(2), pp.290–300.

Lee, S.M., Olson, D.L. & Trimi, S., 2012. Co-innovation: convergenomics, collaboration, and co-creation for organizational values. *Management Decision*, 50(5), pp.817–831.

Leedy, P.D. & Ormrod, J.E., 2001. *Practical research* Ninth, New Jersey: Pearson Merrill Prentice Hall.

Leeuwis, C. & Aarts, N., 2011. Rethinking Communication in Innovation Processes: Creating Space for Change in Complex Systems. *The Journal of Agricultural Education and Extension*, 17(1), pp.21–36. Available at: http://ifsa.boku.ac.at/cms/fileadmin/Proceeding2010/2010_WS1.1_Cees.pdf.

Leiponen, A., 2006. Managing Knowledge for Innovation: The Case of Business-to-Business Services. *Journal of Product Innovation Management*, 23(3), pp.238–258.

Liebeskind, J.P. et al., 1996. Social networks, Learning, and Flexibility: Sourcing Scientific Knowledge in New Biotechnology Firms. *Organization science*, 7(4), pp.428–443.

Liles, D.H., Johnson, M.E. & Meade, L., 1995. The enterprise engineering discipline. In Proceedings of the Fifth Annual Industrial Engineering Research Conference. Orlando, Florida.

Lundvall, B.A., 1988. Innovation as an interactive process: user-producer interaction to the national system of innovation: research paper. In G. Dosi & N. Freeman, eds. *Technical change and economic theory*. Pisa, Italy: Laboratory of Economics and Management.

Lutters, D., Vaneker, T.H.J. & van Houten, F.J.A.M., 2004. “What-if” design: a synthesis method in the design process.

CIRP Annals - Manufacturing Technology, 53(1), pp.113–116.

Lynn, L.H., Mohan Reddy, N. & Aram, J.D., 1996. Linking technology and institutions: the innovation community framework. *Research Policy*, 25(1), pp.91–106.

Marais, S., 2010. *The definition and development of Open Innovation models to assist the innovation process*. Stellenbosch University, South Africa.

March, S.T. & Smith, G.F., 1995. Design and natural science research on information technology. *Decision support systems*, 15(4), pp.251–266.

Markman, G.D. et al., 2005. Innovation speed: Transferring university technology to market. *Research Policy*, 34(7), pp.1058–1075.

Martin, J., 1995. *The Great Transition*, AMACOM/American Management Association.

McAdam, R., 2000. Knowledge management as a catalyst for innovation within organizations: a qualitative study. *Knowledge and process management*, 7(4), pp.233–241.

McEvily, B. & Zaheer, A., 1999. Bridging Ties: A Source of Firm Heterogeneity in Competitive Capabilities. *Strategic management journal*, 20(12), pp.1133–1156.

McPhee, C., Westerlund, M. & Leminen, S. eds., 2012. *Living Labs*, Technology Innovation Management Review.

Mead, G.H. & Morris, C.W., 1938. *The Philosophy of the Act*, Chicago: University of Chicago.

Mehandjiev, N. & Grefen, P., 2012. *Dynamic Business Process Formation for Instant Virtual Enterprises*, Springer-Verlag New York Incorporated.

Melese, T. et al., 2009. Open innovation networks between academia and industry: an imperative for breakthrough therapies. *Nature Medicine*, 15(5), pp.502–507.

Mentzer, J.T. et al., 2001. Defining Supply Chain Management. *Journal of Business Logistics*, 22(2), pp.1–25.

Miedema, J. et al., 2007. Synergy of technical specifications, functional specifications and scenarios in requirements specifications. In *The Future of Product Development*. Berlin Heidelberg: Springer, pp. 235–245.

MIH ed., 2014. *MIH*, Available at: http://www.naspers.com/company-structure.php [Accessed April 28, 2014].

Moore, G.A., 2008. *Dealing with Darwin*, Portfolio Trade.

Mouton, J., 2001. *How to succeed in your master's and doctoral studies*, Van Schaik Publishers.

Muller, G. & Hole, E., 2006. *Architectural descriptions and models*, Washington: White Paper Resulting from Architecture Forum Meeting.

Myers, M.D., 2007. Introduction to Qualitative Research. Johannesburg.

Myers, M.D., 1997. Qualitative research in information systems. *Management Information Systems Quarterly*, 21(2).

Myers, M.D. & Avison, D., 2002. *Qualitative research in information systems*, London: Sage Publications.

Newman, B.D. & Conrad, K.W., 2000. A Framework for Characterizing Knowledge Management Methods, Practices, and Technologies. In Third International Conference on Practical Aspects of Knowledge Management. Basel, Switzerland.

Nieto, M.J. & Santamaria, L., 2007. The importance of diverse collaborative networks for the novelty of product innovation. *Technovation*, 27(6), pp.367–377.

Ning, K. et al., 2006. Semantic innovation management across the extended enterprise. *International Journal of Industrial and Systems Engineering*, 1(1), pp.109–128.

Nohria, N. & Eccles, R.G., 1992. Networks and Organizations: Structure, Form and Action. *Harvard Business School Press*.

Nonaka, I., 1994. A dynamic theory of organizational knowledge creation. *Organization science*, 5(1), pp.14–37.

Nonaka, I., 1991. The knowledge creating company. *Harvard Business Review*, pp.96–104.

Nonaka, I., 1995. *The Knowledge-creating Company: How Japanese Companies Create the Dynamics of Innovation*, New York: Oxford University Press.

Nonaka, I. & Krogh, von, G., 2009. Tacit Knowledge and Knowledge Conversion: Controversy and Advancement in Organizational Knowledge Creation Theory. *Organization science*, 20(3), pp.635–652.

Nonaka, I. & Nishiguchi, T. eds., 2000. *Knowledge Emergence: Social, Technical, and Evolutionary Dimensions of Knowledge Creation*, Oxford University Press.

Nonaka, I., Toyama, R. & Konno, N., 2000. SECI, Ba and Leadership: a Unified Model of Dynamic Knowledge Creation. *Long range planning*, 33, pp.5–34.

Noran, O., 2003. An analysis of the Zachman framework for enterprise architecture from the GERAM perspective. *Annual Reviews in Control*, 27(2), pp.163–183.

Noran, O.S., 2004. *Enterprise reference architectures and modelling frameworks*. Griffith University, Australia.

Ojasalo, J., 2008. Management of innovation networks: a case study of different approaches. *European Journal of Innovation Management*, 11(1), pp.51–86.

Okhuysen, G.A. & Eisenhardt, K.M., 2002. Integrating Knowledge in Groups: How Formal Interventions Enable Flexibility. *Organization science*, 13(4), pp.370–386.

Oliver, A.L., 2009. *Networks for Learning and Knowledge Creation in Biotechnology*, Cambridge University Press.

Oliver, A.L. & Ebers, M., 1998. Networking Network Studies: An Analysis of Conceptual Configurations in the Study of Inter-organizational Relationships. *Organization Studies*, 19(4), pp.549–583.

Olivier, M.S., 2004. *Information Technology Research: A practical guide for Computer Science and Informatics* Second, Van Schaik Publishers.

Orlikowski, W.J., 2002. Knowing in Practice: Enacting a Collective Capability in Distributed Organizing. *Organization science*, 13(3), pp.249–273.

Orlikowski, W.J. & Baroudi, J.J., 1991. Studying Information Technology in Organizations: Research Approaches and Assumptions. *Information systems research*, 2(1), pp.1–28.

Owen, C.L., 1998. Design research: building the knowledge base. *Design Studies*, 19(1), pp.9–20.

Oxford Dictionaries, Artefact. *oxforddictionaries.com*. Available at: http://www.oxforddictionaries.com/definition/english/artefact [Accessed June 13, 2014].

Oxford Dictionaries, Epistemology. Available at: http://www.oxforddictionaries.com/definition/english/epistemology [Accessed June 9, 2014].

Oxford Dictionaries, Knowledge. Available at: http://www.oxforddictionaries.com/definition/english/knowledge [Accessed May 5, 2014].

Oxford Dictionaries, Objective. *oxforddictionaries.com*. Available at: http://www.oxforddictionaries.com/definition/english/objective [Accessed October 14, 2013].

Oxford Dictionaries, Requirement. *oxforddictionaries.com*. Available at: http://www.oxforddictionaries.com/definition/english/requirement [Accessed October 14, 2013].

Peffers, K. et al., 2008. A Design Science Research Methodology for Information Systems Research. *Journal of Management Information Systems*, 24(3), pp.45–77.

Peffers, K. et al., 2012. Design Science Research Evaluation. In *Design Science Research in Information Systems. Advances in Theory and Practice*. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 398–410.

Perks, C. & Beveridge, T., 2011. *Guide to Enterprise IT Architecture*, Springer-Verlag New York Incorporated.

Perry, N. & Uys, W., 2010. Knowledge integration based on roadmapping and conceptual framework approach to ease innovation management. *International Journal of Computer Applications in Technology*, 37(3/4), p.165.

Pérez-Bustamante, G., 1999. Knowledge management in agile innovative organisations. *Journal of Knowledge Management*, 3(1), pp.6–17.

Piller, F., 2008. Interactive value creation with users and customers. *citeseerx.ist.psu.edu*. Available at: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.91.8463&rep=rep1&type=pdf [Accessed May 14, 2014].

Pittaway, L. et al., 2004. Networking and innovation: a systematic review of the evidence. *International Journal of Management Reviews*, 5(3-4), pp.137–168.

Polanyi, M., 1962. Tacit Knowing: Its Bearing on Some Problems of Philosophy. *Reviews of Modern Physics*, 34(4), pp.601–616.

Pollard, D. ed., 2005. *Will That Be Coordination, Cooperation, or Collaboration?* Available at: http://howtosavetheworld.ca/2005/03/25/will-that-be-coordination-cooperation-or-collaboration/ [Accessed May 1, 2014].

Powell, W.W., Koput, K.W. & Smith-Doerr, L., 1996. Interorganizational collaboration and the locus of innovation: Networks of learning in biotechnology. *Administrative science quarterly*, 41(1), pp.116–145.

Prahalad, C.K. & Ramaswamy, V., 2004a. Co-creation experiences: The next practice in value creation. *Journal of interactive marketing*, 18(3), pp.5–14.

Prahalad, C.K. & Ramaswamy, V., 2004b. *The future of competition: Co-creating unique value with customers*, Boston: Harvard Business Press.

Pries-Heje, J., Baskerville, R. & Venable, J.R., 2008. Strategies for design science research evaluation. In European Conference on Information Systems (ECIS).

Provan, K.G. & Human, S.E., 1999. Organizational learning and the role of the network broker in small-firm manufacturing networks. In A. Grandori, ed. *Interfirm Networks: Organization and Industrial Competitiveness*. Routledge, p. 336.

Pyka, A., 2002. Innovation networks in economics: from the incentive-based to the knowledge-based approaches. *European Journal of Innovation Management*, 5(3), pp.152–163.

Raimann, J., 2000. *Performance Measurement of Communities of Practice*, St. Gallen: Research Center, KnowledgeSource, University of St. Gallen.

Raimann, J. et al., 2000. *Supporting Business Processes through Knowledge Management – A Technology-based Analysis*, St. Gallen: Research Center, KnowledgeSource.

Rechtin, E., 1999. *Systems Architecting of Organizations*, CRC Press.

Rechtin, E. & Maier, M.W., 2000. *The Art of Systems Architecting* Second, CRC Press.

RLabs ed., 2014. *RLabs*, Available at: http://za.rlabs.org/ [Accessed April 28, 2014].

Robertson, P.L. & Langlois, R.N., 1995. Innovation, networks, and vertical integration. *Research Policy*, 24(4), pp.543–562.

Romero, D. & Molina, A., 2011. Collaborative networked organisations and customer communities: value co-creation and co-innovation in the networking era. *Production Planning & Control*, 22(5-6), pp.447–472.

Rosas, J., Macedo, P. & Camarinha-Matos, L.M., 2011. Extended competencies model for collaborative networks. *Production Planning & Control*, 22(5-6), pp.501–517.

Rosenbleuth, A., Weiner, N. & Bigelow, J., 1943. Behavior, Purpose, and Teleology. *Philosophy of Science*, 10(1), pp.18–24.

Ross, J.W., Weill, P. & Robertson, D., 2013. *Enterprise Architecture As Strategy*, Harvard Business Press.

Rothberg, R.R., 1981. *Corporate strategy and product innovation*, New York: The Free Press.

Rothwell, R., 1992. Successful industrial innovation: critical factors for the 1990s. *R&D Management*, 22(3), pp.221–240.

Rouse, W.B., 2004. Embracing the enterprise. *Industrial engineer*, 36(3), pp.31–35.

Saenz, O.A. et al., 2009. Defining Enterprise Systems Engineering. *International Journal of Industrial and Systems Engineering*, 4(5), p.483.

Salvendy, G., 2001. *Handbook of Industrial Engineering*, John Wiley & Sons.

Sammarra, A. & Biggiero, L., 2008. Heterogeneity and specificity of Inter-Firm knowledge flows in innovation networks. *Journal of Management Studies*, 45(4), pp.800–829.

Sanders, E.B.N. & Stappers, P.J., 2008. Co-creation and the new landscapes of design. *Co-design*, 4(1), pp.5–18.

Schekkerman, J., 2004. *How to Survive in the Jungle of Enterprise Architecture Frameworks*, Trafford Publishing.

Schön, B. & Pyka, A., 2012. A taxonomy of innovation networks. In University of Hohenheim. FZID discussion papers.

Schönström, M., 2005. Creating knowledge networks: lessons from practice. *Journal of Knowledge Management*, 9(6), pp.17–29.

Schumpeter, J.A., 1934. *The Theory of Economic Development*, Cambridge, Massachusetts: Harvard University Press.

Schumpeter, J.A., 1939. *Business Cycles. A Theoretical, Historical, And Statistical Analysis of the Capitalist Process*, New York: McGraw-Hill.

Schutte, C., 2010. *Executing innovation projects using the collaborative nature of integrated knowledge networks*. Stellenbosch University, South Africa.

Schutte, C. & Du Preez, N.D., 2008. Knowledge networks for managing innovation projects. In PICMET. Cape Town.

Seufert, A., Krogh, G.V. & Bach, A., 1999. Towards knowledge networking. *Journal of Knowledge Management*, 3(3), pp.180–190.

Sidorova, A. & Kappelman, L., 2010. Enterprise Architecture as Politics: An Actor-Network Perspective. In L. Kappelman, ed. *The SIM Guide to Enterprise Architecture*. Boca Raton: CRC Press.

Simmie, J. et al., 2002. Innovation in Europe: A Tale of Networks, Knowledge and Trade in Five Cities. *Regional Studies*, 36(1), pp.47–64.

Simon, H.A., 1996. *The Sciences of the Artificial* 3rd ed., MIT Press.

Smuts, J.L., 2011. *A Knowledge Management Framework for Information Systems Outsourcing*. University of South Africa.

Sorli, M. et al., 2004. Fostering Innovation in Concurrent Enterprising. In 10th International Conference on Concurrent Enterprising. Sevilla, Spain.

Sowa, J.F. & Zachman, J.A., 1992. Extending and formalizing the framework for information systems architecture. *IBM systems journal*, 31(3), pp.590–616.

Spewak, S.H. & Hill, S.C., 1993. *Enterprise architecture planning*, QED Information Sciences.

Spithoven, A., Clarysse, B. & Knockaert, M., 2011. Building absorptive capacity to organise inbound open innovation in traditional industries. *Technovation*, 31(1), pp.10–21.

Steinke, I., 2004. Quality criteria in qualitative research. In U. Flick, E. van Kardoff, & I. Steinke, eds. *A Companion to Qualitative Research*. London: SAGE Publications.

Sun, Y. & Kantor, P.B., 2006. Cross-Evaluation: A new model for information system evaluation. *Journal of the American Society for Information Science and Technology*, 57(5), pp.614–628.

Surowiecki, J., 2005. *The Wisdom of Crowds*, London: Abacus.

Swan, J. & Scarbrough, H., 2005. The politics of networked innovation. *Human relations*, 58(7), pp.913–943.

Swan, J. et al., 1999. Knowledge management and innovation: networks and networking. *Journal of Knowledge Management*, 3(4), pp.262–275.

Szmytkowski, D., 2005. Innovation definition comparative assessment. *GNU Free Documentation License*.

Szulanski, G., 1996. Exploring internal stickiness: Impediments to the transfer of best practice within the firm. *Strategic management journal*, 17(Winter), pp.27–43.

Sørensen, C. & Lundh-Snis, U., 2001. Innovation through knowledge codification. *Journal of Information Technology*, 16(2), pp.83–97.

Taatila, V.P. et al., 2006. Framework to study the social innovation networks. *European Journal of Innovation Management*, 9(3), pp.312–326.

Takeda, H., Veerkamp, P. & Yoshikawa, H., 1990. Modeling design process. *AI magazine*, 11(4), pp.37–48.

Terre Blanche, M. & Durrheim, K., 2006. Histories of the Present: Social Science Research in Context. In M. Terre Blanche, K. Durrheim, & D. Painter, eds. *Research in Practice: Applied Methods for The Social Sciences*. Cape Town: University of Cape Town Press.

The Open Group ed., 2011. *TOGAF® Version 9.1*, Available at: https://www2.opengroup.org/ogsys/jsp/publications/PublicationDetails.jsp?catalogno=I112 [Accessed June 12, 2014].

Tidd, J. & Bessant, J., 2011. *Managing Innovation* 4 ed., John Wiley & Sons.

Toffler, A., 1990. *The Third Wave*, Bantam.

Towill, D.R., 1997. Successful business systems engineering. Part 1: The systems approach to business processes. *Engineering Management Journal*, 7(1), pp.55–64.

Trauth, E.M. & Jessup, L.M., 2000. Understanding computer-mediated discussions: positivist and interpretive analyses of group support system use. *MIS quarterly*, 24(1), pp.43–79.

Tsai, W., 2001. Knowledge transfer in intraorganizational networks: effects of network position and absorptive capacity on business unit innovation and performance. *Academy of management journal*, 44(5), pp.996–1004.

Tushman, M.L. & Anderson, P., 1986. Technological discontinuities and organizational environments. *Administrative science quarterly*, 31(3), pp.439–465.

Tølle, M., Bernus, P. & Vesterager, J., 2002. Reference Models for Virtual Enterprises. L. M. Camarihna-Matos, ed. *Collaborative Business Ecosystems and Virtual Enterprises*, 85, pp.3–10.

Vaishnavi, V. & Kuechler, B., 2004. Design Science Research in Information Systems. *desrist.org*. Available at: http://desrist.org/design-research-in-information-systems/ [Accessed August 27, 2013].

Vaishnavi, V. & Kuechler, B., 2007. *Design science research methods and patterns: innovating information and communication technology*, Suite 300 Boca Raton, FL: Auerbach Publications, Taylor & Francis Group.

Van de Ven, A.H., 2005. Running in packs to develop knowledge-intensive technologies. *MIS quarterly*, pp.365–377.

Van den Berg, M. & van Steenbergen, M., 2007. *Building an Enterprise Architecture Practice*, Springer.

Van Zyl, H., 2006. *Innovation Models and the Front-end of Product Innovation*. Stellenbosch University, South Africa.

Vanhaverbeke, W. & Cloodt, M., 2006. Open innovation in value networks. In H. Chesbrough, W. Vanhaverbeke, & J. West, eds. *Open Innovation: Researching a New Paradigm*. Open Innovation: Researching a New Paradigm. Oxford University Press.

Venable, J., 2006a. A Framework for Design Science Research Activities. In *Emerging Trends and Challenges in Information Technology Management*. Hershey, PA: Idea Group Publishing.

Venable, J., 2006b. The Role of Theory and Theorising in Design Science Research. In DESRIST. Claremont, CA.

Venable, J., Pries-Heje, J. & Baskerville, R., 2012. A Comprehensive Framework for Evaluation in Design Science Research. In *Design Science Research in Information Systems. Advances in Theory and Practice*. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 423–438.

Vernadat, F., 1996. *Enterprise Modeling and Integration*, Taylor & Francis US. Available at: http://sic.ici.ro/sic1997_3/VERNA.html.

Vesterager, J., Tølle, M. & Bernus, P., 2003. VERA: virtual enterprise reference architecture. In VTT SYMPOSIUM.

Von Hippel, E., 2005. *Democratizing innovation*, The MIT Press.

Von Hippel, E., 2007a. Horizontal innovation networks - by and for users. *Industrial and Corporate Change*, 16(2), pp.293–315.

Von Hippel, E., 2002. *Open Source Projects as Horizontal Innovation Networks - by and for Users*, MIT Sloan School of Management.

Von Hippel, E., 2007b. *The sources of innovation*, Oxford University Press, USA.

Von Hippel, E., 1994. "Sticky Information" and the Locus of Problem Solving: Implications for Innovation. *Management science*, 40(4), pp.429–439.

Walls, J.G., Widmeyer, G.R. & Sawy, El, O.A., 1992. Building an information system design theory for vigilant EIS. *Information systems research*, 3(1), pp.36–59.

Warkentin, M., Bapna, R. & Sugumaran, V., 2001. E-knowledge networks for inter-organizational collaborative e-business. *Logistics Information Management*, 14(1-2), pp.149–163.

Webster, J. & Watson, R.T., 2002. Analyzing the Past to Prepare for the Future: Writing A Literature Review. *MIS quarterly*, 26(2), pp.13–23.

Weinberg, G.M., 2001. *An introduction to general systems thinking*, Dorset House.

Wenger, E., 1998. *Communities of Practice*, Cambridge University Press.

Wikström, S. et al., 1994. *Knowledge and value: A new perspective on corporate transformation*, London: Routledge.

Williams, T.J., 1998. The Purdue Enterprise Reference Architecture and Methodology (PERA). In *Handbook of Life Cycle Engineering*. Springer.

Williams, T.J., 1994. The Purdue Enterprise Reference Architecture. *Computers in Industry*, 24(2-3), pp.141–158.

Williams, T.J. & Li, H., 1999. PERA and GERAM—enterprise reference architectures in enterprise integration. In J. J. Mills & F. Kimura, eds. *Information Infrastructure Systems for Manufacturing II*. Information Infrastructure Systems for Manufacturing II. Boston, MA: Springer US, pp. 3–30.

Williams, T.J. et al., 1994. Architectures for integrating manufacturing activities and enterprises. *Computers in Industry*, 24(2), pp.111–139.

Williams, T.J. et al., 1998. The Life Cycle of an Enterprise. In *Handbook of Life Cycle Engineering*. Springer.

Williams, T.J., Rathwell, G.A. & Li, H., 1996. *A handbook on master planning and implementation for enterprise integration programs*, Purdue Laboratory for Applied Industrial Control.

Winograd, T.A. & Flores, C.F., 1986. *Understanding computers and cognition: A new foundation for design*, Boston: Addison-Wesley.

Wolpert, J.D., 2002. Breaking out of the innovation box. *Harvard Business Review*, 80(8), pp.76–83, 148.

Yin, R.K., 2003. *Case Study Research*, SAGE.

Yoo, Y., Lyytinen, K. & Boland, R.J., 2008. Distributed innovation in classes of networks. pp.58–58.

Zachman, J.A., 1987. A framework for information systems architecture. *IBM systems journal*, 26(3), pp.276–292.

Zachman, J.A., 1997. Enterprise architecture: The issue of the century. *Database Programming and Design*.

Zachman, J.A., 2008. John Zachman's Concise Definition of The Zachman Framework™. *zachman.com*. Available at: http://www.zachman.com/about-the-zachman-framework [Accessed June 12, 2014].

Zachman, J.A., 2011. *The Zachman Framework for Enterprise Architecture* 3rd ed., Zachman International®. Available at: http://www.zachman.com.

Zander, I., 1999. How do you mean “global?” An empirical investigation of innovation networks in the multinational corporation. *Research Policy*, 28(2), pp.195–213.

Appendix A - Reference model sources

This appendix contains a complete listing of the 100 studies employed in a qualitative systematic review conducted in this study. These studies serve as data sources for the development of the reference models that populate the reference architecture for IKNs in chapter 8. Amongst these sources are works that discuss more generic (e.g. Camarinha-Matos & Afsarmanesh 2008a), as well as more particular (e.g. Kruss 2006) descriptions of IKNs.

Every source was analysed according to structured review questions modelled on the six abstractions of the Zachman framework, i.e. "*What?*", "*How?*", "*Where?*", "*Who?*", "*When?*" and "*Why?*" (refer to 3.5.5.1). This implies an analysis where composite (multi-variable) contributions in the sources are interpreted in primitive (single-variable) terms.

Where a source *briefly mentions* an IKN component (i.e. a row 1 identification) that can be interpreted in terms a particular abstraction, that abstraction is shaded in light blue (▒) for the particular source. Where a source makes a *significant contribution* to the understanding of an IKN component (i.e. a row 2 definition) that can be interpreted in terms of a particular abstraction, that abstraction is shaded in darker blue (▓) for the particular source.

Sources that make significant contributions in terms of a number of abstractions are seen as primary sources for the reference architecture for IKNs, and are indicated in dark blue (█) to the left of Table A.1. These sources and their use are discussed further in section 8.5.1.2 on the nature of the reference models that populate the reference architecture for IKNs.

Table A.1: Complete sources of the reference models

Source	What	How	Where	Who	When	Why
Acha & Cusmano (2005) Governance and co-ordination of distributed innovation processes: patterns of R&D co-operation in the upstream petroleum industry		▓		▒		▓
Adamides & Karacapilidis (2006) Information technology support for the knowledge and social processes of innovation management	▓	▒		▒		
█ **Anklam** (2007) Net Work	▒	▓	▒	▒	▓	▓
Asheim et al. (2011) MNCs between the local and the global: knowledge bases, proximity and distributed knowledge networks			▓			
█ **Back et al.** (2005) Putting Knowledge Networks Into Action	▓	▓	▒	▒	▓	▓

Source	What	How	Where	Who	When	Why
Badii & Sharif (2003) Information management and knowledge integration for enterprise innovation	■	▨				
Becker et al. (2009) Coordinating distributed innovation processes: The case of the automotive and open source software industries		■		▨		■
▌ **Berasategi et al.** (2011) A comprehensive framework for collaborative networked innovation	▨	■	■	■	■	■
▌ **Bogers & West** (2012) Managing Distributed Innovation: Strategic Utilization of Open and User Innovation	■	■	▨	■	▨	■
Brennan & Dooley (2005) Networked creativity: a structured management framework for stimulating innovation	■	▨		▨		▨
▌ **Bullinger et al.** (2004) Managing innovation networks in the knowledge-driven economy	■	▨	■	■	▨	▨
Camarinha-Matos et al. (2011) Collaborative Networks in Support of Service-Enhanced Products			■			
Camarinha-Matos (2009) Collaborative networked organizations: Status and trends in manufacturing	▨	▨		▨	▨	▨
Camarinha-Matos et al. (2009) Collaborative networked organizations – Concepts and practice in manufacturing enterprises	▨	▨	▨	▨	▨	▨
▌ **Camarinha-Matos & Afsarmanesh** (2008a) Collaborative Networks: Reference Modeling	■	■	■	■	■	■
Camarinha-Matos & Afsarmanesh (2008b) On reference models for collaborative networked organizations	▨	▨	▨	▨	▨	▨
Carayannis & Campbell (2006) Knowledge Creation, Diffusion, and Use in Innovation Networks and Knowledge Cluster	■	■	▨	▨	▨	▨
Cavusgil et al. (2003) Tacit knowledge transfer and firm innovation capability	■	■	▨	▨	▨	▨
▌ **Chesbrough & Prencipe** (2008) Networks of innovation and modularity: a dynamic perspective	■	■		■	■	■
Chesbrough et al. (2006a) Open innovation: a new paradigm for understanding industrial innovation	■	■		■	▨	■
▌ **Chesbrough et al.** (2006b) Open Innovation: Researching a New Paradigm	■	■	▨	■	▨	■
Chesbrough (2003) Open innovation: The new imperative for creating and profiting from technology	■	■		■	▨	■

Source	What	How	Where	Who	When	Why
Coombs et al. (2001) Analysing distributed innovation processes: a CRIC position paper	▨	■	▨	▨	▨	■
Cowan & Jonard (2009) Knowledge portfolios and the organization of innovation networks	■	▨		▨	▨	▨
Cowan et al. (2007) Bilateral Collaboration and the Emergence of Innovation Networks	■	■	▨	▨	■	■
Desouza et al. (2009) Crafting organizational innovation processes	■	■		▨	▨	▨
Dhanaraj & Parkhe (2006) Orchestrating innovation networks	▨	▨		■	▨	■
Dooley & O'Sullivan (2007) Managing within distributed innovation networks	■	■	▨	■	▨	▨
Dooley & O'Sullivan (2000) Systems innovation manager	■	▨				
Douthwaite (2006) Enabling innovation: Technology-and system-level approaches that capitalize on complexity	■	▨		■	■	▨
Eschenbächer et al. (2011) Improving distributed innovation processes in virtual organisations through the evaluation of collaboration intensities		■	▨	■	▨	▨
Essmann (2009) Toward innovation capability maturity		▨	▨	■	▨	■
Etzkowitz & Leydesdorff (2000) The dynamics of innovation: from National Systems and “Mode 2” to a Triple Helix of university–industry–government relations		▨		■		▨
Etzkowitz et al. (2000) The future of the university and the university of the future: evolution of ivory tower to entrepreneurial paradigm	▨	▨		■		
Flynn et al. (2003) Idea management for organisational innovation	■	■		▨		▨
Geels (2004) From sectoral systems of innovation to socio-technical systems	■			■	▨	■
Gerybadze (2004) Knowledge management, cognitive coherence, and equivocality in distributed innovation processes in MNCs	■	▨	■	▨	▨	▨
Harris (2011) Use Innovation Network Design to Unleash Open Innovation	▨	▨		▨	▨	▨
Von Hippel (2007a) Horizontal innovation networks - by and for users	▨	▨		■	▨	■
Von Hippel (2007b) The sources of innovation	▨	▨		■	▨	▨

Source	What	How	Where	Who	When	Why
Von Hippel (2005) Democratizing innovation	▒	▒	▒	■	▒	■
Von Hippel (2002) Open Source Projects as Horizontal Innovation Networks - by and for Users	▒	▒		■	▒	■
Von Hippel (1994) “Sticky Information” and the Locus of Problem Solving: Implications for Innovation	■	▒	■	▒		▒
Howells (2006) Intermediation and the role of intermediaries in innovation	▒	■	▒	■	▒	▒
Howells et al. (2003) The sourcing of technological knowledge: distributed innovation processes and dynamic change	■	▒	▒	▒	■	▒
Huizingh (2010) Open innovation: State of the art and future perspectives	■	■	▒	▒	▒	▒
Koskinen & Vanharanta (2002) The role of tacit knowledge in innovation processes of small technology companies	■	■		▒	▒	▒
Kotlarsky (2008) Knowledge Processes in Globally Distributed Contexts	■	■	▒	▒		
Kruss et al. (2006) Creating knowledge networks	■	■	■	■	■	■
Kuczynski et al. (2005) Set-up and maintenance of ontologies for innovation support in extended enterprises	■	▒	▒	■		▒
Lakhani & Panetta (2007) The Principles of Distributed Innovation	■	▒	▒	■		▒
Lavie (2006) The competitive advantage of interconnected firms: An extension of the resource-based view	▒			▒		■
Lee et al. (2012) Co-innovation: convergenomics, collaboration, and co-creation for organizational values	▒	■		■	▒	▒
Lee et al. (2010) Open innovation in SME's - An intermediated network model	▒	■		■	▒	▒
Leeuwis & Aarts (2011) Rethinking Communication in Innovation Processes: Creating Space for Change in Complex Systems	▒	■		▒		■
Lynn et al. (1996) Linking technology and institutions: the innovation community framework		▒		■		▒
Markman et al. (2005) Innovation speed: Transferring university technology to market	▒	■		■		▒

Source	What	How	Where	Who	When	Why
McAdam (2000) Knowledge management as a catalyst for innovation within organizations: a qualitative study	■	■		▨	▨	▨
McEvily & Zaheer (1999) Bridging Ties: A Source of Firm Heterogeneity in Competitive Capabilities		▨		■		▨
McPhee et al. eds. (2012) Living Labs	▨	■	▨	■		■
Melese et al. (2009) Open innovation networks between academia and industry: an imperative for breakthrough therapies	▨	▨		■		▨
Nieto & Santamaria (2007) The importance of diverse collaborative networks for the novelty of product innovation	▨	▨		■		■
Ning et al. (2006) Semantic innovation management across the extended enterprise	■	▨		▨		▨
Nonaka et al. (2000) SECI, Ba and Leadership: a Unified Model of Dynamic Knowledge Creation	■	■		▨		■
Nonaka & Nishiguchi eds. (2000) Knowledge Emergence: Social, Technical, and Evolutionary Dimensions of Knowledge Creation	■	■	▨	▨	▨	
Nonaka (1994) A dynamic theory of organizational knowledge creation	■	■		▨		
Nonaka (1991) The knowledge creating company	■	▨		▨		■
Nonaka & Von Krogh (2009) Tacit Knowledge and Knowledge Conversion: Controversy and Advancement in Organizational Knowledge Creation Theory	■	■		▨		
Ojasalo (2008) Management of innovation networks: a case study of different approaches	▨	▨		▨	▨	■
Oliver (2009) Networks for Learning and Knowledge Creation in Biotechnology	■	■		▨	▨	
Pittaway et al. (2004) Networking and innovation: a systematic review of the evidence	▨	■		■	▨	■
Powell et al. (1996) Inter-organisational collaboration and the locus of innovation: Networks of learning in biotechnology	▨	■		■	▨	▨
Prahalad & Ramaswamy (2004) Co-creation experiences: The next practice in value creation				▨		▨
Du Preez & Louw (2008) A framework for managing the innovation process	▨	■	▨	■	■	■

Source	What	How	Where	Who	When	Why
Du Preez et al. (2008) A knowledge network approach supporting the value chain	■	■	▨	■	■	■
Provan & Human (1999) Organizational learning and the role of the network broker in small-firm manufacturing networks		▨		■		▨
Raimann et al. (2000) Supporting Business Processes Through Knowledge Management	■	■		▨		▨
Robertson & Langlois (1995) Innovation, networks, and vertical integration	▨	■		▨	■	▨
Romero & Molina (2011) Collaborative networked organisations and customer communities: value co-creation and co-innovation in the networking era	▨	▨		■	▨	■
Rosas et al. (2011) Extended competencies model for collaborative networks	▨	▨		▨		■
Sammarra & Biggiero (2008) Heterogeneity and specificity of Inter-Firm knowledge flows in innovation networks	■	■		▨		▨
Sanders & Stappers (2008) Co-creation and the new landscapes of design	■	▨		■	▨	▨
Schön & Pyka (2012) A taxonomy of innovation networks		▨		▨	■	■
Schutte (2010) Executing innovation projects using the collaborative nature of integrated knowledge networks	■	■	▨	■	■	■
Schutte & Du Preez (2008) Knowledge networks for managing innovation projects	■	■	▨	■	■	■
Seufert et al. (1999) Towards knowledge networking	■	▨	▨	▨	▨	▨
Simmie et al. (2002) Innovation in Europe: A Tale of Networks, Knowledge and Trade in Five Cities	■	▨	▨	■		▨
Sørensen & Lundh-Snis (2001) Innovation through knowledge codification	■	▨		▨		
Sorli et al. (2004) Fostering Innovation in Concurrent Enterprising	■	■		▨	▨	▨
Spithoven et al. (2011) Building absorptive capacity to organise inbound open innovation in traditional industries	▨	▨		▨		▨
Swan & Scarbrough (2005) The politics of networked innovation	▨	▨		■	▨	■

Source	**What**	**How**	**Where**	**Who**	**When**	**Why**
Swan et al. (1999) Knowledge management and innovation: networks and networking	■	■	▨	▨		■
Tidd & Bessant (2011) Managing Innovation	■	■	▨	▨	▨	■
Tsai (2001) Knowledge transfer in intra-organizational networks: effects of network position and absorptive capacity on business unit innovation and performance	■	■		▨		▨
Vanhaverbeke & Cloodt (2006) Open innovation in value networks		▨		■		■
Van de Ven (2005) Running in packs to develop knowledge-intensive technologies	■	▨		▨		■
Warkentin et al. (2001) E-knowledge networks for inter-organizational collaborative e-business	■	▨		▨		▨
Wolpert (2002) Breaking out of the innovation box		▨		■		▨
Yoo et al. (2008) Distributed innovation in classes of networks	▨	■		▨		■
Zander (1999) How do you mean "global?" An empirical investigation of innovation networks in the multinational corporation		▨	■	▨		▨

Appendix B – Qualitative systematic review example

This appendix presents an example of the qualitative systematic review that was undertaken to collect and analyse data in the fourth cycle of the research design. Data obtained from this review served as inputs for the development of primitive reference models that populate the reference architecture.

The study that is reviewed in this appendix, Du Preez et al. (2008), was one of 100 studies that were reviewed with the same method (refer to section 3.5.5.1). The following structured questions were asked during the review of each study:

1. *"What?"*, i.e. which generalisable inventory sets are significant to the construction of IKNs? (🟩)
2. *"How?"*, i.e. which generalisable process flows are significant to the construction of IKNs? (🟥)
3. *"Where?"*, i.e. which generalisable distribution networks are significant to the construction of IKNs? (🟨)
4. *"Who?"*, i.e. which generalisable responsibility assignments are significant to the construction of IKNs? (🟦)
5. *"When?"*, i.e. which generalisable timing cycles are significant to the construction of IKNs? (🟪)
6. *"Why?"*, i.e. which generalisable motivation intentions are significant to the construction of IKNs? (◼)

Where the review method encountered a possible answer to any of these questions in a study, the data point was collected and highlighted in the colours indicated above. These data points are of a primitive (single-variable) nature, and could therefore be directly employed as inputs for the development of the primitive reference models that populate the reference architecture for IKNs. Where significant composite (multi-variable) data points were encountered, these were also highlighted (▨) and subsequently deconstructed into primitive data points.

The complete set of data from the qualitative systematic review, i.e. from the 100 studies reviewed, was analysed to identify normalised, generic architecture patterns and components. These patterns and components formed the basis of the identification models that populate the first row of the reference architecture for IKNs. These first row models, as well as definitions for the identified architecture components as provided by the data from the review, were used as inputs for the development of the business concept definition models that populate the second row of the reference architecture for IKNs.

The following review of the study "A knowledge network approach supporting the value chain" by Du Preez et al. (2008), is accompanied by a synthesis of the data collected from the study in Table B.1. As discussed above, this data was included as an input to the primitive reference models presented in section 8.5.2.

A knowledge network approach supporting the value chain

Niek Du Preez[1], Louis Louw[1], Eric Lutters.[2]

[1] Global Competitiveness Centre (GCC) in Engineering,
Stellenbosch University, South Africa.
nddp@sun.ac.za; Louis@indutech.co.za
[2] Faculty of Engineering Technology,
University of Twente, Enschede, The Netherlands
e.lutters@utwente.nl

Abstract

Pro-active management of the knowledge supply chain facilitates rapid technology, product and enterprise innovation. Collaboration has become an imperative for innovation. The knowledge "explosion" and abundant connectivity hampers rapid innovation and leads to communication overload. Structuring collaborative knowledge, exchanged via an integrated knowledge network, fosters the rapid exploitation of knowledge. An adequate (adaptable) configuration of network components within a domain of knowledge is required. This paper provides a framework for such an Integrated Knowledge Network (IKN); it also provides a navigation space to access knowledge contextualized with project life cycles. A practical case study that facilitates innovation research in this manner, spanning different private and public domains and including more than 100 projects, 130 users and in excess of 30 000 documents is briefly discussed.

Keywords

Design methodology; Knowledge management; Innovation management

1. Introduction

All enterprises are increasingly under pressure to innovate in order to compete [1]. Some of the reasons are changing markets and intensified competition, the rapid pace of technological change, product complexity and globalisation. Enterprise competitiveness is promoted by the innovation of products, processes and technologies, also referred to as a material supply chain, supported by a knowledge supply chain [2].

A Knowledge Network signifies a number of people and resources, and the relationships between them, that are able to capture, transfer and create knowledge for the purpose of creating value. An Integrated Knowledge Network spans all domains, communities, and trust relationships with the goal of fostering sustainable innovation that will continue to promote the competitiveness of its users.

The purpose of using Integrated Knowledge Networks is to initiate, facilitate and govern the innovation processes in an organisation. This is done in such way that the Integrated Knowledge Network and the innovation process together create much more added value than any innovation process can produce in isolation.

The innovation process within an organisation is the result of evolutionary and repetitive cycles of tacit and explicit knowledge creation and knowledge exchanges between different members of the participating innovation teams within a knowledge value chain. Figure 1 indicates the typical (a) tacit and (b) explicit knowledge exchange in an innovation project. People who belong to different communities in both competitive, pre-competitive and user domains (see section 4) are the essential contributors to this knowledge life cycle.

This chapter presents the framework for an Integrated Knowledge Network that offers support to and expedites innovation in an era of exponential knowledge development and abundant connectivity. Understanding the associated knowledge supply chain and configuring it by using an enterprise-wide innovation management system improves the efficiency of the knowledge.

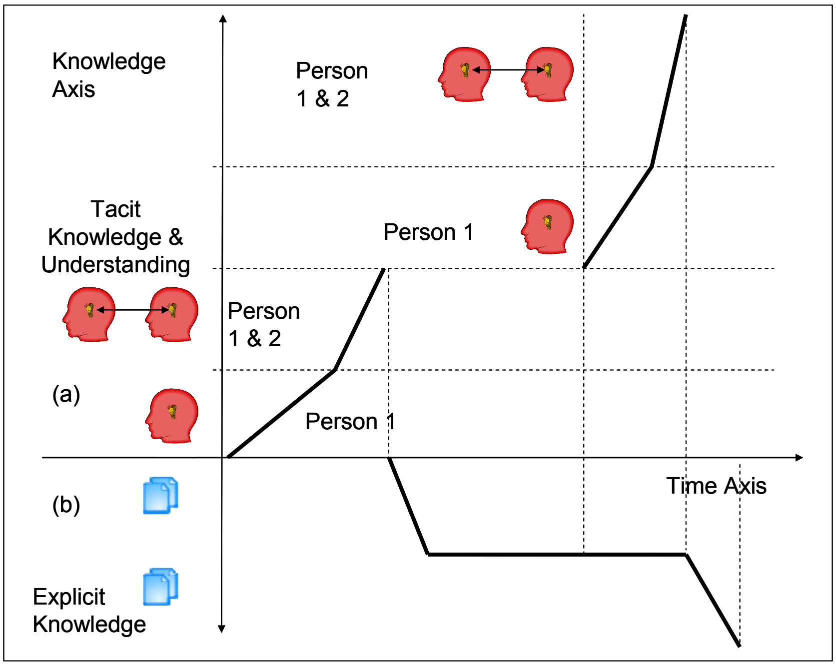


Fig. 1. Interrelated tacit- and explicit knowledge cycles.

2. Innovative Design Knowledge Evolution

During any design project, an interrelated tacit and explicit knowledge development cycle evolves until the final project objectives are reached or the project timeline is reached. This process (depicted in Figure 2) reflects the knowledge creation process as described by Nonaka and Takeuchi [3].

It is important that the associated tacit and explicit knowledge networks that support an innovation project are efficiently managed. This is eloquently described in Seufert et al. [4] who give specific emphasis to the integration of the epistemological and ontological dimensions of knowledge work. Linking structured knowledge to design is common engineering knowledge of which there are many referenced examples (see [5]). However, innovation also requires the right (heterogenic) combination of knowledge, know-how and tools [6], and such combinations in Knowledge Frameworks is not a new approach ([7] and [8] provide examples).

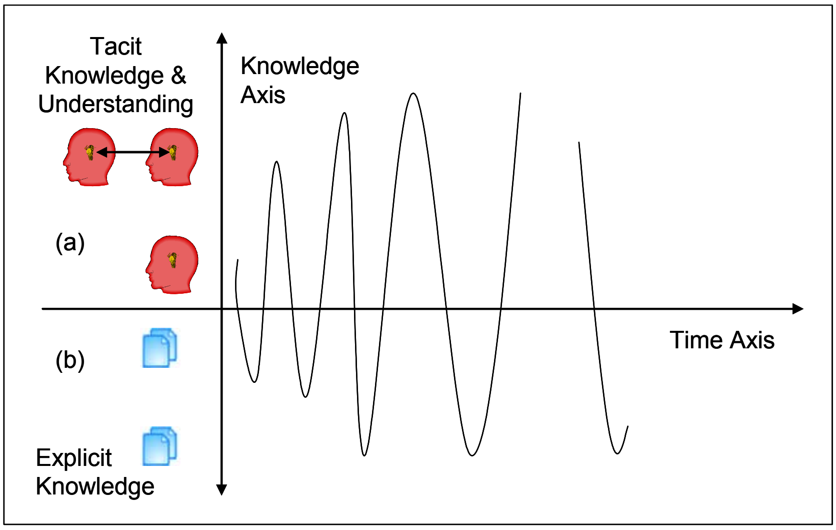


Fig. 2. Interrelated knowledge life cycle.

> Note that in Figure 2 the vertical axis is an indication of the tacit and explicit knowledge and is not a conventional + and – axis. Explicit knowledge actually increases over time.

Configuring integrated knowledge networks, however, requires proactive knowledge management and knowledge processing to facilitate the competitive speed required in innovation ([9] and [10]). A well-defined network consisting of different communities who participate in many different innovation projects is important. A project aimed at improving product, service, process or technology is thus seen as the common smallest unit of innovation.

3. The project as the common denominator

Innovations are executed in projects with associated specific project goals and common team members, drawn from different formal and informal communities, each with their associated expertise, experience, and specialist equipment. The context of an innovation project is thus delimited by the associated project parameters governing the interactive tacit and explicit knowledge exchange. Each innovation project life cycle furthers the knowledge progression. Within a knowledge network, then, the innovation project life cycle is thus considered the smallest common denominator for managing the associated project knowledge. In addition, it must be recognised that such projects are normally also subsets of larger design life cycles, such as enterprise-, product- or technology life cycles.

3.1 Common coordinates for multiple projects

Different innovation projects are initiated at different times, and this timing has an impact on different aspects of product, process and technology development. As this impact is on one or more of three fundamental life cycles, a three axis coordinate system integrates and contextualises different projects in different domains (product/service design, enterprise design and technology development life) and a common coordinate system provides navigation between different innovation projects [11]. See Figure 3.

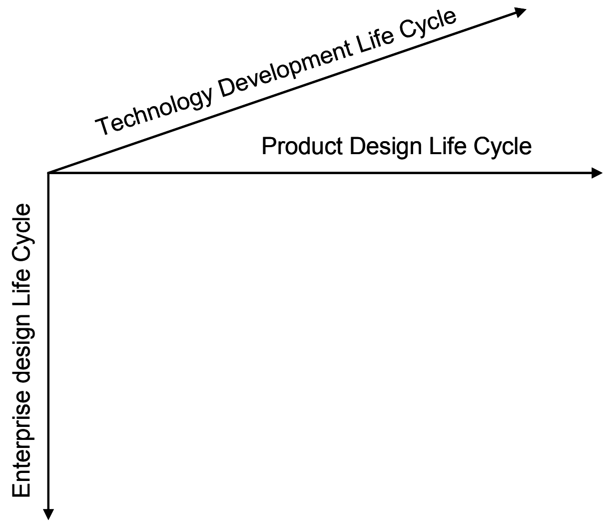


Fig. 3. Coordinate system to navigate inter- and intra-enterprise innovation projects.

The project life cycle forms the primary common denominator as it provides a context for the development of innovation knowledge of a specific project. The full product and enterprise design life cycles provide inter- and intra-enterprise contextualisation coordinates as they link the innovation knowledge of different design projects. The fourth dimension of time allows for the integration and sequencing of knowledge creation components. This “journey” makes it possible to exploit the associated knowledge from different past projects in order to expedite and improve the quality of a current innovation project.

4. Components contributing to innovation

To be effective in supporting innovation, Integrated Knowledge Networks must encompass the following interrelated components:

- People organised into different communities that interact with different formal and informal
- Trust relationships and contracts that allow different collaborative arrangements to share in innovation experience.
- Competencies and experience of the people organised in
- Formal organisational structures such as institutes, research units and departments at universities that have access to
- Various resources like laboratories, networks and technologies,
- Making use of the said tacit, latent and explicit knowledge that resides in the different communities.
- Different role players are participating in
- The public domain, private domain and the user's domain to exploit
- Pre-competitive, competitive and user domain knowledge
- In innovation of products, processes, enterprises and technologies.

These components are represented in Figure 4; which dissects the knowledge supply chain into different knowledge domains, different corresponding supply chain outputs and different role players. In many cases the different role-players are also organised into smaller, less formal Knowledge Networks.

Some aspects are clear when analysing this diagram:

- Public and private domain information together constitute an abundance of knowledge. This implies an extensive risk of information overload.
- The innovation process that offers support the material supply chain is much too complex to be addressed by a single team in a single project, it is imperative to divide the work in order to conquer it.
- Thus, a multiple-team approach of proactive knowledge creation, evaluation, filtering and deployment is advised.
- Extensive interaction between public domain activities and private domain development work is an essential.

If such a hierarchy of interrelated teamwork were devised, it would facilitate the rate at which innovation is deployed.

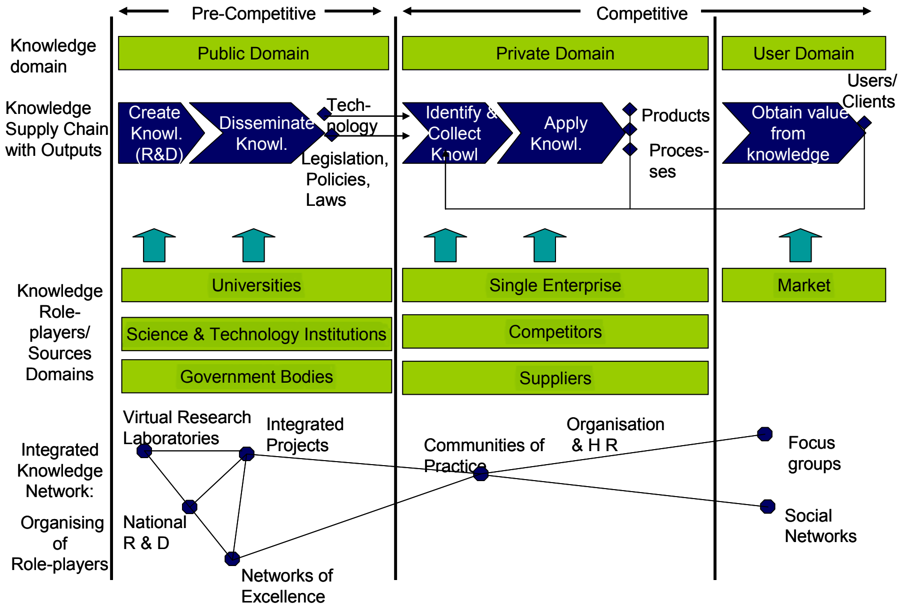


Fig. 4. Components of the knowledge supply chain.

5. Networking different components contributing to innovation

The largest single community networked for sharing explicit knowledge comprises the users of the Internet. This network is, however, not agile or focussed enough to facilitate the rapid innovation required, and also lacks the facility for the exchange of tacit knowledge.

On the other end of the spectrum is a much smaller community, comprising the team members of a specific innovation project. In between these two extremes exists a wide range of different communities who are all focussing on innovation. Two examples of this spectrum may be found in the EU Networks of Excellence and Integrated Projects of the Sixth Framework Programme. However, an Integrated Knowledge Network should provide access to all these different communities. (The VRL-KCiP is an example of such access.)

6. Structuring an integrated knowledge network to support innovation

Ontologies are used to describe and depict relationships between entities within a knowledge network. Such relationships are not static but vary over time, across different projects and as the objectives of particular communities are modified. However, it is necessary to decide on some of the parameters of an integrated network in order to start collaborating and knowledge sharing, and a conceptual framework may be used to model and modify applicable relationships as the network evolves [12].

6.1 Project types: Internal and external

In every organisation, (sub-) projects can be divided in two groups: internal and external. Internal projects entail primary responsibility for execution. The execution of external projects is the responsibility of external partners.

6.2 Project categories

As project life cycles are important drivers of the contextualisation of knowledge, categorising projects is an effective way of distinguishing the various types. In an academic environment this includes all projects (undergraduate, masters, doctoral and those of industrial partners). Existing knowledge repositories

Completed projects, knowledge are then categorised and project documentation indexed. Knowledge Matrices or knowledge maps that represent some graphical categorisation of such knowledge repositories are used as tools to navigate through the IKN. Conceptual frameworks are then constructed that model and maintain the relationships and entities, such as different communities of expertise, domains of knowledge and resource availability. In addition, extensive evaluation of a number of professional society publications in fields such as manufacturing engineering is also executed and such knowledge entries updated regularly.

6.3 Collaborative Enterprise-wide Innovation platform

Integrated Knowledge Networks are operated within an Enterprise-wide Innovation Management Platform. Generic project roadmaps are available to each project team, who can then configure their individual project roadmaps to suit their specific requirements. Configured security and access control, e-mail notification of document activities, as well as progress checklists facilitate the collaborative creation, refinement and reuse of knowledge.

7. Integrated Component view: case study

As a case study, the foundation of the Integrated Knowledge Network of the Global Competitiveness Centre (Stellenbosch University, SA) was assessed. This IKN provides selective context access to more than 100 individual projects, over 30.000 documents and about 50 generic life cycle roadmaps. About 100 internal and 30 external users are registered. Based on the analysis in Figure 4, the following scalable and configurable components were identified as elements of the IKN (this list is indicative rather than complete):

- Project life cycle (Primary Building Block).
  - Common objective(s)
  - Team members
  - Configured roadmap(s)
  - Documents in context of project life cycle
- Within a 3D solution space
  - Based on Bodies Of Knowledge, best practices and generic reference architectures
  - Product design life cycle
  - Enterprise design life cycle
  - Technology development life cycle
  - Innovation program instances
  - Concurrent project roadmaps with context information linked to a common research programme
- Other repositories
  - Publications of professional institutes
    - IEEE
    - South African Institute of Industrial Engineers
    - CIRP
  - Proceedings of specific conferences
    - COMA
    - CIRP Design Conference
- Specialist networks and focus groups (EU and others)
  - Networks of Excellence
    - VRL-KCiP
  - Integrated Projects
  - Design courses/projects

- Public domain
    - Selection of technology roadmaps and foresight studies
        - Regional level
        - Country
        - Industry
        - Supply chain
- Conventional electronic library access
    - Books
    - Dissertation
    - Electronic journals
- Broader Internet access
    - Search engines

In addition, there are relationships between these elements that are affected over time, through project team dynamics, through technology development and competitor activities, to name but a few. For example, public domain technology roadmaps may provide start-up input for a new product development innovation project and the experience of the various industrial partners may then have an impact on the choice of team members.

The different dynamic interrelationships are accommodated by using an innovation management platform. Dividing up and managing smaller portions assists in meeting the knowledge management challenge. Within interrelated projects with common goals, a lot of common information can be shared. This was demonstrated in multiple courses/projects that were aimed at different target groups, ranging from second year students to final year projects, and even graduate research projects.

The projects all had different time frames, varied in complexity and levels of aggregation/detail, and group sizes, with up to 50 different teams involved in one project. Advice from external consultants and domain experts both locally and internationally was made available and lead to substantiated improved designs by four integration teams.

8. Concluding remarks

Turning innovation initiatives into practical solutions is for most companies not only a day-to-day challenge; but is often the real barrier to success. Behind this lies an overwhelming amount of information, knowledge and communication that is involved in innovative design processes. Added to this, the innovation process itself brings an enormous amount of information, knowledge and communication. However, innovation also intercepts with a company's primary processes and their own store of information, knowledge and communication.

Many existing approaches attempt to reduce complexity by prescribing partial solutions in one domain, phase or aggregation level. However, when such borders are crossed into other areas (as is often the case when being innovative) the complexity increases greatly.

Integrated Knowledge Networks is an approach that ameliorates the explosion of complexity, as it provides a broad, encompassing structure that is dynamic, deals with knowledge in real-time, that is content-based, and that can accommodate the knowledge realm of anything from a small project to the largest organisation. Because it is non-prescriptive, an Integrated Knowledge Network offers additional ways to address project content, and thus provides an organisation with tools for addressing that content at the required time, in an appropriate manner and to the desired level of detail.

The case study indicates that every project can indeed be seen as part of a larger whole, and that mapping that larger whole in itself makes a valuable contribution to better understanding and sharing the complete content. By using adequate software tools, this content can be made accessible in a well-ordered, dynamically navigable manner. This not only helps to avoid projects being drowned in complexity, but it also enormously reduces the gap between innovation initiatives and practical solutions.

9. References

1. Commonwealth of Australia, 2001, Technology Planning for Business Competitiveness, A Guide to Developing Technology Roadmaps.
2. NGM Group, 1997, Knowledge Supply Chains: A Next-Generation Manufacturing Imperative, Agility Forum.
3. Nonaka, I., Takeuchi, H., 1995, The Knowledge-Creating Company, Oxford University Press, Oxford.
4. Seufert, A., von Krogh, G, and Bach, A., 1999 Towards knowledge networking, Journal of Knowledge Management, 3:180–190.
5. Meijer, B.R., Tomiyama, T., van der Holst, B.H.A., and van der Werff, K., Knowledge Structuring for Function Design, CIRP Annals, 52(1): Stc Dn.
6. Noël, F., Brissaud, D., Tichkiewitch, S., Integrative Design Environment to Improve Collaboration between Various Experts, CIRP Annals, 52(1): Stc Dn.
7. Thannhuber, M., Tseng, M. M., Bullinger, H., 2002, An Autopoietic Approach for Building Knowledge Management Systems in Manufacturing Enterprises, CIRP Anals, 50(1):Stc O.

8. Heebyung, K., Sungdo, H., Taesoo, K., Hyung-Min R., Soo, H. L., 2003, Design Knowledge Management with Reconstructible Structure, CIRP Annals, 52(1):Stc Dn.
9. Rozenfeld H., 2002, An Architecture for Shared Management of Explicit Knowledge Applied to Product Development Processes, CIRP Anals, 51(1):Stc O.
10. Maropoulos, P.G., Rogers, B.C., Chapman, P., McKay, K.R. and Bramall, D.G., 2003, A Novel Digital Enterprise Technology Framework for the Distributed Development and Validation of Complex Products, CIRP Annals, 52(1):Stc O.
11. Lutters, D., Uys, W., DuPreez, N.D., 2007, Design(er) support based on Conceptual Frameworks, COMA 07 proceedings, ISBN 0797211640, pp. 371-376.
12. Uys, W., 2007, The Implementation of a Conceptual Framework Based Approach for the Improved Viewing and Utilisation of Organisational Information, COMA 07 proceedings, ISBN 0797211640, pp. 363-370.

Table B.1: Data synthesis from Du Preez et al. (2008)

	"What?", i.e. which generalisable inventory sets are significant to the construction of IKNs? (🟩)	***"How?"***, i.e. which generalisable process flows are significant to the construction of IKNs? (🟥)	***"Where?"***, i.e. which generalisable distribution networks are significant to the construction of IKNs? (🟨)

Executive perspective
(Scope contexts in scope identification lists)
	• **Innovation artefacts**
	  • Products
	  • Services
	  • Processes
	  • Technologies
	  • Projects
	• **Resources**
	  • Equipment
	  • Laboratories
	  • Information systems
	    • Innovation management platform
	    • Internet
	    • Search engines
	• **Knowledge**
	  • Information artefacts
	    • Ontologies
	    • Knowledge matrices
	    • Knowledge maps
	    • Books
	    • Dissertations
	    • Publications
	    • Electronic journals
	  • Tacit knowledge
	  • Explicit knowledge
		• **Innovation**
		  • Design
		• **Knowledge creation and transfer**
		  • Knowledge management
		  • Knowledge processing
		  • Communication
			• **Regional locations**
			• **National locations**

	"Who?", i.e. which generalisable responsibility assignments are significant to the construction of IKNs? (🟦)	***"When?"***, i.e. which generalisable timing cycles are significant to the construction of IKNs? (🟪)	***"Why?"***, i.e. which generalisable motivation intentions are significant to the construction of IKNs? (◼)

Executive perspective
(Scope contexts in scope identification lists)
	• **Network stakeholders**
	  • People
	  • Domains
	  • Communities
	  • Teams
	  • Science and technology institutions
	  • Research units
	  • University departments
	  • Government bodies
	  • Supply chain
	  • Competitors
	  • Industry enterprises
	  • Market
	• **Knowledge contributors**
		• **Knowledge life cycle**
		• **Innovation project life cycle**
		  • Project timeline
		• **Enterprise / network life cycle**
		• **Technology life cycle**
			• **Value creation**
			• **Competitiveness**
			• **Innovation intention**
			  • Sustainable innovation
			  • Innovation facilitation and governance
			  • Project objectives

Composite constructs	• **Knowledge network** • **Integrated knowledge network** • **Knowledge supply (value) chain** • **Competitive speed** • **Innovation project context** • **Product, process and technology development** • **Trust relationships** • **Collaborative arrangements** • **Internal and external projects** • **Innovation program** • **Publications and conferences**