The London School of Economics and Political Science

Architectural Evolution through Softwarisation:

On the Advent of Software-Defined Networks

Reuel Edison Ocho

A thesis submitted to the Department of Management of the London School of Economics for the degree of Doctor of Philosophy, London, December 2016

Declaration

I certify that the thesis I have presented for examination for the PhD degree of the London School of Economics and Political Science is solely my own work other than where I have clearly indicated that it is the work of others (in which case the extent of any work carried out jointly by me and any other person is clearly identified in it).

I warrant that this authorisation does not, to the best of my belief, infringe the rights of any third party.

I declare that my thesis consists of 73,258 words.

Abstract

Digital infrastructures characteristically expand and evolve. Their propensity for growth can be attributed to the self-reinforcing mechanism of positive network externalities, in which the value and attractiveness of any digital infrastructure to users, is generated from and sustained as a function of the size of its existing user community. The expansion of any digital infrastructure, though, is ultimately underpinned by an inherent architectural capacity to support unanticipated change, that may include changes to architecture itself. However, as digital infrastructures scale, their usage grows, and they encounter and become entangled with other digital infrastructures. As such, the capacity of digital infrastructure architecture to accommodate change, under conditions of positive network externalities that attract users, conversely leads to intensified social and technical dependencies that eventually resist certain kinds of change. That is, it leads to *sociotechnical ossifications*. Changing underlying architecture in existing digital infrastructures, thus, becomes increasingly prohibitive over time.

Information Systems (IS) research suggests that architectural change or *evolution* in digital infrastructures occurs primarily via a process of replacement through two means. An existing digital infrastructure is either completely replaced with one that has an evolved architecture, or intermediary transitory gateways are used to facilitate interoperability between digital infrastructures of incompatible architectures. Recognising the sociotechnical ossifications that resist architectural evolution, this literature has also tended to focus more on social activities of cultivating change of which the outcome is architectural evolution in digital infrastructures, than directly on architectural evolution itself. In doing so it has provided only a partial account of underlying architectural evolution in digital infrastructures.

The findings of this research come from an embedded case study in which changes to underlying architecture in existing networking infrastructures were made. Networking infrastructures are a prime instance of sociotechnically ossified digital infrastructures. The case's primary data sources included interviews with 39 senior networking and infrastructure virtualisation experts from large Internet and Cloud Service Providers,

Standards Development Organisations, Network Equipment Vendors, Network Systems Integrators, Virtualisation Software Technology Organisations, Research Institutes, and as well technical documents. A critical realist analysis was used to uncover generative mechanisms that promote underlying architectural evolution in sociotechnically ossified digital infrastructures.

This thesis extends IS understanding of architectural evolution in digital infrastructures with the complementary finding of, *architectural evolution through softwarisation*. In architectural evolution through softwarisation, the architecture of sociotechnically ossified digital infrastructures, is evolved via the exploitation of features inherent to digital entities, which have been overlooked in extant research on architecture in digital infrastructures.

Acknowledgements

I would like to thank my first supervisor, Carsten Sørensen, for his advice, guidance, support and encouragement throughout my time at the Information Systems and Innovation Group as a PhD student. I genuinely appreciate his dedication to my success, and his skilfulness in manoeuvring me through the maze that is a PhD!

To my first educators, my parents, Edris Ocho and Albertina Ocho, thank you for always emphasizing the importance of education to my sisters and I, from an early age. To my sisters, Candace Ocho-Stewart and Arlene Ocho, thank you for your support and encouragement during my journey.

I would also like to thank my uncle Simeon Franklin Alexander, whose technical ingenuity inspired me at a young age, and I must also express my deepest gratitude to my uncle Joseph Kenrick Alexander, who asked me while I was yet a young child, "Why don't you study computers?" That was the beginning of this journey! To my uncle, Dr. Oscar Ocho, thank you so much for the long talks, advice, and reassurances along the way!

I need to thank Jannis Kallinikos, my second supervisor, for providing valuable knowledge and guidance particularly at the beginning of my time as a research student. Finally, I would like to thank Chrisanthi Avgerou, who willingly took time to provide clear, helpful advice and feedback on my work, even though she was not officially one of my supervisors.

Table of Contents

Acronyms

Acronym	Meaning
5G	5th Generation Mobile Networks
ACI	Application Centric Infrastructure
API	Application Programming Interfaces
ARP	Address Resolution Protocol
ASIC	Application-Specific Integrated Circuit
BGP4	Border Gateway Protocol version 4
Capex	Capital Expenditure
CAQDAS	Computer-Assisted Qualitative Data Analysis
CLI	Command Line Interface
COTS	Commercial Off The Shelf
DOS	Denial of Service
ETSI	European Telecommunications Standards Institute
FIRE	Future Internet Research and Experimentation
ForCES	Forwarding and Control Element Separation
FTP	File Transfer Protocol
GENI	Global Environment for Network Innovations
GGSN	Gateway General Packet Radio Service Support Node
HTTP	Hyper Text Transfer Protocol
IEEE	Institute of Electrical and Electronics Engineers
IETF	Internet Engineering Task Force
IP	Internet Protocol
IPS	Intrusion Prevention System
IPv4	Internet Protocol version 4
IPv6	Internet Protocol version 6
IS	Information Systems
IS-IS	Intermediate System to Intermediate System
ISP	Internet Service Provider
IT	Information Technology
ITU-T	Telecommunication Standardization Sector of the International Telecommunication Union
MEF	Metro Ethernet Forum
MVNO	Mobile Virtual Network Operator
NFV	Network Functions Virtualisation
NVRAM	Non-Volatile Random Access Memory
ONF	Open Networking Foundation
Opex	Operational Expenditure
OSPF	Open Shortest Path First
PDF	Portable Document Format

QoS	Quality of Service
RAM	Random Access Memory
RFC	Request for Comments
RIP	Routing Information Protocol
ROM	Read-Only Memory
SAVI	Strategic Network for Smart Applications on Virtual Infrastructure
SDN	Software Defined Networking
SDO	Standards Development Organisation
SS7	Signalling System No. 7
TCAM	Ternary Content-Addressable Memory
TCP	Transmission Control Protocol
TMSA	Transformational Model of Social Action
VLAN	Virtual Local Area Network
VM	Virtual Machine
VPN	Virtual Private Network
WAN	Wide Area Network

Table of Figures

List of Tables

1 Introduction: Architectural Evolution in Sociotechnically Ossified Digital Infrastructures

> *"So, how would you describe the problem that SDN (Software-Defined Networking) is a solution for?"* – ***Interviewer (Reuel E. Ocho)***
>
> *"Essentially, it's that, so far, network infrastructures have been very, the term you use – hear very often, is 'ossified'. They have been very rigid…"* – ***Interviewee I33, Head of a technical committee at a Standards Development Organisation****, (Meaning of SDN added.)*

1.1 Introduction

Digital infrastructures are perpetual (Star & Ruhleder, 1996; Hanseth, et al., 1996; Edwards, et al., 2007; Ribes & Finholt, 2009), and yet they characteristically expand and evolve (Edwards, et al., 2007; Hanseth & Lyytinen, 2010; Tilson, et al., 2010b). Change in digital infrastructures is partly attributed to their openness to extension by diverse innovators, and to their unboundedness – the absence of a finite boundary that restricts their growth (Hanseth & Lyytinen, 2010). Aside from technical innovators, users that create or consume content that populate or traverse these infrastructures contribute to continued expansion and evolution (Benkler, 2000; Zittrain, 2008; Tilson, et al., 2010b).

The described propensity for sociotechnical expansion and evolution is partly underpinned by a self-reinforcing mechanism, *positive network externalities* (Katz & Shapiro, 1985; Katz & Shapiro, 1986; Katz & Shapiro, 1994; Shapiro & Varian, 1999, pp. 173-225), in which the value of an infrastructure is generated from and sustained as a function of the size of its active user community, i.e., innovators, content creators, users with whom they share strong ties, and with other users (Suarez, 2005; Zhu, et al., 2006; Edwards, et al., 2007; Hanseth & Lyytinen, 2010; Monteiro, et al., 2013). Under the conditioning of positive network externalities, users find greater benefit in joining the collective of an already established digital infrastructure with a larger user community, than in joining one that has a smaller user community, because each individual user, directly or indirectly, contributes value enjoyed by others in the community (Katz & Shapiro, 1985; Katz & Shapiro, 1994; Suarez, 2005). Similarly, existing users face lower incentive to leave a digital infrastructure due to switching and adoption costs incurred (Katz & Shapiro, 1994; Zhu, et al., 2006; Shapiro & Varian, 1999, pp. 184-186).

Although an important explanatory mechanism, positive network externalities alone do not reveal reasons for digital infrastructures' characteristics of expansion and evolution. Complementary to the role of positive network externalities, are the technical qualities and features that underlie digital infrastructures – specifically *architecture*. The underlying architecture of a digital infrastructure is a stable base whose principles of design and features, inherently give the architecture a flexibility capacity for supporting and cultivating unanticipated change (Tilson, et al., 2010b; Yoo, et al., 2010). This flexibility is a *generative capacity* (Zittrain, 2008), as it lends itself to continuous innovative exploitation (Tilson, et al., 2010b; Yoo, et al., 2010). That is, the architecture tends towards unbounded generativity, rather than towards the imposition of boundaries (Zittrain, 2008).

Concomitant with characteristics of expansion and evolution, however, is the phenomenon of increasing sociotechnical *irreversibility* (Hanseth, et al., 1996; Monteiro, 1998; Edwards, et al., 2007; Hanseth & Lyytinen, 2010). As digital infrastructures expand, positive network externalities reinforce users' dependence on them, but as well, they become entangled with other digital infrastructures (Monteiro, 1998; Hanseth, 2001; Egyedi & Spirco, 2011; Monteiro, et al., 2013). These social and technical dependencies intensify and *ossify* over time, and eventually resist certain kinds of change – in particular, evolution of the underlying architecture in digital infrastructures.

In Information Systems (IS) research on digital infrastructures, architectural evolution is often discussed in the context of the formation, growth and evolution of deployed digital infrastructures. With the exception of a few studies that directly confront architectural evolution, such as Monteiro (1998), the majority of studies have emphasized the addition of compatible, complementary, or interoperable components to the edges of existing digital infrastructures, post formation, and to a lesser degree, the super-imposition of digital infrastructures. A selection of examples includes (Hanseth, 2001; Sahay, et al., 2009; Egyedi & Spirco, 2011; Sanner, et al., 2014; Grisot, et al., 2014; Rodon & Silva, 2015).

Commentaries on architecture note that although digital infrastructures are not designed *de novo* by a single designer, but rather extend or build on top of existing infrastructures (Star & Ruhleder, 1996; Hanseth & Lyytinen, 2010; Aanestad & Jensen, 2011), initial design decisions (Edwards, et al., 2007; Eriksson & Ågerfalk, 2010; Hanseth & Lyytinen, 2010; Egyedi & Spirco, 2011; Rodon & Silva, 2015), and architectural design principles play a role in the kind of digital infrastructure changes that are possible (Zittrain, 2008; van Schewick, 2010, pp. 23, 37-81; Grisot, et al., 2014; Rodon & Silva, 2015). Most prominent, are arguments that highlight the role of modularity-derived architectural design principles in digital infrastructures. These design principles are explained as being critical enabling architectural features for generativity (Zittrain, 2008) in digital infrastructures (Hanseth, et al., 1996; Hanseth & Lyytinen, 2010; Aanestad & Jensen, 2011; Grisot, et al., 2014; Rodon & Silva, 2015). This perspective on architecture articulates a *connection-oriented conceptualisation* of architecture, in which the means by which information technology (IT) components are interconnected, takes importance. Modularity's standardised interfaces, and low internal inter-relatedness between modules, is credited for its accommodation of extensions to digital infrastructures at the edges by connecting compatible IT components (Langlois & Robertson, 1992; Hanseth, et al., 1996; Schilling, 2000; Yoo, et al., 2010).

In this connection-oriented architectural perspective, when modules or digital infrastructures with incompatible interfaces meet, *gateways* become indispensable adaptors for interconnection (Hanseth, 2001), and therefore continued digital infrastructure expansion. The connection-oriented architectural perspective has also been applied to change of underlying architecture. Gateways facilitate interoperability between digital infrastructures of incompatible underlying architectures, and may serve as a transitory technical strategy until one infrastructure is transformed architecturally to be like another with which it interconnects, or as a long-term solution for backward compatibility or interoperability between digital infrastructures of permanently disjoint architectures (Hanseth, 2001; Hanseth & Lundberg, 2001; Edwards, et al., 2007; Egyedi & Spirco, 2011).

With IS research on architectural evolution in digital infrastructures largely premised on a conceptualisation of architecture that emphasizes the interrelatedness of modules, *architectural evolution by interconnection* is most eminent. Underlying architecture either does not evolve, or it evolves with difficulty via complete *replacement* of its instantiating digital infrastructures or via incremental *replacement* facilitated by the attachment of transitory gateways, and accompanied by an exercise of cultivating the sociotechnical installed base of the digital infrastructures (Monteiro, 1998; Hanseth, 2001; Hanseth & Lundberg, 2001; Edwards, et al., 2007; Egyedi & Spirco, 2011; Sanner, et al., 2014; Grisot, et al., 2014).

1.2 Research Problem Statement

But architectural evolution by interconnection might be overstating the role of modularity in digital infrastructure architectural evolution. Recent research suggests that *materiality* is significant in issues of architecture (Henfridsson, et al., 2014). *Digital materiality* admits changes to the data structure and functionality of digital entities, post their production and distribution (Manovich, 2002, pp. 27-48; Kallinikos, et al., 2010; Yoo, et al., 2010; Kallinikos, et al., 2013; Henfridsson, et al., 2014). This raises a question of whether and to what extent digital materiality may admit architectural evolution, comparably to the way that it admits changes in the functionality of digital entities. That is, can underlying architecture be evolved via the exploitation of the digital materiality of digital infrastructures and if so, how does it occur? Helpful insights on this question may be borrowed from the neighbouring discipline of computer science, particularly in relation to networking infrastructures.

The connection-oriented perspective on architecture and the use of gateways is well-acknowledged in computer science, but as explained by Monteiro (1998), the use of gateways as a means of underlying architectural evolution in networking infrastructures has been limited to problems that are narrow in scope. In some contemporary developments in computer science the *upward flexibility* (Tilson, et al., 2010b), which is the ability to superimpose unanticipated innovation on digital infrastructures, as facilitated by the generative capacity of underlying architecture in networking infrastructures, was exploited (Anderson, et al., 2005; Chowdhury & Boutaba, 2010) to evolve underlying architecture of networking infrastructures (McKeown, et al., 2008; Casado, et al., 2009). A number of owners of large networking infrastructures have since appropriated this innovation and have used it to evolve the underlying architecture of their production networking infrastructures. The innovation was selected specifically because the approach to architectural evolution simultaneously circumvented intense sociotechnical ossification of the architecture of networking infrastructures (Anderson, et al., 2005; Casado, et al., 2009; Chowdhury & Boutaba, 2010; Gartner Inc., 2015a; International Data Corporation, 2015).

Although the generative capacity of the architecture in digital infrastructures and its relationship with digital infrastructure expansion has been well-studied in IS research on digital infrastructures, the mechanisms of underlying architectural evolution in sociotechnically ossified digital infrastructures have not been as widely elaborated, and explanations typically focus on the social cultivation of an installed base (Monteiro, 1998; Hanseth & Lyytinen, 2010; Aanestad & Jensen, 2011; Sanner, et al., 2014; Grisot, et al., 2014). The computer science literature, on the other hand, seems to be engaging architectural evolution in digital infrastructures, but in the absence of any explanatory theorisation. Thus, architectural evolution in sociotechnically ossified digital infrastructures post the advent of these new networking infrastructures, remains imperfectly theoretically articulated.

1.3 Research Objectives and Research Strategy

The objective of this research, then, was to develop an explanatory theory (Gregor, 2006) of how architectural evolution in sociotechnically ossified digital infrastructures occurs. This thesis aims to extend the extant IS theorisations of architectural evolution by interconnection in digital infrastructures, with a complementary finding of *architectural evolution through softwarisation*. The literature to which the findings contributes is the broad IS literature on digital infrastructures and not only towards understanding of architectural evolution in digital infrastructures. The reason is that on the journey through the explication of architectural evolution through softwarisation, characteristics of digital infrastructures not previously elucidated in IS research were uncovered as important corollary theoretical insights. These characteristics were found to be instrumental in continued architectural evolution in digital infrastructures.

The research question investigated was the following:

> **Research Question:** Which mechanisms promote architectural evolution in sociotechnically ossified digital infrastructures?

A critical realist embedded case study (Easton, 2010; Wynn & Williams, 2012), was implemented as the research design strategy for finding an answer to the research question (Cohen, et al., 2011, pp. 291-292; Yin, 2014, pp. 49-56). The advent of Software-Defined Networking infrastructures, the type of networking infrastructure that the aforementioned owners of large networking infrastructures have been implementing, was selected as a revelatory case (Mabry, 2008; May & Perry, 2011, pp. 228-233; Yin, 2014, p. 52). The case is revelatory because networking infrastructures are a prime example of sociotechnically ossified digital infrastructures, yet the advent of Software-Defined Networking infrastructures indicates architectural evolution that circumvented these ossifications but that did not occur via either of the aforementioned types of infrastructure replacement processes that have been theorised in IS.

Thirty-nine elite interviews with senior experts in networking and infrastructure virtualisation from large Internet and Cloud Service Providers including from Tier 1 providers, Standards Development Organisations, Network Equipment Vendors, Network Systems Integrators, Virtualisation Software Technology Organisations, and researchers that held experience working at owners of large networks were conducted. Additionally, a comprehensive analysis of documentation and archival data to understand the architectural details of Software-Defined Networking infrastructures and reasons for their advent, was conducted. These provided complementary findings out of which the theoretical contributions, explained in terms of the operation of social, technical, and sociotechnical generative mechanisms, were developed.

The word "promote" was chosen for the research question instead of "cause" in deference to critical realism's formulation of causality which may include transfactually acting generative mechanisms. Critical realist causality and explanation is explained in Chapter 3.

1.4 Thesis Organisation

This thesis continues with an exposition of the theoretical construct of the research question, namely *architectural evolution in sociotechnically ossified digital infrastructures*. Each concept in the theoretical construct is explained in turn via a review of related literature in Chapter 2. The following chapter presents critical realism as the philosophical base of the research. Chapter 4 introduces the embedded case study research design strategy, and explains the implications of critical realism on the research design. It gradually elucidates the data collection and analysis approaches, articulating the rationale behind various decisions made. The chapter ends by identifying the generalisation argument for the theoretical contributions, and provides a detailing of how ethical considerations were implemented throughout the research.

Chapters 5 and 6 present the findings of the research. Chapter 5 introduces the advent of Software-Defined Networking infrastructures through a critical realist theoretic lens, specifically the morphogenesis of structure (Archer, 1982; Archer, 1995), covering antecedent conditions of structuring, intermediary catalysing conditions, and Software-Defined Networking infrastructures as the outcome. Chapter 6 then proceeds to identify and argue for the reality of particular generative mechanisms whose operation provide a complementary causal explanation of how architectural evolution in sociotechnically ossified digital infrastructures occurs. A discussion of the findings is presented in Chapter 7, after which Chapter 8 summarises the theoretical contributions.

2 Related Literature

2.1 Introduction

This chapter presents related literature that explain the main theoretical construct of the research question: *architectural evolution in sociotechnically ossified digital infrastructures*. The first section of the chapter clarifies what architecture is, and what is meant by its evolution. Three types of architectural evolution related to digital infrastructures and to the case study are presented. Following this exposition of architectural evolution, sociotechnical ossification is developed out of the technical and social implications of one of the types of architectural evolution. Next, technical details of how networking infrastructures work, which are pertinent to the case study and to the findings of this research are presented. The final section of the chapter articulates an abstract formulation of what the term “digital” which precedes “infrastructure” connotes. This explanation of the digital, is particularly important for understanding how architectural evolution in sociotechnically ossified digital infrastructures occurs.

2.2 Architectural Evolution

2.2.1 What is Architectural Evolution?

Two general definitional perspectives of architecture are the prescriptive and the descriptive perspectives (Taylor, et al., 2010, pp. 59-60,65-68). In the prescriptive perspective, architecture is the abstract structure of *required* IT components in digital infrastructures, their *required* functionality, how they *should* interact, and the relationships that *should* exist between them (Maier & Rechtin, 2002, p. 285; Taylor, et al., 2010, pp. 58-60,65-68; Bass, et al., 2013, pp. 4-6,10-21; Crawley, et al., 2016, pp.

16-23), and it is implicated in the kind and amount of innovation in[1] digital infrastructures that can occur, and who is allowed to innovate (van Schewick, 2010, pp. 21-81; Crawley, et al., 2016, pp. 16-23). Design principles outline high-level architectural objectives and rules, such as where functionality should be located or how the architecture should be decomposed into smaller IT components (van Schewick, 2010, pp. 23, 37-81; Taylor, et al., 2010, pp. 58-75; Bass, et al., 2013, pp. 10-19). These principles explain the rationale for the original architectural design, and provide guidelines for future architectural evolution in a digital infrastructure while safeguarding important long-term objectives of the digital infrastructure (van Schewick, 2010, pp. 23, 37-81; Taylor, et al., 2010, pp. 58-59,65-68; Bass, et al., 2013, pp. 10-19; Crawley, et al., 2016, p. 211).

The second definitional perspective of architecture is the descriptive perspective (Taylor, et al., 2010, pp. 59-60,65-68). From a descriptive perspective, architecture is the abstract structure of *deployed* IT components in digital infrastructures, their functionality, how they interact, and the relationships that exist between them *as derived from* an *operating* digital infrastructure (or other digital entity) *at a point in time* (Taylor, et al., 2010, pp. 59-60,65-68).

It is important to understand that these are perspectives. To better understand that these are perspectives consider the following. A prescriptive *deployment architecture* is the abstract structure that connotes how a digital infrastructure is *to be deployed,* while a descriptive *deployment architecture* describes the literal finite abstract structures (Taylor, et al., 2010, pp. 59-60,65-68; Bass, et al., 2013, p. 14) of an *operating* digital infrastructure.

Architectural evolution may be understood as changes in the architecture of an existing digital infrastructure, that are the outcome of innovations that modify or replace IT

[1] Here, I use the phrase "innovation in" in isolation from the amplifications of Grisot, et al. (2014).

components, add [new types of] IT components, or change the responsibilities of, or basis of interactions or relationships between IT components (Grisot, et al., 2014). Innovation which adds, modifies, or duplicates IT components, or inter-connects or super-imposes digital infrastructures (Henderson & Clark, 1990; Hanseth, 2001; Hanseth & Lyytinen, 2010; Grisot, et al., 2014), fall within the scope of the evolution of deployment architecture (from the descriptive perspective) as derived from an operating digital infrastructure at a point in time. For terminological brevity, I shall refer to deployment architecture as from the descriptive perspective as "deployment architecture"[2] in the remainder of this thesis.

As this research investigated architectural evolution via a case study of the advent of a particular type of networking infrastructure, the remainder of this section will explain architectural evolution in terms of the architecture found in networking infrastructures, namely the Internet layered architecture.

2.2.2 The Internet Layered Architecture

Three design principles have guided the definition of the Internet layered architecture: *modularity* which facilitates the independent modifiability of modules by co-locating related functionality within the same IT component (abstractly called a *module*) and by maintaining weak relationships (i.e., *loose coupling*) between modules such that they interact with each other only via standardised interfaces; *layering*, a specialisation of modularity (van Schewick, 2010, p. 46), which logically co-locates related modules, and imposes rules that restrict interactions between the modules of different layers (Sanchez & Mahoney, 1996; Schilling, 2000; Yoo, et al., 2010; van Schewick, 2010, pp. 38-47), and *the end-to-end principle* which advocates minimising the amount of problem-specific functionality that is natively embedded within networking infrastructures, and deferring the responsibility of correctly implementing that functionality to digital hardware and software entities that use the infrastructures and

[2] The prescriptive perspective of deployment architecture is not relevant in this thesis.

that have a more complete knowledge of the required functionality (Saltzer, et al., 1984).

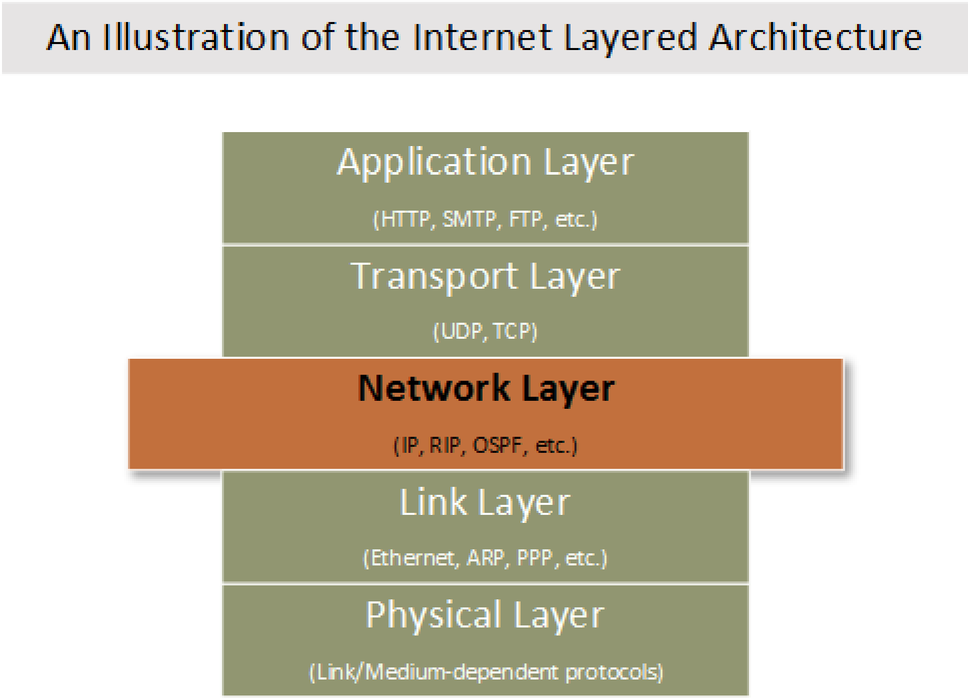

Figure 2-1 The Internet Layered Architecture

The outcome of the application of these design principles has been a *prescriptive* five-layer architecture of the Internet, visually illustrated in Figure 2-1, centred around its *network layer* whose *Internet Protocol (IP)* (Postel, 1981; Deering & Hinden, 1998) standardises generic rules for packaging, interpreting, and transporting digital data between digital hardware and software entities that are connected to the Internet (Kurose & Ross, 2013, pp. 75-79). By observing the end-to-end principle in its implementation of the Internet Protocol, the network layer acts as a *portability* layer to other layers (van Schewick, 2010, pp. 47, 89-90), deferring auxiliary problem-specific functionality to upper layers and their standardised *protocols* – rules for exchanging and interpreting messages between digital hardware and software entities at a layer (Kurose & Ross, 2013, p. 35); for instance, like guaranteeing data delivery (Saltzer, et al., 1984) to the Transport Layer via the Transmission Control Protocol (TCP) (Postel, 1981), or defining protocols for super-imposed applications or infrastructures like File Transfer Protocol (FTP) and the Web's Hyper Text Transfer Protocol (HTTP) to the Application layer (Postel & Reynolds, 1985; Fielding, et al., 1999).

Portability, allows upper layers to change (such as via the addition of new protocols), without compromising the generic interconnectivity and data transportation functionality of the Internet. The Link and Physical layers can also transparently change or be substituted without necessitating significant work that updates existing digital hardware and software entities at layers above the network layer to accommodate new means of transporting data (such as wireless, fibre optics, etc.) (van Schewick, 2010, pp. 48, 89-90).

The five-layer Internet architecture is a type of prescribed architecture that is invariant and generally present at some level of abstraction (Maier & Rechtin, 2002, pp. 283-287; Taylor, et al., 2010, pp. 59-81) in networking infrastructures (Kurose & Ross, 2013, pp. 75-79). For terminological brevity, I shall henceforth refer to this type of prescriptive architecture as *core architecture*[3] in the remainder of this thesis.

The Internet's modular, layered, core architecture guided by the end-to-end principle bears minimal assumptions about future usage scenarios, and its network layer tends towards portability, relative to upper layers, i.e., it embodies a capacity for *upward flexibility*, and portability relative to lower layers, i.e., it embodies a capacity for *downward flexibility* (Tilson, et al., 2010b). This capacity for upward and downward flexibility is furthered by the homogenisation of the base signification of the data exchanged between and processed within the layers, into a binary script that is separated from and agnostic to the software, hardware or other medium that ultimately processes or generates it (Manovich, 2002, pp. 27-48; Yoo, et al., 2010; Faulkner & Runde, 2013). This base signification is explained in the last section of this chapter. The flexibility capacity of the Internet's core architecture, together with generally public open access to participation in the Internet, has been exploited not only for innovating extensions to the Internet's core architecture through the addition of protocols, but more predominantly by innovation through recombination and use

---

[3] Indeed, from a descriptive perspective, core architecture may be derived from a deployed networking infrastructure.

(Bygstad, 2010; Henfridsson & Bygstad, 2013) of existing standardised protocols and services, to create new innovations in hardware and software.

2.2.3 Propagation of the Internet's Core Architecture Instantiations

The definition of the Internet's invariant core architecture with its standard protocols (Internet Engineering Task Force, 2016), as well, has been continuously and extensively exploited to proliferate new network deployments. The Internet, as a globally distributed network of networks (the term "network" is interchangeable with "networking infrastructure"), is implemented in a distributed manner (van Schewick, 2010, pp. 50-51; Kurose & Ross, 2013, pp. 75-76). The implementation of the Internet's five layers' protocols is distributed across network devices that implement interconnectivity between Internet Service Provider's (ISP) and Content Provider networks which collectively form the Internet's *global networking infrastructure* (Tanenbaum, 2003, pp. 14-30; Kurose & Ross, 2013, pp. 58-61).

Under the conditioning of positive network externalities which make an IP-based network preferable, many smaller networks, in enterprises, universities and homes, implement the Internet's core architecture protocols across network hardware and software components to facilitate communication and data transfer between the digital hardware and software entities that are connected to those networks (van Schewick, 2010, pp. 50-51; Kurose & Ross, 2013, pp. 38-61). These networks connect to the Internet's global networking infrastructure via communication links to ISP networks which, in turn are interconnected with other ISP networks that collectively form the Internet's global networking infrastructure (Tanenbaum, 2003, pp. 14-30; Kurose & Ross, 2013, pp. 38-61). Depending on their role and public visibility, networks that connect to the Internet's global networking infrastructure via its standardised protocols, may expand the overall Internet. Thus, in abstraction, the proliferation of new networks yields an evolution of the deployment architecture of the Internet, but this evolution occurs in conformance with an invariant core architecture.

2.2.4 The Generative Capacity of Core Architecture

In summary, the Internet's core architecture can be characterised as a stable base whose design principles, and architectural features, inherently give it a flexibility capacity for supporting and occasioning unanticipated change. This capacity, is a *generative*[4] one (Zittrain, 2008), as it lends itself to exploitation, and has been exploited to innovate new protocols, services, hardware and superimposed infrastructures in the application, transport, link and physical layers. It has been exploited to create a global continuously expanding proliferation of interconnected networks, within which the five-layer core architecture is replicated, that extends the Internet's deployment architecture. It has also been exploited to create conditions that have led to an explosion of user-generated content. That is, the core architecture of the Internet tends towards unbounded generativity, rather than towards the imposition of boundaries (Lessig, 2001, pp. 120-141; Zittrain, 2008).

Collectively, the Internet's expansion and inherent capacity for unboundedness, as a networking infrastructure, can be traced to the exploited generative capacity of its core architecture, and is underpinned by positive network externalities (Katz & Shapiro, 1985; Katz & Shapiro, 1986; Katz & Shapiro, 1994; Shapiro & Varian, 1999, pp. 173-225; Edwards, et al., 2007; Hanseth & Lyytinen, 2010).

But this is architectural evolution of the deployment architecture.

This description, which highlights unboundedness, flexibility and generativity, has another side: the unbounded growth of the Internet is accompanied by the phenomenon of sociotechnical *irreversibility* (Hanseth, et al., 1996; Monteiro, 1998; Edwards, et al., 2007; Hanseth & Lyytinen, 2010) which resists architectural evolution of core architecture.

---

[4] Generative capacity as used here, borrows from (Zittrain, 2008) and is distinguished from the formulation proposed by (Avital & Te'eni, 2009), who define it as an attribute of people or artificially intelligent entities relative to a task performance context. Its use here is not synonymous with their definition of generative fit either, which likewise has a narrow scope of task performance. The generative capacity of the Internet's core architecture may include, but is broader than *fit* for task performance.

2.3 Sociotechnical Ossification of Digital Infrastructures

The change dynamics of digital infrastructures is somewhat paradoxical. On the one hand, the long-term endurance of digital infrastructures is a required quality, but that quality is confronted by, as strong, the necessity to accommodate change (Ribes & Finholt, 2009; Hanseth & Lyytinen, 2010; Tilson, et al., 2010b; Monteiro, et al., 2013) – change that may include core architectural evolution. Changing the core architecture of a widely proliferated and extensively utilised digital infrastructure is difficult, and in IS research, its accommodation is not usually considered to be a required quality of digital infrastructures.

To elaborate, upward flexibility of the Internet's network layer necessitated the creation of standardised stable interfaces, upon which billions of dependencies have been created as IP-based networks have replicated the Internet's core architecture, and as digital hardware and software entities connect to the global networking infrastructure (van Schewick, 2010, pp. 89, 98, 151; Gartner Inc., 2015a; International Data Corporation, 2015). As exemplified by the challenges which underlie the slow transition from IP version 4 (IPv4) to IP version 6 (IPv6) (Monteiro, 1998; Wu, et al., 2013), these dependencies constrain core architectural evolution, *ossifying* as it were, the pre-existing architecture, and by extension, dependencies (Tilson, et al., 2010b).

As networks grow, they encounter and become entangled with other networks, and with digital infrastructures that rely on them for communication, ossifying the mechanisms (i.e., the use of Internet standard protocols) chosen to technically integrate them (Monteiro, 1998; Hanseth, 2001; Egyedi & Spirco, 2011; Monteiro, et al., 2013). Further, innovations that would fall outside of the generative capacity of the architecture are resisted (Zittrain, 2008; van Schewick, 2010, p. 152), paradoxically by an architecturally imposed ossification of the *scope* of generative possibilities. To illustrate, it is not possible to innovate in a manner that would exploit new features of IPv6, within a pure IPv4 network (Wu, et al., 2013). That is, the very capacity of architecture to accommodate change, under conditions of positive network externalities that prompt massive enrolment in a digital infrastructure, leads to a resistance to certain kinds of change (Hanseth, et al., 1996; Tilson, et al., 2010b). It is a condition of technical irreversibility. There is social irreversibility too.

Unlike simple software systems that can be upgraded at once, or one installation at a time, the composite of IP-based network functionality is dispersed across five layers of protocols that are implemented by physically distributed network hardware devices and software, such that partially changing the core architecture could create incompatibilities and ultimately a partially functioning networking infrastructure with areas across which data cannot traverse (van Schewick, 2010, p. 138; Wu, et al., 2013). The core architecture imposes restrictions on which parts of it must change together, and the order in which they can be changed (Hanseth, et al., 1996; van Schewick, 2010, p. 138). But the distribution of networking infrastructures also implies distribution of control among heterogeneous actors over their architectural evolution (Tilson, et al., 2010b; Rodon & Silva, 2015), and hence the requirement for motivation, agreement, and coordination of change activities (Monteiro, 1998; van Schewick, 2010, p. 138; Wu, et al., 2013).

Consider again, the network layer of the Internet's core architecture. In addition to the Internet Protocol, *routing protocols* exist at this layer. Routing protocols prescribe how data paths through a network are discovered (Tanenbaum, 2003, pp. 350-384; Lammle, 2011, pp. 347-354; Kurose & Ross, 2013, pp. 334-348,389-439). Although the Internet's core architecture does not prescribe the balance of responsibility between hardware and software for implementing its protocols, for reasons of performance, it has become the de facto standard architectural strategy, to implement routing in network hardware devices, i.e., in routers, with the Internet Protocol's implementation distributed across hardware and software (Kurose & Ross, 2013, pp. 346-348). In other words, over time, the way that core architecture is implemented, has itself become an invariant architectural feature in most IP-based networks.

This internal architecture of routers is the structural or *implementation architecture* (Maciaszek, et al., 2005, pp. 56-58; Taylor, et al., 2010, p. 65), which is distinct from the concept of deployment architecture, in that its details are hidden (though broadly known), and cannot be derived solely from inspecting networks' interconnectivity.

But this architecture matters, as there has developed a tight coupling between the abstract stipulations of the Internet's core architecture, and the de facto standard architectural implementation of those stipulations. In the context of distributed control, changes to the core architecture, may require a propagation of updated protocol implementations across a vast number of network devices deployed in networks, motivation of competing manufacturers of network hardware devices (e.g., such as Cisco Systems, Juniper Networks, HP, Alcatel-Lucent and others) to create and make available the updates, and motivation of the administrators of networks in which the devices are deployed, to update devices (Wu, et al., 2013). Significantly more problematic, are changes that necessitate adjustments to the implementation architecture, ossified within network hardware devices, not only due to technical difficulty (Anderson, et al., 2005; Chowdhury & Boutaba, 2010; Wu, et al., 2013), but also due to the interest that some organisations, such as manufacturers of network hardware devices and those that make or rely on complementary products and services, have in retaining the existing architecture.

Professional careers, complementary products and services, organisational structure and practice, and industry structure relate to the architecture of products (Henderson & Clark, 1990; Sanchez & Mahoney, 1996; Baldwin, 2008; Yoo, et al., 2010). Examples from the networking industry include, an industry structured around manufacturers of network hardware devices, owners of networking infrastructures, and network systems integrators, and certification programs that train networking professionals for careers premised on directly intervening at network hardware devices where they can configure routing behaviour (e.g., Cisco's Career Certification Program or Juniper's Certification Program (Cisco Systems, n.d.-a; Juniper Networks Inc., n.d.)). Convincing diverse, economically motivated social actors to embrace core architectural evolution or associated implementation architectural evolution is a considerably challenging exercise in *installed base cultivation* (Monteiro, 1998; Hanseth & Lyytinen, 2010; Aanestad & Jensen, 2011; Sanner, et al., 2014; Grisot, et al., 2014).

Architectural evolution is not strictly a technical exercise, as architecture is interrelated with complex social actors and economic systems, in ways that create and sustain ossifications, reinforced by positive network externalities, leading to a state of increasing irreversibility (Hanseth, et al., 1996; Monteiro, 1998; Edwards, et al., 2007; van Schewick, 2010, p. 138; Hanseth & Lyytinen, 2010; Egyedi & Spirco, 2011).

2.4 Introduction to Networking

2.4.1 Introduction

Having explained architectural evolution, and sociotechnical ossification that resists certain types of architectural evolution, a technical introduction to networking infrastructures is presented next. Indeed, understanding general characteristics of digital infrastructures which apply to networking infrastructures is important. However, to grasp the findings of this research requires understanding some basic technical details of how networking infrastructures work. A comprehensive tutorial on networking is beyond the scope of this thesis, and many of the particularities of networking are not pertinent to the answer to the research question. In what follows, an ample but simplified introduction to networking concepts that are central to an understanding of the research's findings is given.

2.4.2 The Function of a Network

Networks are communication infrastructures that facilitate interconnectivity and data transportation between geographically or logically dispersed digital hardware and software entities (Benkler, 1998; Hanseth & Lyytinen, 2010; Kurose & Ross, 2013, p. 31). The primary function of networks is the facilitation of interconnectivity and transportation of digital data from some source at a physical or logical location to a destination at another physical or logical location (Benkler, 1998; Hanseth & Lyytinen, 2010; Kurose & Ross, 2013, p. 31). *Switches* and *routers* are the physical network devices primarily tasked with enabling this functionality (Tanenbaum, 2003, pp. 19-20,326-328; Lammle, 2011, pp. 3-28; Kurose & Ross, 2013, pp. 331-352,506-508). The distinction between the two is extraneous to this thesis, but for clarity, routers, where a router is defined as a network layer packet switch (Kurose & Ross, 2013, p. 506), is exclusively referenced in relation to the findings in this thesis.

2.4.3 The Forwarding Function

As visually illustrated in Figure 2-2, each individual router within a network has input links or *ports*, via which data is received and output ports onto which data is transferred to an intermediary or final destination (Lammle, 2011, pp. 3-28; Kurose & Ross, 2013, pp. 331-352).

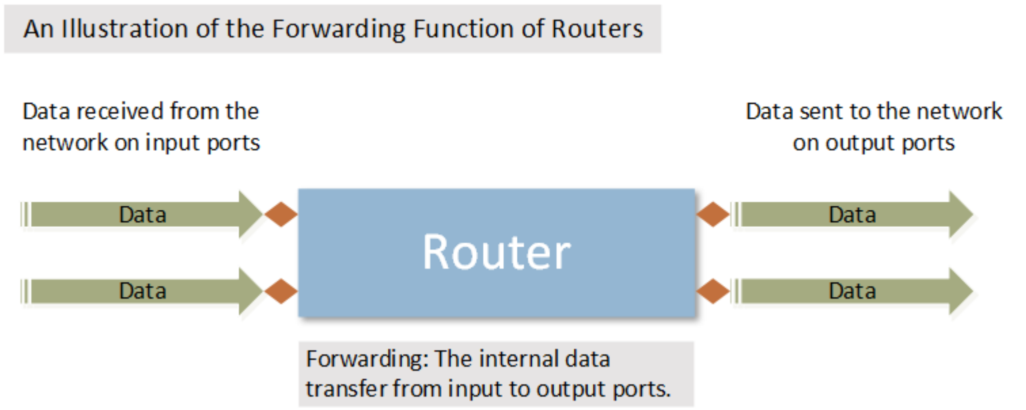

Figure 2-2 The Forwarding Function of Routers

The internal transfer of data from an input port to its appropriate output port ***within a router*** is called *forwarding* (Tanenbaum, 2003, p. 350; Kurose & Ross, 2013, pp. 334-336,346-348). The forwarding function of multiple routers in a network is what collectively realises the functionality of networks – interconnectivity and transportation of data from a source to its destination.

2.4.4 The Control Function

A router determines the output port onto which received data should be placed with the assistance of its *control* function[5] (Lammle, 2011, pp. 3-28; Kurose & Ross, 2013, pp. 331-352).

[5] The control function is tasked with other responsibilities, but only an explanation of routing is relevant for answering the research question.

Whereas the forwarding function is a conceptually simple internalised process of, looking up the assigned output port for data received on an input port using the destination information that labels the incoming data in a *forwarding table*, and then placing the data onto the appropriate output port identified in the forwarding table (Kurose & Ross, 2013, pp. 334-336, 346-348), the control function is not necessarily as strongly internally encapsulated.

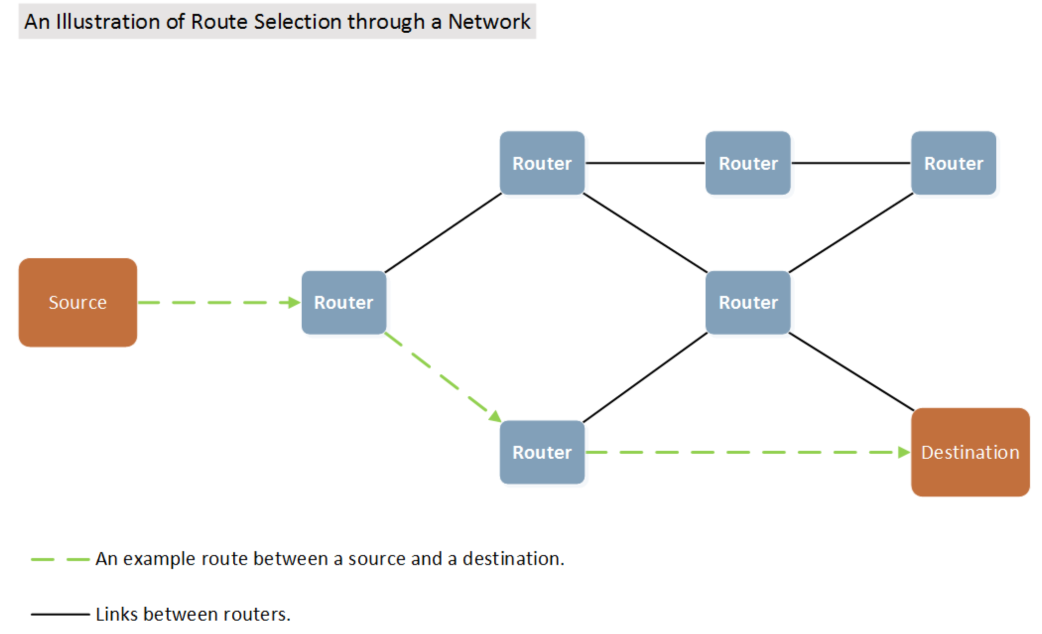

Figure 2-3 Route Selection through a Network

An implicit non-functional requirement of a networking infrastructure is the adherence to performance and data delivery expectations stipulated by standards, network operators, and its ultimate end users. Consequently, routers are tasked with being aware of the best path *or route* through the network from a source to a destination and this is facilitated by *routing algorithms* which attempt to determine these routes through the network (Tanenbaum, 2003, pp. 350-384; Lammle, 2011, pp. 347-354; Kurose & Ross, 2013, pp. 334-348,389-439). Figure 2-3 visually illustrates what an optimal route from some source to a destination in a network might be, as calculated via routing algorithms.

In networks, the control function of <u>each</u> router executes routing algorithms as specified by routing protocols (mainly Routing Information Protocol (RIP), Open

Shortest Path First (OSPF), Intermediate System to Intermediate System (IS-IS), and Border Gateway Protocol version 4 (BGP4)), to calculate routes through a network, and each router exchanges the calculated best routes or *routing information* with other routers. The control function of each router processes received routing information and updates a *routing table* of best routes (Lammle, 2011, pp. 20-21; Kurose & Ross, 2013, pp. 334-347, 389-439). As well, the control function may be manually invoked and updated by direct human intervention with preferred routing information (Lammle, 2011, p. 347; Kurose & Ross, 2013, p. 392).

A distinction between the forwarding and control functions is that while the forwarding is internal to a router, the routing responsibility of the control function is distributed across the network as a collaborative effort of its routers (Kurose & Ross, 2013, p. 357).

2.4.5 The Relationship between the Forwarding and Control Functions

The relationship between the forwarding and control functions is illustrated in Figure 2-4. The control function updates the forwarding table with simplified decision making information (i.e., for selecting the appropriate output port based on destination information that labels incoming data) taken from the more elaborate routing table that contains network route information (Kurose & Ross, 2013, p. 335).

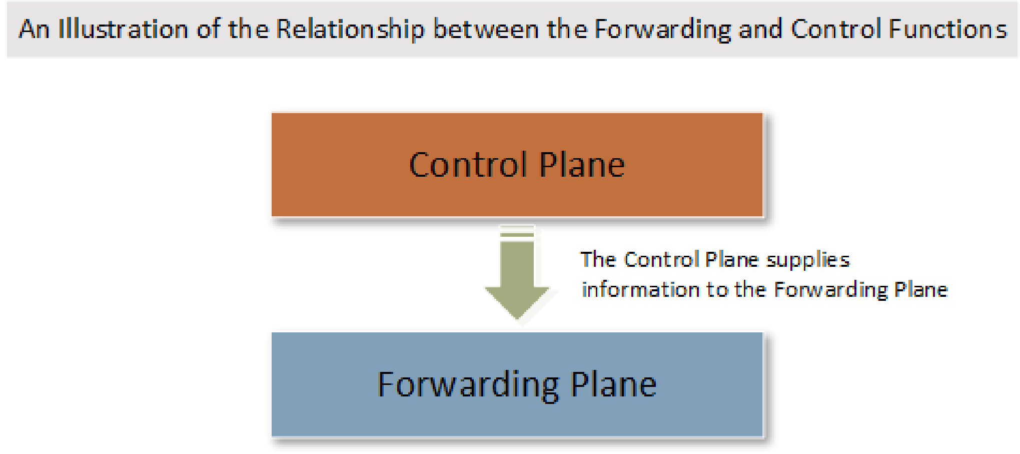

Figure 2-4 The Relationship between the Forwarding and Control Functions

In networking, the demarcation of forwarding and control functionality, though co-located within routers, is referred to as a *logical* stratification of routers into a *forwarding plane* and a *control plane* (Kurose & Ross, 2013, pp. 347-348).

2.4.6 Functionality is Embedded within the Hardware

It is important to reiterate the co-location of the forwarding and control planes within a single physical router in spite of the *internal* logical stratification.

For emphasis, taking a standard Cisco router architecture as an example, the forwarding function of the logical forwarding plane is implemented in hardware via the use of application-specific integrated circuits (ASIC), with the forwarding table of the forwarding plane typically stored in specialised [hardware] memory known as ternary content-addressable memory (TCAM) (Lammle, 2011, pp. 24, 278-279; Kurose & Ross, 2013, pp. 346-350).

Similarly, the control function of the control plane, though implemented as software, executes within the confinement of a routing processor implemented in the hardware of the router, and this software is stored on read-only memory (ROM) or on Flash Memory in the router (Lammle, 2011, pp. 278-279; Kurose & Ross, 2013, pp. 346-350). Manually configured routing information is typically made persistent by storing them with other router configuration settings in non-volatile random access memory (NVRAM), such that on start-up of a router these static routes are loaded into the routing table (and dynamic routes re-generated) in random access memory (RAM) (Lammle, 2011, pp. 372-373).

The main point being made here through the provision of this kind of technical detail of functionality embedded in hardware, is that the control plane and forwarding plane, that is, what makes a router a router, are tightly coupled to the hardware of the physical router. This is the de facto standard implementation architecture of routers.

2.5 Understanding Digital Materiality

2.5.1 Introduction

According to extant IS theorising, digital infrastructures are shared, continuously evolving and expanding, collectives of heterogeneous IT capabilities, along with their communities of users, innovators, and organisational and institutional structures and forces that sustain them (Edwards, et al., 2007; Hanseth & Lyytinen, 2010; Tilson, et al., 2010b). Digital infrastructures are never designed *de novo*, but instead arise as the outcome of reproduction, or extensions and changes to earlier infrastructures (Star & Ruhleder, 1996; Hanseth & Lyytinen, 2010; Tilson, et al., 2010b; Aanestad & Jensen, 2011), or out of the adaptation and transfer of local information systems into broader non-local contexts (Star & Ruhleder, 1996; Edwards, et al., 2007). Although they undergo change, digital infrastructures are characteristically perpetual (Star & Ruhleder, 1996; Hanseth, et al., 1996; Edwards, et al., 2007; Ribes & Finholt, 2009). These are important characteristics of digital infrastructures, but they do more to explain the term “infrastructure” than they do to explain what is meant by “digital”.

This section considers what is meant by “digital”. In so doing it produces a somewhat abstract definition of the make-up of digital infrastructures, that provides a highly important foundation for the later analysis.

2.5.2 Digital Materiality

The terms *materiality* and *digital materiality* have been used in IS research to describe what constitutes digital entities such as digital infrastructures, but there remains, nonetheless, a lack of consensus on what the objects of study referred to by these terms are (Kallinikos, et al., 2012; Leonardi, 2012).

At times, studies of the materiality of digital entities seek to explicate technical constitution and distinctive characteristics, and to relate these in an explanatory manner to social phenomena suffused by digital entities (Manovich, 2002, pp. 27-48; Kallinikos, et al., 2010; Yoo, et al., 2010; Blanchette, 2011; Kallinikos, et al., 2013). Other studies elude the particularities of technical constitution and characteristics, to focus on the space of interaction between digital entities and humans. They conceptualise digital entities as being relatively meaningless outside of a context of practice, and suggest that whatever defines materiality, is constituted through interaction (Leonardi, 2010; Faraj & Azad, 2012). That is, either social interaction completes the account of the materiality of digital entities (Faulkner & Runde, 2009), or only actions and their consequences are material (Pentland & Singh, 2012).

Studies of *sociomateriality* go further to suggest an *"inseparability of meaning and matter"* (Scott & Orlikowski, 2014, p. 873), such that the digital entity itself is in some sense ontologically materially co-constituted in practice (Orlikowski, 2007; Orlikowski & Scott, 2008; Scott & Orlikowski, 2012; Scott & Orlikowski, 2014). A few studies have tried to acknowledge the close intertwining of the social and the technical of interest in sociomateriality, while avoiding a fusion of epistemology and ontology, and ontological co-constitution of digital entities through practice, by treating materiality of digital entities, and materiality constituted through interactions of digital entities and humans as ontologically separate (Leonardi, 2011; Leonardi, 2013). Still others argue that some of the materiality that constitutes digital entities is really an *immateriality* (Kallinikos, 2012; Faulkner & Runde, 2013). (See also (Leonardi, 2010; Pentland & Singh, 2012; Kallinikos, et al., 2012, pp. 8-9)).

Whatever the philosophical arguments on materiality, what remains inescapable is the existence of digital entities whose technical constitution cannot be avoided if factual rather than counterfactual or incomplete statements about their make-up are to be made. For this research, it was particularly important to leverage an understanding of the technical constitution of infrastructures indicated by the term "digital" distinctly from aspects of the social, in order to grasp how architectural evolution in digital infrastructures circumvents sociotechnical ossification. The following section therefore explains another approach to articulating the materiality of digital infrastructures that involves considering abstract elements of what makes up digital entities such as digital infrastructures. These elements are *form*, *function*, and *matter,* and they are complemented by the concepts of *bearers* and *binary signification.*

2.5.3 Digital Materiality in Abstraction

So what do these terms, bearer, form, function, and matter mean? Bearer refers to whatever carries out or helps to manifest function. For example, routers are bearers of function in networking infrastructures. A bearer is constituted of form and matter (Kallinikos, 2012; Faulkner & Runde, 2013; Crawley, et al., 2016, pp. 27-29;67-68). Broadly, matter gives things their embodying constitution, and may manifest as materials distinguished by intrinsic properties (Kallinikos, 2012). Form, *"provides the mold to which matter enters"* (Kallinikos, 2012, p. 71). It imposes organisation onto the matter that participates in, or is constitutive of the bearer. Quoting Crawley, et al. (2016) who elaborate form:

> *"Form has shape, configuration, arrangement, or layout. Over some period of time, form is static and perseverant (even though form can be altered, created, or destroyed). Form is the thing that is built; the creator of the system builds, writes, paints, composes, or manufactures it. Form is not function, but form is necessary to deliver function."* – ***(Crawley, et al., 2016, pp. 27-29)***

> *"<u>Form is the physical or informational embodiment of a system that exists or has the potential for stable, unconditional existence, for some period of time, and is instrumental in the execution of function</u>. Form includes the entities of form and the formal relationships among entities. Form exists prior to the execution of function."* – ***(Crawley, et al., 2016, p. 68)*** *(Underline emphasis added.)*

Function refers to the activities, actions, purposes for which a thing exists, or that it carries out (Kallinikos, 2012; Crawley, et al., 2016, p. 97). Quoting again Crawley, et al. (2016) on function:

"Function is what a system does; it is the activities, operations, and transformations that cause, create, or contribute to performance[6]*. Function is the action for which a thing exists or is employed. Function is not form, but function requires an instrument of form.*

…

Function consists of a process and an operand. The process is the part of function that is pure action or transformation, and thus it is the part that changes the state of the operand. The operand is the thing whose state is changed by that process. Function is inherently transient; it involves change in the state of the operand (creation, destruction, or alteration of some aspect of status of the operand)." – ***(Crawley, et al., 2016, p. 29)*** *(Underline emphasis and footnote annotation added.)*

Traditionally, it has been that form, function and matter share a close relation where, depending on the perspective from which they are considered, function proscribes types of physical materials unsuitable for executing it, and form bears upon what functions are possible or are occasioned (Kallinikos, 2012). Alternatively, it may be said that matter – intermediated by its configuration, that is, its form – admits certain types of function and not others (Kallinikos, 2012). This remains true and is straightforward to understand for physically executed (via manual or mechanical operation) function. Digital infrastructures are not confined primarily to physical execution of function against a physical operand. As a corollary, form, function, and matter relate differently to one another. The relationship between form, function and matter in digital infrastructures is mediated by another abstract concept called *binary signification*, which helps to account for key differences with the traditional relationships between these abstract elements.

[6] In this quote, the authors are referring strictly of how well technical execution of function is carried out (Crawley, et al., 2016, pp. 24-25).

Binary signification refers to the representation of information, regardless of its ultimate meaning, as basic sequences of one of two binary values: '1' or '0' (Yoo, et al., 2010; Blanchette, 2011; Berry, 2011, pp. 54-55; Faulkner & Runde, 2013). It is a standardisation and homogenisation of how information is semiotically captured (Yoo, et al., 2010; Berry, 2011, pp. 54-55; Kallinikos, 2012). Binary signification, allows function (i.e., what actions must be carried out) in digital infrastructures (and in digital entities in general) to be encoded in abstraction from whatever bearer eventually executes it (Blanchette, 2011; Berry, 2011, pp. 94-97). The bearer must only understand binary signification, that is, it must be digital (Kallinikos, et al., 2010; Kallinikos, et al., 2013), and be able to interpret a function expressed in binary signification to carry out the function.

Thus, binary signification is implicated in a looser coupling within digital infrastructures, of relationships between form, function, and matter (Kallinikos, 2012), in that *definition of* function *can be* decoupled from and be indifferently transferrable between different bearers. In other words, function is not necessarily tied to a particular *instance* of a bearer (with its form and matter). Still, binary signification is manifested via matter that admits its representation (such as silicon transistors, magnetic variations, optical patterns, electric voltages, physical pits on plastic, etc. (Berry, 2011, pp. 96-97; Blanchette, 2011)), though as a type of signification, it remains logically indifferent to and decoupled from bearers, and from what it signifies – here, definition of function, but also the operands of function, i.e., data (Yoo, et al., 2010; Kallinikos, et al., 2010; Blanchette, 2011; Berry, 2011, pp. 54-55; Kallinikos, et al., 2013).

In summary, digital infrastructures are technologically constituted by digital materiality, and permeated by binary signification (Tilson, et al., 2010b). Irrespective of any variations in thought on digital materiality in the IS literature, what remains invariant are these abstract elements of form, function and matter. Moreover, the word "digital" in the term "digital materiality" connotes the involvement of binary signification. As defined, at a fundamental level, the social lies outside of this technologically constituting digital materiality, though the social may historically bear upon it, or become intertwined with it (Leonardi, 2011; Kallinikos, 2012; Leonardi, 2012; Kallinikos, et al., 2013) such as through processes of standardisation which prescribe what is manifested through form, function, matter and their bearers[7] (Hanseth, et al., 1996; Hanseth & Monteiro, 1997; Hanseth, 2000; Iannacci, 2010; Eriksson & Ågerfalk, 2010; Egyedi & Spirco, 2011).

2.6 Chapter Summary

This chapter presented related literature that clarified the meanings intended by each aspect of the theoretical construct in the research question. Three types of architectural evolution were presented: deployment architectural evolution from a descriptive perspective which is the outcome of continued expansion of digital infrastructures, core architectural evolution which is change in prescribed architecture that is present throughout a type of digital infrastructure such as the Internet architecture in networking infrastructures, and implementation architectural evolution which is change in the internal constitution of IT components in digital infrastructures. The formation of social and technical dependencies on digital infrastructures and the interrelationships between these dependencies was explicated to establish what is meant by sociotechnical ossification.

Next details of how networking infrastructures work which are pertinent to the case study and to the findings of this research were presented. Of primary importance is the function of the control and forwarding planes, the relationship between the planes, and the co-location of these functional planes within the confines of network hardware devices called routers. Finally, because as will later be shown the architectural evolution in digital infrastructures which circumvents sociotechnical ossification relies on an exploitation of the digital materiality of digital infrastructures,

---

[7] For this reason, throughout this thesis, digital infrastructures remain characterised as being sociotechnical, in spite of the abstract definition of digital materiality.

an explanation of digital materiality in terms of abstract elements of form, function, matter, complemented by concepts of bearers and binary signification was presented.

The foundation provided in these four aspects of the research question's theoretical construct are sufficient for understanding the case study, and the later analysis and theoretical contributions of this thesis.

3 Theory

3.1 Introduction

This chapter introduces *critical realism* as the philosophical basis of this research. Methodological implications of critical realism are presented in the Methodology chapter. In this introduction, an overview of key principles of critical realism is provided.

Critical realism holds a philosophical position on ontology, that was developed as an alternative to anthropocentric philosophies that define ontology in terms of what humans experience, or are able to experience (Bhaskar, 1998a; Hartwig, 2007, pp. 40-41). In concordance with *realist* ontology (Collier, 1994, pp. 3-30; Crotty, 1998, pp. 10-11), critical realism emphasizes that there is a distinction between knowledge and methods of knowledge production, and the objects *about which* knowledge is. The distinction demarcates knowledge into two dimensions: an *intransitive* dimension and a *transitive* dimension (Collier, 1994, pp. 3-30,50-51; Bhaskar, 1998a; Sayer, 2000, pp. 10-28).

Within the intransitive dimension of knowledge are the objects of knowledge – that is, the objects about which knowledge is. For the natural world, these objects of knowledge are *enduring*, whereas in the social, the objects of knowledge, though considered intransitive, are *relatively enduring* (Archer, 1982; Collier, 1994, pp. 137-168; Archer, 1995, pp. 135-161; Archer, 1998; Bhaskar, 1998a; Engholm, 2007). Nonetheless, for both the natural world and social reality, as per critical realist conceptualisation of reality, objects of knowledge exist and endure independently of humans and their knowledge or experience of these objects.

But what are these objects of knowledge? Critical realism, premised on *transcendental realism* (Collier, 1994, pp. 3-30; Bhaskar, 1998a; Norris, 2007), suggests that these are *real* entities that constitute reality, and are constituted of what is called *structure,* along with structure's *properties and generative mechanisms* (Archer, 1982; Collier, 1994, pp. 7-12,31-69,107-115,137-151; Archer, 1995, pp. 135-161; Archer, 1998;

Bhaskar, 1998a; Archer, 2000). Structure with its properties, through the operation of its generative mechanisms, give rise to *events* (whatever these events may specifically be), some of which, though not all, may be noticeable or noticed by humans (Collier, 1994, pp. 7-12,31-69; Bhaskar, 1998a; Sayer, 2000, pp. 10-17). Critical realism thus suggests that when events are empirically captured or experienced, causality is attributed to the operation of generative mechanisms that produced them (Bhaskar, 1998a; Sayer, 2000, pp. 10-28; Psillos, 2007; Hartwig, 2007, pp. 57-60; Pinkstone & Hartwig, 2007). Generative mechanisms and causality are detailed later in this chapter.

While the objects of knowledge are intransitive, what is deemed knowledge or knowable, and the means by which knowledge is attained, that is, the ways of doing natural or social science, is provisional. Knowledge and its methods of production are continuously subject to change, correction, refinement, and replacement over time. These belong to the transitive dimension of knowledge (Collier, 1994, pp. 50-51; Bhaskar, 1998a; Sayer, 2000, pp. 10-28).

Aside from indicating a critique of other philosophical perspectives, the word critical, in critical realism, connotes that the more precisely that underlying reality is expressed, the more likely it is to be evaluative (Collier, 1994, p. 178).

On the basis of this introduction, the remainder of the chapter presents key principles of critical realism, and reasons for the choice of critical realism as this research's underlying philosophical perspective.

3.2 A Stratified Ontology

An anthropocentric philosophy tends to obscure the distinction between knowledge and its methods of production, and the objects of knowledge, leading to the mistake of conflating epistemological statements *about* being, with being itself, i.e. *ontology* (Bhaskar, 1998a; Hartwig, 2007, pp. 40-41). For instance, philosophy premised on empirical realism pre-supposes the observability of whatever exists, and commits to *actualism* which attributes the make-up of reality entirely to events that occur (Collier, 1994, pp. 7-12,75,107-134; Bhaskar, 1998a; Sayer, 2000, pp. 10-17; Hartwig, 2007, pp.

14-16). This, critical realism contends, is an epistemic fallacy in that what through empirical capture can be postulated about reality, is somehow erroneously equated with the make-up of that reality (Collier, 1994, pp. 76-85; Bhaskar, 1998a; Norris, 2007; Hartwig, 2007, pp. 173-175). With a basis in transcendental realism, critical realism does not allow this conclusion. Rather, in response to the transcendental question of, "What must the world be like for some phenomenon (such as an event) to exist or come into being?" it directs attention towards ontology, arguing that the epistemic fallacy is eluded if a *stratified* ontology is substituted for an otherwise flat ontology consisting only of events (Collier, 1994, pp. 20-29,31-69,107-134; Bhaskar, 1998a; Sayer, 2000, pp. 10-17; Norris, 2007; Hartwig, 2007, pp. 116-119; Archer, 1998, p. 196).

Critical realism contends that experiments cultivate conditions within a closed environment that aim to exclude countervailing factors that normally exist in an otherwise *open* environment. In an open environment, these countervailing factors would be less inhibited to intervene and to disrupt the perceived empirical regularity of antecedent causal event and consequent event that has been mistaken for ontology in empiricist philosophy (Collier, 1994, pp. 7-12,31-45; Bhaskar, 1998a; Sayer, 2000, pp. 10-17; Psillos, 2007; Pinkstone & Hartwig, 2007; Norris, 2007; Hartwig, 2007, pp. 14-16,57-60). More than this, social reality is fundamentally open, with humans that are as well intrinsically internally open and able to act in ways not derived from externally imposed conditions (Collier, 1994, pp. 128-129; Archer, 1995, p. 166). Thus, there is an ontological distinction between events, and real underlying causal generative mechanisms (Collier, 1994, pp. 7-12,31-45; Bhaskar, 1998a; Sayer, 2000, pp. 10-17; Psillos, 2007; Pinkstone & Hartwig, 2007; Norris, 2007; Hartwig, 2007, pp. 14-16,57-60). Further, critical realism argues that given the very methods of knowledge production (whether for natural or social sciences) are continuously subject to change, the knowledge that they produce about objects of knowledge, and the underlying objects of knowledge themselves can be out of phase with each other – suggesting that there is an independent existence of the two (Collier, 1994, pp. 50-59; Bhaskar, 1998a; Sayer, 2000, pp. 10-28).

On the basis of this argumentation, the first part of the answer to the transcendental realist question of, "What must the world be like for some phenomenon to exist or to come into being?" is that there is an ontological stratification of reality into at least two domains: the *domain of the real* in which there are objects of knowledge, and the *domain of the empirical*, in which there are empirically captured events and methods of knowledge production (Collier, 1994, pp. 42-45; Bhaskar, 1998a; Sayer, 2000, pp. 10-11).

The second part of the answer is that there is an intervening third domain: *the domain of the actual* (Collier, 1994, pp. 42-45; Bhaskar, 1998a; Sayer, 2000, pp. 10-11). Knowledge and its methods of production are transitive and may be out of phase with the intransitive objects of knowledge that exist in the domain of the real. In other words, counter to empirical realism, existence of the real is not dependent on empirical observability (Collier, 1994, pp. 70-85; Bhaskar, 1998a; Sayer, 2000, pp. 10-17). Critical realism claims that while some events generated by objects of knowledge in the domain of the real, may become known, this is not so for all events. Therefore, there is a domain of events that have manifested, *the domain of the actual*, of which a subset of events may exist in the domain of the empirical as known or knowable events (Collier, 1994, pp. 42-45; Bhaskar, 1998a; Sayer, 2000, pp. 10-12; Hartwig, 2007, p. 316). Importantly, critical realism points out that whether an event is classified as knowable may be dependent on the state of methods of knowledge production at a point in time (Bhaskar, 1998a). By taking this as fundamental, critical realism as a philosophy aims to avoid the mistake of formulating foundational philosophical arguments on the basis of whatever is the current state of methods of knowledge production (Bhaskar, 1998a).

The three domains of the real, the actual and the empirical, form the stratified ontology postulated by critical realism. This stratified ontology does not mean that critical realism proposes an ontology in which some entities are less real than others, such that events are less real than structure, properties and generative mechanisms (Bhaskar, 1998a). The domains are in a subsumptive relationship (Collier, 1994, pp. 42-45; Bhaskar, 1998a; Sayer, 2000, pp. 10-12). The real is the domain of everything. The actual is a subset of the real. The empirical is a subset of the actual. Thus, the constituents of each domain are very much real (Bhaskar, 1998a). There is another type of stratification: a stratification of the objects of knowledge (Collier, 1994, pp. 45-

50; Bhaskar, 1998b, pp. 66-67). I return to this in the discussion of emergence in section 3.5.2.

3.3 Generative Mechanisms

Generative mechanism, *power* and *tendency*, are three interrelated concepts of critical realism. Power is a general term that connotes what a thing[8] (whatever such thing is), can do or does (Collier, 1994, p. 62). Generative mechanism, on the other hand is a term used in critical realism to indicate intransitivity, and a distinction between underlying causation in the domain of the real, and events in the domain of the actual and empirical. It captures an enduring characteristic *way of acting* of a thing (Collier, 1994, p. 62; Bhaskar, 1998a). That a generative mechanism is a way of acting of a thing, does not mean that the generative mechanism is *exercised*, or in other words, in operation (Collier, 1994, pp. 61-69; Bhaskar, 1998a; Sayer, 2000, pp. 10-17; Psillos, 2007; Hartwig, 2007, pp. 57-60; Fleetwood, 2009; Fleetwood, 2011).

Tendency[9] is a term used in critical realism to denote a power that is exercised (Collier, 1994, pp. 61-69; Bhaskar, 1998a; Psillos, 2007; Pinkstone & Hartwig, 2007; Hartwig, 2007, pp. 57-60; Fleetwood, 2011). Tendencies do not always *actualise* to produce events, as countervailing factors such as the exercise of other tendencies may impede such actualisation (Collier, 1994, pp. 61-69; Bhaskar, 1998a; Psillos, 2007; Pinkstone & Hartwig, 2007; Hartwig, 2007, pp. 57-60; Fleetwood, 2011). But as per critical realism's stratified ontology, the lack of events (or un-observability of events) is ontologically separate from the exercise of a tendency. It is claimed then that, tendencies act

---

[8] Following Fleetwood (2009, p. 346), the generic word "thing" is used here to refer to anything physical, artefactual, social, or ideal. Note that, in critical realism, powers are not of structures only. People, for example, possess powers. See for instance (Archer, 1995, pp. 183-190).

[9] Though prominent in critical realism and foundational to causal explanation in open environments (Collier, 1994, p. 63), some question the ontological distinctiveness of tendencies and of kinds of tendencies (Collier, 1994, pp. 123-130; Fleetwood, 2011).

transfactually in that a power *is acting* (whether exercised in closed or in open environments), although its effects may not be manifest (Collier, 1994, pp. 61-69; Bhaskar, 1998a; Psillos, 2007; Pinkstone & Hartwig, 2007; Hartwig, 2007, pp. 85-87; Fleetwood, 2009; Fleetwood, 2011).

Transfactual conditionals are not the same as *counterfactual* conditionals. Counterfactual conditionals describe hypotheticals of what would happen if antecedent conditions were met for a power to be exercised, but ontologically the power is not really exercised. In contrast, transfactuality asserts that ontologically something is really occurring, namely that a power is exercised, but that it's exercise may not actualise events (Bhaskar, 1998a; Psillos, 2007; Pinkstone & Hartwig, 2007; Hartwig, 2007, pp. 85-87; Fleetwood, 2009).

There is an enumeration of eight tendencies which are ontologically classified by which of their *intrinsic* or *extrinsic* (Hartwig, 2007, pp. 265-266) *enabling* conditions that make the tendency possible – the counterpart being constraint which prohibits it, *stimulating* conditions that *"trigger, facilitate or reinforce the exercise of a tendency"* (Pinkstone & Hartwig, 2007, p. 458), and *releasing* conditions (insufficient countervailing factors) are satisfied (Hartwig, 2007, pp. 57-60,80,344-345; Pinkstone & Hartwig, 2007; Fleetwood, 2011). The specificities of the enumeration did not add greater explanatory power to this research. Therefore, generative mechanism, which denotes intransitivity and captures both power and tendency (Psillos, 2007, p. 57), is the term used in the Analysis chapter of this document. Notwithstanding, enabling, stimulating and releasing conditions of generative mechanisms sought by the research question are identified. Like events, generative mechanisms might not be directly observable. It is for this reason that critical realism establishes the reality of generative mechanisms through retroductive argumentation (Collier, 1994, pp. 22-23). Retroduction is discussed in Chapter 4.

3.4 Causality and Explanation

Critical realism's stratified ontology is accompanied by a formulation of causality that differs from a regularity-based model of causation (Collier, 1994, pp. 7-12, 31-69;

Bhaskar, 1998a; Sayer, 2000, pp. 10-17; Psillos, 2007). In the latter, causation is formulated as a conjunction of events such that whenever an antecedent event occurs, one or more causally related outcome events occur, or put differently, if some event *y* has occurred, it is only because a preceding causal event *x* had occurred. Causal attribution is made directly to the antecedent event due to perceived regularity in the succession of antecedent and subsequent events. Therefore, causal explanation is articulated in terms of the succession of events (Collier, 1994, pp. 7-12,31-69,75; Bhaskar, 1998a; Sayer, 2000, pp. 10-17; Psillos, 2007; Hartwig, 2007, pp. 14-16,57-60).

In contrast, for critical realism, a sequence of events can express some regularity or be used for prediction, but it does not constitute causation. Via its proposed stratified ontology, critical realism interjects between the seemingly contiguous antecedent and subsequent events, the operation of causal generative mechanisms, instantiated with real structures and the structures' properties that produce the subsequent events (Collier, 1994, pp. 7-12,31-69,75; Bhaskar, 1998a; Sayer, 2000, pp. 10-18; Psillos, 2007; Hartwig, 2007, pp. 14-16,57-60). Attribution of causality to the operation of generative mechanisms thus forecloses the notion that causation can be wholly captured by causality articulated as successive events (Collier, 1994, pp. 7-12,31-69,75; Bhaskar, 1998a; Sayer, 2000, pp. 10-18; Psillos, 2007; Hartwig, 2007, pp. 14-16,57-60). It also fundamentally accommodates causal explanation where the investigatory environment is not closed, but instead is open to the operation and interaction of various generative mechanisms that may bear upon each other in ways causal to the outcome or non-outcome of events (Collier, 1994, pp. 31-60,122-130; Bhaskar, 1998a; Sayer, 2000, pp. 10-17; Psillos, 2007; Hartwig, 2007, pp. 57-60). Causal explanation in critical realism involves the identification and exposition of generative mechanisms, the operation of which may *contingently* produce some outcomes (Collier, 1994, pp. 107-130,169-181; Bhaskar, 1998a; Sayer, 2000, pp. 10-17; Pratten, 2007; Hartwig, 2007, pp. 57-60).

Because generative mechanisms are instantiated with structures and the structures' properties, critical realist causal explanation includes some elucidation of structure which itself has causal efficacy not directly synonymous with the operation of its generative mechanisms (Archer, 1982; Collier, 1994, pp. 107-134; Bhaskar, 1998a;

Archer, 2000; Sayer, 2000, p. 14; Pratten, 2007). I return to the causal force of structure in section 3.5.3, when discussing how structure impinges on human agency.

3.5 Structure

3.5.1 What is Structure and what are Properties?

First, in this research, the structures of relevance are not naturally occurring structures (of this world or of the universe). They are artefactual – specifically technical – and social. Formative critical realist writings, do not specifically address structures that are technological – and this is a point of criticism to which I return at the end of this chapter, and again in the Discussion chapter. Accordingly, this section explains structure in terms of critical realist claims about social structures.

Recall that in critical realism the objects of knowledge are constituted of what is called structure, along with structure's properties and generative mechanisms (Archer, 1982; Collier, 1994, pp. 7-12,31-69,107-115,137-151; Archer, 1995, pp. 135-161; Archer, 1998; Bhaskar, 1998a; Archer, 2000). Structure is an abstract term, devoid of specificities of any particular object of knowledge, which is explained as a constitution of *necessary internally related* constituent objects of knowledge (henceforth referred to as "parts" or "relata") and the relations between these parts (Archer, 1995, p. 173; Archer, 2000; Sayer, 2000, p. 14). Further, structure, *"suggests a set of internally related elements whose causal powers, when combined, are emergent from those of their constituents"* (Sayer, 2000, p. 14). The internal relations are designated necessary, because any structure's existence is dependent on a particular arrangement of encapsulated relations which are not, and cannot be, contingent or external to the structure (Archer, 1995, p. 173).

Properties describes the ways in which the arrangement of necessary internal relations give structure its characteristics that underpin its generative mechanisms (Archer, 1995, pp. 172-183; Elder-Vass, 2007; Fleetwood, 2009).

3.5.2 Structure as Emergent

In critical realism, structure, properties, and generative mechanisms are explained heavily on the basis of *emergence.* The emergence principle asserts that although structure is dependent on the presence of its necessary internal relations and their relata, and that fundamental changes to these may see corresponding change in structure, structure is not entirely reducible to the aggregate of its parts (Archer, 1982; Collier, 1994, pp. 107-134,138-141; Archer, 1995, pp. 135-161,172-183; Archer, 1998; Archer, 2000; Morgan, 2007a; Elder-Vass, 2007). This means that any structure's properties and generative mechanisms are *sui generis*, and hence not of its parts (Archer, 1982; Collier, 1994, pp. 107-134,138-141; Archer, 1995, pp. 135-161,172-183; Archer, 1998; Archer, 2000; Elder-Vass, 2007). These *emergent properties* and *emergent generative mechanisms* exist only by virtue of *synchronic relations*: the specific arrangement of the necessary internal relations of parts *at a particular point in time* (Archer, 1982; Elder-Vass, 2007). Critical realism maintains that structure itself is emergent[10] (Archer, 1982; Collier, 1994, pp. 107-134,138-141; Archer, 1995, pp. 135-161,172-183; Archer, 1998; Archer, 2000; Elder-Vass, 2007). Moreover, the relata of structure may themselves be structures. Archer calls these, and the relations between them second-order and third-order emergent strata[11] (Archer, 1995, pp. 202-218), although structure may be comprised of multiple levels of emergent strata (Collier, 1994, pp. 107-134).

3.5.3 The Reality of Structure

Against emergence are arguments that question the reality of structure. These arguments are posed via the avenue of social structure to generally challenge the

---

[10] Elder-Vass, however, admits that not all social structures appear to be emergent (Elder-Vass, 2007).

[11] Archer's specific terminology here is "emergent properties" though not specifically limiting the scope of its referents to properties.

critical realist claim of there being intransitive structure, due to a fundamental difference between naturally occurring structures and social structures.

Social reality is *necessarily peopled* (Archer, 1995, pp. 195-218; Archer, 1998, p. 190). That is, social structures are dependent on the activities of people for their genesis, sustenance, and transformation (Collier, 1994, pp. 138-151; Archer, 1995, pp. 141-161; Archer, 1998; Archer, 2000; Engholm, 2007, p. 468). Thus, the causal generative mechanisms of social structures (e.g., organisations) are only efficacious through the mediation of human agency, unlike the generative mechanisms of nature (Porpora, 1989; Collier, 1994, pp. 138-151; Archer, 1995, pp. 141,149-154,193,195-201). Methodological individualism, a philosophical position that detracts from emergence, argues that what is deemed emergent properties and emergent generative mechanisms are ultimately reducible to the actions of contemporary individuals, and as such, structure is no more than an unnecessary reification of human agency into an abstract un-real entity (Archer, 1982; Porpora, 1989; Collier, 1994, pp. 110-115,138-141,143; Archer, 1995, pp. 33-46; Archer, 1998; Hartwig, 2007, p. 409).

To address this challenge to structure and emergence, critical realism argues that it is by incumbency of social positions (relata of social structure) already predisposed with causal generative mechanisms, that individuals act accordingly to these generative mechanisms (Porpora, 1989, pp. 196,199-200; Collier, 1994, pp. 137-151; Archer, 1995, pp. 147-154; Archer, 1998, pp. 200-203; Elder-Vass, 2007). When not occupying a particular social position, individuals may not and in some cases cannot mediate these generative mechanisms (Porpora, 1989, pp. 206-208; Collier, 1994, pp. 137-151; Archer, 1995, pp. 147-154; Archer, 1998, pp. 200-203; Elder-Vass, 2007). This difference between individuals and individuals *as incumbents of social positions*, indicates an ontological independence between individuals and social positions defined by social structures, and suggests that emergent properties and generative mechanisms are of social structures and are not entirely reducible to individual human agency (Porpora, 1989, pp. 206-208; Collier, 1994, pp. 137-151; Archer, 1995, pp. 147-154; Archer, 1998, pp. 200-203).

Continuing on the emergence and reality of structure, social structures cannot be understood apart from their historicity (Archer, 1982; Archer, 1995, pp. 149-161,165-

194). Though from a synchronic perspective, social structures may be sustained and transformed by contemporary individuals, from a diachronic perspective, their genesis, properties and generative mechanisms can usually be traced to actions of non-contemporaries of social positions including, as Archer puts it, individuals long dead (Archer, 1995, pp. 135-161; Archer, 1998). Yet these social structures endure and impinge upon human activity, shaping the circumstances of individuals, and mediating *to* them powers which they may exercise as incumbents of social positions (Porpora, 1989, pp. 206-208; Collier, 1994, pp. 137-151; Archer, 1995, pp. 137-161,195-201; Archer, 1998). The claim then, is that the historicity and pre-existence of structure, make social structure analytically distinguishable from the agency of individuals (Archer, 1982; Archer, 1995, pp. 149-161,165-194). This is a major argument which Archer, in particular, makes against the conflation of structure and agency as mutually constitutive of each other in the present tense (Archer, 1982; Archer, 1995).

This does not mean that the past actions of individuals are somehow erroneously transformed to have ontological status as structure, an accusation of methodological individualism (Archer, 2000; Elder-Vass, 2007). Critical realism recognises the difference between the diachronic continuity of activity and the synchronic continuity of individual agency (Archer, 1998, pp. 201-202). Structure presents social positions of which individuals may become incumbent. These social positions pre-exist their occupancy, and may over time be occupied by different individuals, breaking the continuity of any particular individual's agency in that position (Collier, 1994, pp. 137-151; Archer, 1998, pp. 201-202). Activity, however, does not necessarily break in concordance with a break in a particular individual's agency relative to a social position (Archer, 1998, pp. 201-202). Activity continues on the basis of the pre-existence – relative to individual incumbents at a particular time – of social positions in particular synchronic relations (Archer, 1998, pp. 201-202; Elder-Vass, 2007) that continue to be filled by new incumbents (Collier, 1994, pp. 137-151; Archer, 1998, pp. 201-202).

The reality of the emergence of structure as intransitive and of the domain of the real (Archer, 1982; Collier, 1994, pp. 7-12,31-69,107-115,137-151; Archer, 1995, pp. 135-161; Archer, 1998; Bhaskar, 1998a; Archer, 2000), is then established and maintained against counter-arguments, broadly on a combination of synchronic and diachronic argumentation (Archer, 1982; Elder-Vass, 2007).

3.5.4 The Relative Endurance of Social Structure

Social structures may change; they are not immutable (Collier, 1994, pp. 141-151; Archer, 1995; Archer, 1998, pp. 195-203; Engholm, 2007). Though social structures may condition the circumstances of individuals, social structures cannot deterministically confine human actions (Archer, 1982; Porpora, 1989; Collier, 1994, pp. 137-151; Archer, 1995, pp. 165,196-218; Archer, 1998, p. 190). Individuals may deliberately or unknowingly act in ways that lead to the transformation of social structures, and the properties and generative mechanisms instantiated with them (Archer, 1982; Collier, 1994, pp. 137-151; Archer, 1995, pp. 149-161,165-194,195-218; Archer, 1998; Engholm, 2007). As well on the other hand, the actions that generate, sustain and transform social structures may not even be those that mediate social structures' generative mechanisms (Collier, 1994, pp. 245-246; Outhwaite, 1998, p. 289).

Critical realism recognises that the transformation of social structures which exists intransitively in the domain of the real seems contradictory. The solution it proposes looks to the continuity of activity, and suggests that it is relations between social positions predisposed with generative mechanisms that endure in social structures, but given that social structures do undergo transformation, it is asserted that social structures are only relatively enduring[12] (Collier, 1994, pp. 150,244-245; Archer, 1995, p. 167; Archer, 1998; Engholm, 2007, p. 468).

---

[12] There are some criticisms of the degree to which social structures are relatively enduring (Collier, 1994, pp. 244-245; Benton, 1998, pp. 306-307), but these relate specifically to critical realism's arguments on the consequences for doing natural and social sciences and are not relevant to the concerns of this research.

3.5.5 The Transformation of Social Structure

Diachronically, social structure is either reproduced as it is, or transformed over time (Archer, 1982; Collier, 1994, pp. 141-151; Archer, 1995; Engholm, 2007). Social structure is transformed when existing necessary internal relations are modified or new ones created, otherwise it is reproduced (Archer, 1995, pp. 165-194). Transformation of social structure occurs in the context of conflicts between the structural conditioning of the circumstances of individuals, and the interests of these individuals (Archer, 1995, pp. 152,195-246). Two critical realist models of the transformation of social structure are the Transformational Model of Social Action (TMSA) (Collier, 1994, pp. 141-151; Engholm, 2007), and the Morphogenetic approach (Archer, 1982; Archer, 1995). Though compatible (Archer, 1995, pp. 135-161), in this thesis, where transformation of structure is addressed, it is the morphogenetic approach that was relied upon for analysis. The reason for this is that the morphogenetic approach avoids certain ambiguities present in the articulation of the TMSA model, which have catalysed criticisms of the reality of structure, and of the diachronic aspect of structure (See also (Archer, 1995, pp. 135-161)).

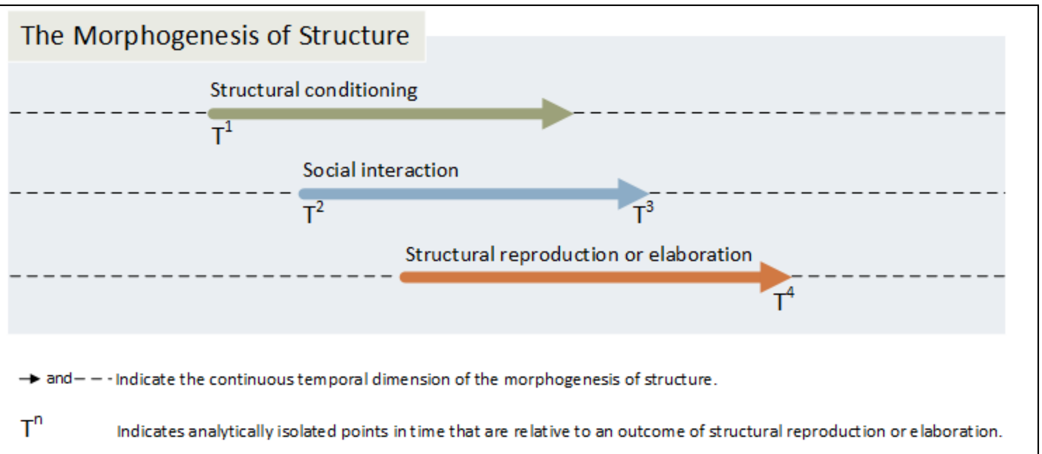

Figure 3-1 The Morphogenesis of Structure

The morphogenetic approach takes an explicitly diachronic perspective of the transformation of social structure, through human actions (Archer, 1982; Archer, 1995). Per the morphogenesis of structure[13], depicted in Figure 3-1 as proposed by Archer (Archer, 1982; Archer, 1995, pp. 157,193-194), social structure and human actions that mediate its generative mechanisms are distinctly separate, but exist simultaneously and temporally continuously with one another (Archer, 1982; Archer, 1995, pp. 149-161, 165-194; Archer, 2000). Moreover, social structure and human actions partake in an *interplay* of *necessarily anterior* structural conditioning of human actions, followed by social interaction that either leads to *morphostasis* – the reproduction of existing social structure, or *morphogenesis* – the transformation of structure, termed *structural elaboration* (Archer, 1982; Archer, 1995, pp. 149-161, 165-194; Archer, 2000). Archer asserts further that morphogenesis and morphostasis *are* generative mechanisms (Archer, 1995, p. 217). Specifically, what plays out from an analytically isolated temporal point of structural conditioning until a corresponding analytically isolated later temporal point of structural elaboration, *is a generative mechanism of either morphogenesis or morphostasis*:

> *"Therefore, insofar as a form of situational logic is strategically carried through, it represents the generative mechanism of either morphogenesis or morphostasis."* – ***(Archer, 1995, p. 217)***

Social structure that is the outcome of prior morphostasis or morphogenesis, subsequently conditions later human actions that either reproduce or transform it. That is to say, the morphogenesis of structure is articulated as a cycle, though it is temporally continuous (Archer, 1982; Archer, 1995, pp. 149-161,165-194; Archer, 2000). Accordingly, to understand any instance of structural elaboration, necessitates specificity about whose actions and which actions contributed to it, along with a diachronic analytic history of pertinent antecedent structural conditioning of those human actions (Archer, 1995, pp. 149-161,165-218; Archer, 1998).

---

[13] Archer explains three interrelated cycles of the morphogenetic approach: the morphogenesis of structure, culture and agency (Archer, 1995, pp. 165-194). Here, only the morphogenesis of structure is relevant.

3.6 A Word on the Relationship between Structure, Properties and Generative Mechanisms

Fleetwood explains that in critical realism there are three formulations of ontology for the domain of the real (Fleetwood, 2009). In the first, structure and properties are entirely derived from, and reducible to powers, such that powers are primary. The second treats properties as primary, and asserts that structure and powers are entirely derived from, and reducible to properties. Both formulations of ontology lead to unresolved challenges of what makes up, or sustains powers and properties in these ontologies (Fleetwood, 2009). A third formulation of ontology to which this research subscribed, is one of a unity of structure, properties and generative mechanisms, such that the three emerge simultaneously, and none is more primary than any of the others (Fleetwood, 2009).

3.7 Why Critical Realism for this Research

A number of IS scholars have argued for the use of critical realism as an underlying philosophy for IS research, citing the shortcomings of positivism which leans towards causality as a succession of events, eluding delineation of complex causation, and interpretivism which tends not to emphasize an independent enduring reality, but instead focuses on individuals' interpretations of digital technology at a moment in time (Mutch, 2002; Carlsson, 2004; Mingers, 2004a; Mingers, 2004b; Smith, 2006). These calls for critical realist IS research have been followed by several publications that explicate how the theoretical constructs of critical realism extend to IS contexts, and how they add explanatory power to studies in IS.

For example, Faulkner and Runde (2013), recognising the absence of digital technology in formative critical realist writings, used the TMSA (Collier, 1994, pp. 141-151; Engholm, 2007) to explain social positions occupied by non-human technological incumbents. Volkoff, et al. (2007), on the other hand, demonstrated how Archer's morphogenesis of structure (Archer, 1982; Archer, 1995) brings into view structural conditioning and structural elaboration in an IS context, to uncover an enduring material aspect of IS that is often unacknowledged in interpretivist IS research. Other researchers such as (Bygstad, 2010; Volkoff & Strong, 2013; Henfridsson & Bygstad, 2013) searched for generative mechanisms, to produce causal accounts that are more comprehensive than explanation formulated as the introduction of digital technology succeeded by causally attributed events. Still other IS scholars have sought to demonstrate that the somewhat abstract theoretical constructs of critical realism have concrete manifestations in IS contexts that can be distilled via procedures that maintain philosophical consistency with critical realism (Wynn & Williams, 2012; Volkoff & Strong, 2013; Bygstad, et al., 2016).

The critical realist philosophical perspective of these cited publications simultaneously accommodates the acknowledgement of an objective reality, and the place of social interaction and meaning making (Mutch, 2002; Carlsson, 2004; Mingers, 2004a; Mingers, 2004b; Smith, 2006; Carlsson, 2012), while transcending the aforementioned limitations of purely positivist or interpretivist philosophies. Similarly, this research used critical realism and its theoretical constructs to arrive at an answer to the research question.

This research's question necessitated a search for causal explanation of architectural evolution specifically in the context of sociotechnically ossified digital infrastructures. The vast sociotechnical ubiquity, and proliferation of networking infrastructures demanded causal explanation formulated for open environments. Additionally, an explanation of architectural evolution in this context could not be adequately captured as a succession of causally antecedent events, and resulting events. As this chapter has shown, critical realism provides theoretical constructs that are suitable for developing complex causal explanation in open environments.

Through positive network externalities (Katz & Shapiro, 1985; Katz & Shapiro, 1986; Katz & Shapiro, 1994; Shapiro & Varian, 1999, pp. 173-225), digital infrastructures condition human actions (Suarez, 2005; Zhu, et al., 2006; Edwards, et al., 2007; Hanseth & Lyytinen, 2010; Monteiro, et al., 2013), and by their objective material aspect, place limitations on the interpretive dimension of human interaction (Kallinikos, 2004; Volkoff, et al., 2007; Kallinikos, 2011b). Digital infrastructures being never designed *de novo* but instead being the outcome of reproduction, or extensions and changes to earlier infrastructures (Star & Ruhleder, 1996; Hanseth & Lyytinen, 2010; Tilson, et al., 2010b; Aanestad & Jensen, 2011), are characteristically diachronic and enduring relative to any contemporary individuals. Philosophical positions that encourage focus on contemporary moments of instantiation between individuals and technology, such as constructivist and its derivative approaches, under-emphasize these important aspects of digital infrastructures (Kallinikos, 2004; Volkoff, et al., 2007; Kallinikos, 2011b), making them less suitable for this research. Depending on the type of extensions and changes made, digital infrastructures are either reproduced or transformed. When core architecture is changed, digital infrastructures are transformed (Monteiro, 1998; Hanseth, 2001; Hanseth & Lundberg, 2001; Edwards, et al., 2007; Egyedi & Spirco, 2011; Grisot, et al., 2014).

Given the preceding, digital infrastructures can be considered to be a type of structure. Archer's morphogenetic approach, introduced earlier, explicitly argues for attention to the historicity and pre-existence of structure that makes structure analytically separate from the agency of contemporary individuals, and provides a framework for understanding structural conditioning, reproduction and elaboration. It is a critical realist approach that frames the analysis through which the answer to the research question was produced. The research therefore looked at the morphogenesis of digital infrastructures – specifically networking infrastructures – as sociotechnical structure, but because of the particulars of the case studied, the morphogenesis of a second social structure, features in the findings and analysis.

Finally, in a manner somewhat contradictory to the arguments that originally proposed critical realism as a suitable philosophy for IS research, some incongruences were discovered, which I explicate in the Discussion chapter.

4 Methodology

4.1 Introduction

This chapter details the research design strategy and methods used to find an answer to the research question. The chapter proceeds with an overview of the research design strategy, followed by sections on the particulars of data collection, data analysis, retroducing a generative mechanism, generalisability of theoretical contributions, and ethical considerations.

4.2 Case Study as the Research Design Strategy

4.2.1 Introducing the Research Design Strategy

The research question sought more than a literal listing of generative mechanisms. Critical realist explanation requires the exposition of generative mechanisms whose operation in either a closed or open environment produce or sustain an event or phenomenon (Collier, 1994, pp. 107-130,169-181; Bhaskar, 1998a; Sayer, 2000, pp. 10-17; Pratten, 2007; Psillos, 2007; Pinkstone & Hartwig, 2007; Hartwig, 2007, pp. 57-60). Further, the antecedent structural conditioning, and subsequent reproduction or elaboration of structure instantiated with generative mechanisms, must be elucidated in critical realist explanatory accounts (Archer, 1995, pp. 149-161,165-218; Archer, 1998). Understood then, through a critical realist theoretic lens, the research question sought an explanation of *how* architectural evolution occurs. That is, the research objective was to uncover how architectural evolution in sociotechnically ossified digital infrastructures occurs.

To answer the research question, I undertook a qualitative embedded case study (Eisenhardt, 1989; Mabry, 2008; Robson, 2011, pp. 135-142; Cohen, et al., 2011, pp. 289-302; Yin, 2014) of the advent of Software-Defined Networking (SDN) infrastructures. The purpose of the case study research design strategy (May & Perry, 2011, pp. 228-233; Cohen, et al., 2011, pp. 289-294; Yin, 2014, pp. 29-33) was to build an *explanatory theory* (Markus & Robey, 1988; Eisenhardt, 1989; Gregor, 2006) of architectural evolution in sociotechnically ossified digital infrastructures, that complements the extant IS theorising on architectural evolution in digital infrastructures. The advent of Software-Defined networking infrastructures was selected as the phenomenon under study, i.e. the *case* (May & Perry, 2011, pp. 228-233; Yin, 2014, pp. 31-32), whose investigation would provide insights on the theoretical construct (Yin, 2014, p. 34) of the research question.

4.2.2 Embedded Case Study Units of Analysis

Case study is a research design strategy useful for answering research questions of the form "how" and "why", in depth, about a contemporary phenomenon of interest that is not easily separable from its context (Yin, 2014, pp. 10-11,14-19,24). The phenomenon of interest is contemporary in that it is not confined to the *"'dead' past where no direct observations can be made and no people are alive to be interviewed"* (Yin, 2014, p. 24). Additionally, the phenomenon under study either exists within an open environment, or is itself intrinsically internally open (Collier, 1994, pp. 128-129; Archer, 1995, p. 166), denying the researcher complete control over it (Yin, 2014, pp. 14-17). The advent of Software-Defined networking infrastructures (henceforth referred to as "SDN infrastructures") features these characteristics as a phenomenon of study, and is a complex phenomenon. For this reason, its study required separate dedicated analysis of its technical and social aspects, in the pursuit of an answer to the research question. In accordance with these requirements, an *embedded* case study research design strategy (Cohen, et al., 2011, pp. 291-292; Yin, 2014, pp. 49-56) with two units of analysis, was created.

The first unit of analysis was of primarily physical and logical artefacts[14] (Seaman, 1999; Esterberg, 2002, pp. 117-121; Runeson & Höst, 2008; Yin, 2014, pp. 31-32,117-118). Specifically, this was the technical details of the architecture of SDN infrastructures. An artefactual boundary was defined to delineate what technical details were admitted into the study. I return to the case's boundary later. The second unit of

[14] I explain later why "logical" does not mean the complete absence of physicality.

analysis was of the social objectives of the advent of SDN infrastructures entrenched in the interests (Archer, 1995, pp. 203-205) of three major *types* of organisations, in the networking industry, that created networking technology, or owned networking infrastructures, and were involved in SDN technological innovation, and SDN infrastructure deployments. The purpose of this unit of analysis was to uncover why SDN infrastructures came about at a particular point in time, and the sociotechnical processes of how ossified networking infrastructures were being architecturally transformed into SDN infrastructures.

Though there were two units of analysis, the overall case of the advent of SDN infrastructures was kept focal during the conduct of the research so that it did not inadvertently become background context to either unit (Yin, 2014, pp. 55-56). The close relatedness of the units of analysis, and the complementary relationship of their findings, made achieving this less challenging. The two units of analysis required different sources of evidence in order to produce findings that answered the research question. Embedded case study and case studies in general, are a research design strategy that inherently incorporates the use of multiple sources of evidence to arrive at findings (Eisenhardt, 1989; Cohen, et al., 2011, pp. 299-300; May & Perry, 2011; Yin, 2014, pp. 16-17,105-123). Evidence from the various sources may be gathered using more than one method of data collection (Eisenhardt, 1989; Cohen, et al., 2011, pp. 296-300; May & Perry, 2011; Yin, 2014, pp. 16-17,105-123). In the data analysis section, I describe how I employed data and methodological triangulation techniques, to determine the synergy and validity of findings contributed by the different sources of evidence, and methods of data collection (Mabry, 2008; Cohen, et al., 2011, pp. 195-197,295-296,299-300; Flick, 2014a, pp. 182-192; Yin, 2014, pp. 119-122).

4.2.3 The Advent of SDN Infrastructures as a Revelatory Case

The advent of SDN infrastructures, followed the advent of SDN *as an innovation*. I included the word “advent” in naming the case, because in the context of sociotechnical ossification, architectural evolution is more than the creation of a technological innovation. A technological innovation alone does not surmount sociotechnical ossification of pre-existing digital infrastructures (Monteiro, 1998; Hanseth & Braa, 2000; Egyedi & Spirco, 2011; Sanner, et al., 2014; Grisot, et al., 2014).

Therefore, the focus of the case study was on how, from an architectural perspective, *production* SDN infrastructures came about in spite of conditions of sociotechnical ossification. Here, the definition of production networking infrastructures, is networking infrastructures that are used by organisations to carry out their commercial activities. Production networking infrastructures are placed in contradistinction to networking infrastructures used only for testing and experimentation. Notwithstanding, it was not possible to answer the research question in the absence of an understanding of SDN as an innovation.

The advent of SDN infrastructures occurred within the context of sociotechnical ossification of pre-existing networking infrastructures. Sociotechnical ossification of networking infrastructures is not local to any singular organisation (Hanseth, et al., 1996; Monteiro, 1998; Edwards, et al., 2007; Hanseth & Lyytinen, 2010). Consequently, the case study considered the advent of SDN infrastructures from the perspective of what was occurring across the networking industry. As stated already, in the networking industry, three major types of organisations that created networking technology or owned networking infrastructures, were involved in SDN innovation, or in deploying SDN infrastructures, and possessed significant powers[15] for changing the architecture of networking infrastructure:

1. ***Network Operators*** are one of the three types of organisations. For clarity, network operator is defined in this document as any owner of large networking infrastructure. Relevant examples of network operators are ISPs, cloud service providers, and other organisations with large internal networking infrastructures.
2. ***Network Equipment Vendors*** are a second major type of networking technology organisation. Network equipment vendors are manufacturers of network hardware devices. Cisco and Juniper, for example, are network equipment vendor organisations.

---

[15] Here, "power" is used in the critical realist sense (Collier, 1994, pp. 61-69; Bhaskar, 1998a; Sayer, 2000, pp. 10-17; Psillos, 2007; Hartwig, 2007, pp. 57-60; Fleetwood, 2009; Fleetwood, 2011).

3. ***Network Systems Integrators*** are the third major type of networking technology organisation. The primary business of network systems integrators is to build and integrate network solutions into disparate network operators' networking infrastructures.

As a construct, in this research the networking industry was defined as a social structure that is emergent from necessary internal relations of interdependence between the aforementioned three types of organisations. The necessary internal relations are:

1. Network operators' dependence on network equipment vendors for proprietary networking products and services for use within their networking infrastructures,
2. Network equipment vendors' dependence on network operators for the continued purchase of their networking products and services,
3. Network operators' dependence on network systems integrators for network solutions implementation and networking infrastructure management services,
4. Network systems integrators' dependence on network operators for the continued purchase of their products and services,
5. Network systems integrators' dependence on network equipment vendors to implement network solutions in network operators' networking infrastructures featuring the network equipment vendors' products and services,
6. Network equipment vendors' dependence on network systems integrators for continued implementation of network solutions in network operators' networking infrastructures that include their proprietary products and services.

The lack of a geographical boundary for the network industry construct is possibly a limitation of this research, but the social structure with these necessary internal relations of interdependence and the relata of network operators, network equipment vendors, and network systems integrators, were found to be consistently manifested across the geographic locations of interviewees.

There is a fourth type of organisation called a **Standards Development Organisation** (SDO). SDOs cultivated the installed base (Monteiro, 1998; Hanseth & Lyytinen, 2010;

Aanestad & Jensen, 2011; Sanner, et al., 2014; Grisot, et al., 2014) of network operators, network systems integrators, and network equipment vendors towards deploying SDN infrastructures. The focus of this research was not on particular standardisation activities, or on the cultivation of the installed base, and as well, data access restrictions prevented detailed analysis of the role of SDOs in the advent of SDN infrastructures. Thus, the role of SDOs features only in the background of the case study, but that did not hinder the identification of generative mechanisms that promote architectural evolution in sociotechnically ossified digital infrastructures.

The case of the advent of SDN infrastructures is summarised in the following. Network operators wanted to architecturally evolve their existing networking infrastructures to circumvent complications and limitations that resisted network services innovation, in the midst of increasing capital and operational expenditures. They saw the technological innovation of SDN as facilitative of this. Network equipment vendors, on the other hand, had benefited from the maintenance of a de facto standard implementation architecture in network hardware devices that resisted what network operators wished to have. So there was a socialisation of the term "SDN" to mean more than the technological innovation. SDN connotes a *strategy*, primarily by network operators and by SDOs, that reframes SDN as a sociotechnical methodology by which an ossified implementation architecture around which the networking industry is arranged, is evolved. The technical means by which the implementation architecture is evolved yields, in parallel, *occasion* for core architectural evolution, (i.e., of the Internet's core architecture).

SDN infrastructures feature architecture that is in some ways fundamentally different from that instantiated in predecessor networking infrastructures (henceforth for clarity preceded by the adjective "traditional"). Extant IS theorising on digital infrastructures suggests that fundamental changes to architecture demands replacement of the instantiating digital infrastructures, but traditional networking infrastructures already exist, are widely propagated, and are highly sociotechnically ossified, prohibiting their complete replacement to accommodate SDN's architectural differences. In spite of the extant IS theorising which suggests digital infrastructure replacement, SDN infrastructures exploit the existing digital materiality and upward flexibility of traditional networking infrastructures to evolve the underlying network architecture. The advent of SDN infrastructures, therefore, features *revelatory* characteristics (Mabry, 2008; May & Perry, 2011, pp. 228-233; Yin, 2014, p. 52) as a case study for understanding how the architecture of sociotechnically ossified traditional networking infrastructures evolved towards the architecture of SDN infrastructures.

4.2.4 Boundaries of the Case

Identification of social and technical enabling, stimulating and releasing conditions (Pinkstone & Hartwig, 2007, p. 458; Hartwig, 2007, pp. 57-60,80; Fleetwood, 2011) of generative mechanisms that promote architectural evolution in sociotechnically ossified traditional networking infrastructures towards SDN infrastructures, along with an analytic history (Archer, 1995, pp. 149-161,165-218; Archer, 1998) are necessary explanatory components that must fall within the case study. The word "advent" in the case's name is additionally meant to explicitly indicate the admission of broader sociotechnical aspects into the case study. The case, nonetheless, did not extend endlessly into the sociotechnical aspects of the advent of SDN infrastructures. It was delimited by a defined finite multi-part case boundary that demarcated the case from its surrounding context, and background (Yin, 2014, pp. 33-34).

Temporally, empirical evidence on *sociotechnical ossification* of networking infrastructure, directly causally attributed to the advent of SDN infrastructures by its earliest innovators and by this research's interviewees, was temporally bounded from March 2002 to September 2016 to be within the case study. I provide some additional commentary on events occurring from January 2016 onwards in the data analysis section. The case also included empirical evidence on predecessor *technological innovations* directly related to the networking innovation that was eventually referred to as SDN starting at around June 2007. This empirical evidence on the technological innovation was temporally bounded (Yin, 2014, pp. 33-34) from November 2004 to December 2015.

The case had a concrete artefactual boundary (Yin, 2014, pp. 33-34,117-118). As typical of qualitative case study, over its duration the research reported in this doctoral thesis underwent changes that served to focus the case (Mabry, 2008, p. 216; Robson, 2011, pp. 130-142; May & Perry, 2011, p. 230). An initial focus on infrastructure virtualisation[16], in which the digital materiality of key digital infrastructure components including the network are exploited, yielded a seemingly endless and ambiguous case boundary. In response, I refined the case boundary by limiting artefactual case evidence specifically to the distinctive technological innovations of SDN (these are introduced in Chapter 5). For exactness, a closely related and complementary technological innovation called Network Functions Virtualisation (NFV), which furthers SDN's architectural evolution in sociotechnically ossified networking infrastructures, was explicitly excluded from being within this case's boundary. This thesis reports on the final empirical object – the case of the advent of SDN infrastructures – hence, this chapter details the research design strategy surrounding this final empirical object with the aforementioned artefactual boundary.

Only the social objectives of the advent of SDN infrastructures entrenched in the interests (Archer, 1995, pp. 203-205) of network operators, network equipment vendors, and network systems integrators were included within the case's boundary. Several interviewees were simultaneously senior members of technical committees at SDOs, and of one of these three types of organisations. Their insights on the technical details of SDN infrastructures, and the perspectives of the organisational types based on their experience, were included within the case boundary, but specifics of the processes employed by SDOs to cultivate the installed base (Monteiro, 1998; Hanseth & Lyytinen, 2010; Aanestad & Jensen, 2011; Sanner, et al., 2014; Grisot, et al., 2014) were treated as background context of the case. Many details of actions, responses, and tussles between network operators, SDOs, network systems integrators and network vendors, were encountered during this research, but their specifics were not pertinent to answering the research question. These details were treated as background context residing outside of the case boundary.

---

[16] Virtualisation re-emerges in the findings, and therefore is explained in Chapter 5.

4.2.5 Critical Realism and the Research Design Strategy

The use of critical realism as the philosophical basis of this research carries a number of practical implications for the research design strategy. Critical realist research aims to *retroduce* (Collier, 1994, p. 22; Reichertz, 2014; Kelle, 2014, pp. 561-562) why or how some event or phenomenon that has occurred or exists, was brought into being (Collier, 1994, pp. 22,160-167; Mingers, 2004a; Mingers, 2004b; Easton, 2010; Wynn & Williams, 2012). The objective is to demonstrate the reality of specific generative mechanisms, alongside real structures and properties with which they are instantiated, whose operation are causal to the existence of some event or phenomenon (Collier, 1994, pp. 22,160-167; Mingers, 2004a; Mingers, 2004b; Easton, 2010; Wynn & Williams, 2012; Reichertz, 2014). As such, critical realist research questions are always oriented toward finding causal explanation of how or why an event or phenomenon was brought into being (Easton, 2010; Wynn & Williams, 2012).

Causal explanation in critical realist research is not of hypotheticals, but of temporally situated real events and phenomena existing in the domains of the actual or empirical (Archer, 1995, pp. 135-161,165-194; Archer, 1998; Wynn & Williams, 2012). Finding a causal explanation of an event or phenomenon implies that it already occurred or exists, or has started to occur and continues to exist in the present as a contemporary phenomenon (Easton, 2010; Wynn & Williams, 2012). This does not create a conflict between the transfactuality of generative mechanisms (Collier, 1994, pp. 61-69; Bhaskar, 1998a; Psillos, 2007; Pinkstone & Hartwig, 2007; Hartwig, 2007, pp. 85-87; Fleetwood, 2009; Fleetwood, 2011), and the practical conduct of critical realist research. Rather, it differentiates between what is the focal phenomenon of study (Easton, 2010; Wynn & Williams, 2012), and what are *other* events – whether actualised or not.

These implications of critical realism for research questions and for phenomena of study, are harmonious with case study research design strategy (Sayer, 2000, pp. 19-22; Easton, 2010; Wynn & Williams, 2012), which is suited for answering research questions of the form "how" and "why", in depth, about concrete contemporary phenomena of interest (Yin, 2014, pp. 10-11,14-19,24). In section 4.2.1, I explained that this research's question reflects a critical realist philosophical basis, in its form, and in its requirements for an answer that features an in depth exposition of causality. The case of the advent of SDN infrastructures, was the focal phenomenon of study. It pre-existed the commencement of the research, but continued into the present, relative to the conduct of the research investigation (See also (Yin, 2014, p. 24)). Thus, as per principles of case study and critical realism, the research design strategy for answering the research question necessarily sought causal explanation of the advent of SDN infrastructures retrospectively (Wynn & Williams, 2012).

The process that guides critical realist research in the search of causal explanation, is a reasoning strategy called *retroduction* (Collier, 1994, pp. 22,160-167; Mingers, 2004a; Mingers, 2004b; Easton, 2010; Wynn & Williams, 2012; Reichertz, 2014; Kelle, 2014, pp. 561-562). Given a research question, the retroductive process starts by considering an empirically observed event or phenomenon, and then attempting to explain it using extant theorisation (Reichertz, 2014; Kelle, 2014, pp. 561-562), such as attribution to the operation of previously identified generative mechanisms (Wynn & Williams, 2012). If the use of extant theorisation is deemed, by the researcher, to provide an inadequate explanation, the researcher then postulates a new explanation based on other pre-existing theoretical explanations, empirical data, and newly formed knowledge (Reichertz, 2014; Kelle, 2014, pp. 561-562).

For critical realist research, this involves postulating the existence and operation of generative mechanisms instantiated with structures and properties (Collier, 1994, pp. 160-168; Mingers, 2004a; Mingers, 2004b; Easton, 2010; Wynn & Williams, 2012). The researcher then proceeds as though the new explanation is true, but continues gathering further empirical evidence, while constantly questioning what would account for the way that things are (Easton, 2010; Wynn & Williams, 2012; Reichertz, 2014; Kelle, 2014, pp. 561-562). Whenever further investigation renders a postulated explanation inadequate, the explanation is either refined or replaced on the basis of other available theoretical explanations, empirical data, and newly formed knowledge (Reichertz, 2014; Kelle, 2014, pp. 561-562). The process continues in this manner until a sufficiently adequate and acceptable explanation is reached (Collier, 1994, pp. 160-168; Mingers, 2004a; Mingers, 2004b; Easton, 2010; Wynn & Williams, 2012; Reichertz, 2014; Kelle, 2014, pp. 561-562).

The important question for retroduction is, what constitutes a sufficiently adequate and acceptable explanation? Retroduction is a reasoning strategy (Reichertz, 2014), and hence by critical realism, it is subject to limitations of the transitivity of knowledge (Collier, 1994, p. 163; Mingers, 2004b, pp. 385,390; Easton, 2010; Wynn & Williams, 2012), although it seeks to establish *as ontological*, the reality of generative mechanisms, structures and properties in the domain of the real (Collier, 1994, pp. 22,161-167; Mingers, 2004a; Mingers, 2004b; Easton, 2010; Wynn & Williams, 2012). Retroduction has been described as a process of *informed guessing*, because it relies on continuously postulating explanations on the basis of pre-existing theories, and on acquired knowledge (Reichertz, 2014). It is possible then that causal attribution could be made to the wrong generative mechanism (Sayer, 2000, pp. 16-17), or to inconsequential or non-existent generative mechanisms (Mingers, 2004b, p. 390; Easton, 2010; Wynn & Williams, 2012).

Critical realist research, therefore must demonstrate that proposed explanations offer more than alternative explanations (Sayer, 2000, pp. 13-17; Easton, 2010; Wynn & Williams, 2012; Reichertz, 2014; Kelle, 2014, pp. 561-562), and must demonstrate the reality of the operation of generative mechanisms causal to events and phenomena of study, explicating the structures and properties that are *necessarily* instantiated with them (Archer, 1995, pp. 135-161,165-194; Archer, 2000; Sayer, 2000, p. 14; Wynn & Williams, 2012). The objective must be to demonstrate that the events or phenomena of study could not have come into being, or continue to exist, in the absence of proposed generative mechanisms, structures, and properties as intransitives in the domain of the real (Wynn & Williams, 2012).

In this research, the process of retroduction began with an observation that cloud computing infrastructure virtualisation seemed to violate key principles of modularity theory (Parnas, 1972; Sanchez & Mahoney, 1996; Schilling, 2000; Yoo, et al., 2010). As already mentioned, being formulated around infrastructure virtualisation, the case had an ambiguous boundary. After refining the case (Mabry, 2008, p. 216; Robson, 2011, pp. 130-142; May & Perry, 2011, p. 230) to the advent of SDN infrastructures, I determined that extant IS theorising on architectural evolution in sociotechnically ossified digital infrastructures, which is largely premised on modularity theory, could not explain adequately, the architectural evolution that led to the advent of SDN infrastructures. It is from this point of surprise (Reichertz, 2014, p. 126), that I began the process of retroducing an explanation of the advent of SDN infrastructures. Wynn and Williams (2012) suggested five principles for conducting critical realist research: *explication of events*, *explication of structure and context*, *retroduction*, *empirical corroboration*, *triangulation and multi-methods*. This chapter has introduced the case and will explain how empirical corroboration and triangulation was carried out. Chapter 5 explicates events, structure and context of the advent of SDN infrastructures. The Analysis chapter, presents arguments for the reality of generative mechanisms, sought by the research question, that were identified through a process of retroduction.

Critical realism brings evaluative implications for research design strategy, but still there is debate about how "*critical*", critical realism is, and about what it is being critical (Mingers, 2004b). The particulars of the philosophical debate are beyond the scope of this research. Nonetheless, Collier explains that the more precisely reflective of *the real* an explanation is presented, *"evaluative force arises entirely out of the factual content"* (Collier, 1994, p. 178). As the data analysis of this research proceeded, it became clear that the advent of SDN infrastructures was not a purely technical outcome of continued networking innovation. Rather, it was the result of a purposeful strategy for instrumenting a broad change of the networking industry. Similar technological innovations to SDN pre-existed it, but SDN was purposely appropriated from academic research by network operators and SDOs to accomplish a particular goal. Though factual, this finding carries evaluative force. Once recognised, through the process of retroduction, the explanation of architectural evolution from traditional networking infrastructures to SDN infrastructures was refined to account more precisely for how sociotechnical ossification was circumvented.

4.2.6 Some Additional Comments

SDN as an innovation has matured significantly for deployment in production networking infrastructures, since the beginning of this research undertaking, until the creation of this thesis. In the interim, a number of computer science articles that survey SDN as an innovation, have been published in conferences, and in peer-reviewed academic journals. As well, SDOs, network operators, and network equipment vendors have published standards, reference architectures, and details of deployed SDN infrastructures, some of which have been co-authored by interviewees that participated in this research. I have dealt with this in two ways.

Documents published by SDOs, have been used to extend my technical knowledge of SDN as an innovation, and of related or similar technologies. Journal articles provided technical overviews of SDN as an innovation, along with partial historical recollections of related, and predecessor networking technologies. So there is some overlap between the data that I have collected and these articles, but there are also significant differences.

Surveys in journal articles have been largely composed on the basis of experimental technologies reported in computer science conference papers. The majority of those technologies are not deployed in any network operator's production networking infrastructure. In carrying out my research, however, I took care to either specifically ask interviewees whether what they discussed was based on experience, or alternatively I independently verified interviewee's statements through the use of other sources (such as network operator, or network equipment vendor published documents and press releases). Therefore, the scope of what is included in this research is narrower in some ways than what is in those articles, but its deliberate exclusion of extraneous information and experiments that are not deployed in production networking infrastructure, safeguards *validity* in the research (Mabry, 2008, pp. 221-223; Robson, 2011, pp. 154-159; Cohen, et al., 2011, pp. 180-200; Flick, 2014a, pp. 483-486; Yin, 2014, pp. 45-49).

4.3 Data Collection

4.3.1 Introduction

The choice of data collection methods was guided by a question of what is necessary to be known about the advent of SDN infrastructures, in order to answer the research question (Richards, 2005, p. 41; Green, 2008, pp. 59-60; Robson, 2011, pp. 230-233,407-408). From a critical realist perspective, the methods needed to be appropriate for pursuing the type of evidence necessary to establish, through retroduction, the reality of the generative mechanisms sought by the research question (Easton, 2010). Two methods of data collection were used in this embedded case study. These were the use of documents (Esterberg, 2002, pp. 121-128; Wolff, 2004; Macdonald, 2008; McCulloch, 2011; Flick, 2014a, pp. 298-299,352-364), and qualitative interviewing (Gaskell, 2000; Esterberg, 2002, pp. 83-114; Kvale, 2007; Fielding & Thomas, 2008; Cohen, et al., 2011, pp. 409-443; Robson, 2011, pp. 278-301; Flick, 2014a, pp. 207-241).

The first unit of analysis was of primarily physical and logical artefacts of SDN infrastructures (Seaman, 1999; Esterberg, 2002, pp. 117-121; Runeson & Höst, 2008; Yin, 2014, pp. 31-32,117-118), but these artefacts could not be accessed directly. Consequently, documents were used primarily – though not exclusively – to collect data about the technical details of the architecture of SDN infrastructures – the first unit of analysis. As well, the use of documents provided data to be analysed for why SDN infrastructures came about at a particular point in time, and the sociotechnical processes of how existing ossified networking infrastructures were being architecturally transformed into SDN infrastructures, i.e., the second unit of analysis. Qualitative interviewing was used primarily to collect data for the second unit of analysis, and it was used to gather additional and complementary data to documents about the technical details of the architecture of SDN infrastructures based on interviewees' experience with various production SDN infrastructures. Data collection for both units of analysis was concluded upon theoretical saturation (Eisenhardt, 1989).

The major data collection activities of the research occurred between August 2012 and December 2015. However, cycles of data analysis, and additional monitoring and collection of *documents* continued until September 2016. I explain the extended collection of documents later. The remainder of this section provides the rationale and details of how and which data was selected and collected via the use of documents, and qualitative interviewing for each unit of analysis.

4.3.2 Description of the Data Collection Strategy for the First Unit of Analysis

The purpose of the first unit of analysis was to go beyond explanation that characterises the architectural evolution in sociotechnically ossified traditional networks towards SDN infrastructures primarily in terms of abstract social actions taken by organisations. The architecture of digital infrastructures such as networking infrastructure, is very much a technical issue. It was necessary then to understand, with specificity, the technical details of the architectural changes made during a traditional network's architectural evolution towards SDN infrastructure, and not only the high-level social actions of organisations. A variety of technical documents published in electronic form online, either as independent items, such as Portable Document Format (PDF) documents, or as pages on websites, contributed the majority of these technical details. Data selected for analysis was either downloaded as documents, or URLs recorded if the data was only published on a website. Qualitative interviews with networking experts provided additional technical details, and guided the collection of document data. I present details of the qualitative interviews later in the discussion of how data was collected for the second unit of analysis.

A purposive sampling logic (Richards, 2005, pp. 37,41; Mabry, 2008, p. 223; Cohen, et al., 2011, pp. 156-158; Flick, 2014a, pp. 173-181; Rapley, 2014; Barbour, 2014, pp. 497-498) guided data collection of technical details of the architecture of SDN infrastructures. Data collection of the technical details of the architecture of SDN infrastructures proceeded in an intermingled cycle of collection and analysis (Eisenhardt, 1989; Gibbs, 2007, pp. 1-9; Flick, 2014a, p. 142; Flick, 2014b, pp. 9-10). Technical documents were collected, and analysed for architectural details of SDN infrastructures. This was followed by the collection of documents about aspects of the architecture of SDN infrastructures identified in the preceding analysis, for which either technical data had not already been collected, or had not sufficiently expounded the technical details of interest.

The inter-textuality of documents collected (Wolff, 2004; Flick, 2014a, pp. 356-358,360-361; Coffey, 2014, pp. 373-375), though not an independent focus of the analysis, provided a guide for the continued selection of related documents. The intended audience (Wolff, 2004; Macdonald, 2008, pp. 294-296; Cohen, et al., 2011, p. 253; Flick, 2014a, pp. 356-358; Coffey, 2014) of most documents was individuals that held a technical understanding of existing traditional networks and of computing. Implicit and explicit references (Wolff, 2004; Macdonald, 2008, pp. 294-296; Flick, 2014a, pp. 356-358; Coffey, 2014) to networking technologies, network architecture and topologies, protocols, and products were used to identify pertinent data to be collected. Documents and websites were located primarily using Google's online search engine. Qualitative interviews with networking and infrastructure experts involved in the technical definition of SDN as a networking innovation, and in the deployment of production SDN infrastructure, were used partly to gather details of the architecture of SDN infrastructures. Routinely, interviewees discussed technical aspects of the architecture of SDN infrastructures that I had not previously encountered. Whenever this occurred, technical documents that provided elaborated explanations of these aspects were gathered, and analysed.

Given the highly technical nature of SDN infrastructures, during data collection I also relied on my academic and professional background in computer science and software engineering[17], to help me discern what technical data should be pursued and collected for analysis, and to refine interview questioning in ways that aimed to clarify unclear technical details of the architecture of SDN infrastructures. Having stated this, I still undertook a separate study of the subject of networking, in order to improve my

[17]As a summary, this is in reference to a Bachelor of Science in Computing, Emphasis in Computer Science, a Master of Science in Advanced Software Engineering, professional certifications, and several years working professionally.

technical knowledge, and hence my ability to make informed selection of technical data, and to be able to competently manage interview data collection with expert interviewees (Richards, 2005, pp. 24-25; Kvale, 2007, pp. 37-40; Yin, 2014, pp. 75-76).

The methodological positioning of documents of the first unit of analysis, had implications for the data collection strategy. Documents were not selected solely to corroborate interviewee statements, or as definitive accounts of the architecture of SDN infrastructures (Wolff, 2004, p. 288). As far as possible, they were treated as independent data, contributing their own findings, in parallel with findings contributed by qualitative interviews (Wolff, 2004, p. 288). Still, the content of some types of documents, such as technical specifications, had to be treated as describing intransitives of technology (that is part of SDN infrastructures), and therefore were used to corroborate and to triangulate data collected via interviewing (Mabry, 2008; Cohen, et al., 2011, pp. 195-197; Flick, 2014a, pp. 182-192; Yin, 2014, pp. 107,119-122). Because the aim of the first unit of analysis was to uncover the technical details of the architecture of SDN infrastructures, documents were selected at the level of the *content* of the document, as opposed to on their communicative features (Wolff, 2004; Macdonald, 2008; Runeson & Höst, 2008; Flick, 2014a, pp. 298-299,352-364; Coffey, 2014). But communicative features of documents were not ignored. They were used to make sense of the type of document, the implications of who its publisher was and its original purpose for being created on what it communicated (or did not communicate), and how it presented particular architectural details of SDN infrastructures (Wolff, 2004; Macdonald, 2008; Runeson & Höst, 2008; Flick, 2014a, pp. 298-299,352-364; Coffey, 2014).

As stated previously, qualitative interviewing was used to gather additional and complementary data to documents about the technical details of the architecture of SDN infrastructures based on interviewees' experience with various production SDN infrastructures. However, to avoid repetition, the following sections focus mainly on the use of documents for the first unit of analysis, and instead the details of how qualitative interviewing was carried out is presented within the explanation of data collection for the second unit of analysis.

4.3.3 Details of the Data Collected for the First Unit of Analysis

Documentation and archival data (Esterberg, 2002, pp. 121-134; Runeson & Höst, 2008; Yin, 2014, pp. 105-110) are the two general types of documents that provided data for the first unit of analysis. Documentation was collected to establish the architecture of SDN infrastructures, while archival data were collected for the architectural history of what eventually became known as SDN and later deployed as production SDN infrastructures by network operators.

Documentation used consisted of technical specifications, technical recommendations, technical references, technical whitepapers, requests for comments (RFCs), network equipment vendors' and network systems integrators' product data sheets and SDN infrastructure solutions briefs, SDN reference architectures, online blog posts and press releases explaining architectural details of network operators' SDN infrastructures, and press releases introducing network equipment vendors' and network operators' SDN-based products and services. Importantly, press releases were monitored and used to confirm information and to locate other documents, but they were not all collected. Having stated this, the Findings and Analysis chapters reference a number of sources that are press releases. Though some documents were discovered through references in blogs, blog aggregators, networking websites, other documents or by interviewee statements, to ensure reliability and credibility (Richards, 2005, pp. 43-44; Cohen, et al., 2011, pp. 201-204; Robson, 2011, pp. 93,155,159; Yin, 2014, pp. 48-49), analysed documents were retrieved solely from their originating publisher. The main types of publishers were SDOs, network equipment vendors, network systems integrators and network operators.

Specifically, SDN-related publications from the following SDOs were monitored and collected: Open Networking Foundation (ONF), Internet Engineering Task Force (IETF), Institute of Electrical and Electronics Engineers SDN Initiative (IEEE SDN), Telecommunication Standardization Sector of the International Telecommunication Union (ITU-T), and Metro Ethernet Forum (MEF). I reiterate these in Chapter 5. To be clear, these documents were not collected for insights on the role of SDOs in cultivating the installed base of network operators, network systems integrators and network equipment vendors towards SDN infrastructures. They were collected for architectural details of SDN infrastructures. Although as stated in section 4.2.4, the closely related network functions virtualisation, i.e. NFV, was designated as being outside of the case's boundary, I still collected various publications from the European Telecommunications Standards Institute's Network Functions Virtualisation Industry Specification Group (ETSI NFV ISG). The reason for doing so, was to ensure that I understood clearly the architectural boundary between SDN and NFV, since many network operators – particularly ISPs – were implementing both SDN and NFV (SDN+NFV infrastructure) for their networking infrastructures.

Very importantly, depending on the type of documentation, what SDOs publish may be informational, or of theoretical technical *ideals*. This posed a notable threat to internal validity and to construct validity (Cohen, et al., 2011, pp. 180-181,184-185,188-189; Yin, 2014, pp. 45-49). To contend with this, press releases, and the public offering of SDN-based products and services from network operators were monitored to determine what was being deployed in production SDN infrastructures by network operators, as opposed to what was only being postulated as theoretical technical ideals. Similarly, press releases and information about SDN products from network equipment vendors, and network systems integrators disclosed as being deployed in production SDN infrastructures were included in data monitored and collected. Additionally, interviewees were asked directly about deployment of SDN infrastructures conformant to particular architectural details presented in SDOs' publications, and to respond based on their past experience – a point to which I return later. Any technical details of the architecture of SDN infrastructures that could not be confirmed as being deployed in production SDN infrastructures was explicitly excluded, removed, or discounted from data collected. This exclusion applied to architectural details of experimental and test deployments of SDN infrastructures by network operators.

To manage data collection within the constraints of PhD research, I focused on specific network systems integrators, and network equipment vendors that were heavily involved in creating SDN products and building SDN infrastructures. Importantly, many network systems integrators also manufacture network hardware devices. Specific network equipment vendors and network systems integrators whose SDN-related products, press releases and other publications were studied, monitored and collected were: Cisco Systems, Juniper Networks, Ericsson, Hewlett-Packard, Brocade, Big Switch, Arista Networks, and IBM. Similarly, I focussed on network operators that were most open with the public about their SDN infrastructure implementations. These included Google, Microsoft (Azure Cloud), AT&T, Verizon, Orange, BT, and NTT Communications. For a short period, IBM (as a network operator), Amazon AWS, Rackspace, Deutsche Telekom, and China Mobile, were monitored for publications. Other organisations, not categorised as SDOs, network equipment vendors or network systems integrators, whose press releases and other publications were monitored and collected, because their SDN-related and other products were routinely used in SDN infrastructures were: Nicira Networks, VMware, and Citrix.

Over the duration of this research, I also read blog aggregator websites that reported on SDN, in order to remain aware of the latest architectural developments in production SDN infrastructures, and I pursued technical details at the original publishers. The website, SDxCentral.com (formerly SDNCentral) was the main resource used for this purpose, supplemented by NetworkWorld.com, Lightreading.com and Cio.com. These websites reported SDN-related developments published by diverse network equipment vendors, network systems integrators, network operators and networking experts beyond the aforementioned limited list of network equipment vendors, systems integrators and network operators. In a similar manner to blog aggregator websites, technical presentations on YouTube about implementing aspects of SDN infrastructures, were used to gain clarification on how certain architectural features of SDN infrastructures are practically implemented.

Archival data consisted of archived SDN research project websites, early presentations, computer science conference papers, and peer-reviewed articles that explained predecessor SDN technologies and architecture. Specifically, archival data which provided an architectural history of SDN as an innovation was collected from the SANE, Ethane, Clean Slate Program (Stanford University), FlowVisor, NOX, and OpenFlow projects, and from the OpenFlow Consortium. The PhD thesis of Martín Casado (a pioneer of SDN) was included in the archival data.

As the first unit of analysis was of technical details of the architecture of SDN infrastructures, documentation and archival data was collected about particular areas of SDN infrastructure architecture. Documentation and archival data was purposively sought and selected (Richards, 2005, pp. 37,41; Mabry, 2008, p. 223; Cohen, et al., 2011, pp. 156-158; Flick, 2014a, pp. 173-181; Rapley, 2014; Barbour, 2014, pp. 497-498) based on areas of the architecture of SDN infrastructures about which the data analysis revealed that greater understanding was required.

Though for an information systems PhD thesis these might be considered to be mundane details, for the purpose of reliability and credibility (Richards, 2005, pp. 43-44; Cohen, et al., 2011, pp. 201-204; Robson, 2011, pp. 93,155,159; Yin, 2014, pp. 48-49), key areas are listed next, and the core architecture of SDN infrastructures is later introduced in Chapter 5. Appendix D additionally contains a sample of technical data used for the first unit of analysis.

Areas of the architecture of SDN infrastructures about which data was collected included: software-defined network deployment topologies, SDN controllers deployed in production, SDN southbound interfacing, SDN northbound interfacing, SDN southbound management interfacing, SDN controller southbound interfacing support, commercial hardware router SDN integration and southbound API support, production-deployed software-based virtual switches, and network virtualisation technologies and techniques. Aside from these, I studied related technologies in order to gain an understanding of how SDN infrastructures are situated within the overall digital infrastructures of network operators. Appendix D includes a sample of these related technical data.

Finally, I attended two conferences and held informal conversations with presenters and attendees that helped to secure formal interviews, and to clarify architectural details of SDN infrastructures. The conferences were: the 1st IEEE Conference on Network Softwarization, held April 13th - 17th, 2015 and CloudCamp London, held November 6th, 2013.

4.3.4 Data Collection for the Second Unit of Analysis

As already introduced, qualitative interviewing (Gaskell, 2000; Esterberg, 2002, pp. 83-114; Kvale, 2007; Fielding & Thomas, 2008; Cohen, et al., 2011, pp. 409-443; Robson, 2011, pp. 278-301; Flick, 2014a, pp. 207-241) was used primarily to collect data about why SDN infrastructures came about at a particular point in time and the sociotechnical processes of how ossified networking infrastructures were being architecturally transformed into SDN infrastructures, i.e. the second unit of analysis, and it was used to gather additional and complementary data to documents about the technical details of the architecture of SDN infrastructures based on interviewees' experience with various production SDN infrastructures. Technical documentation and archival data (Esterberg, 2002, pp. 121-134; Runeson & Höst, 2008; Yin, 2014, pp. 105-110) in the form of press releases, from network operators, network equipment vendors (henceforth, interchangeably referred to as "vendors"), and network systems integrators (henceforth, interchangeably referred to as "systems integrators") were monitored for corroborating interviewee statements that contributed to the second unit of analysis (Mabry, 2008; Cohen, et al., 2011, pp. 195-197,295-296,299-300; Flick, 2014a, pp. 182-192; Yin, 2014, pp. 107,119-122).

Production SDN infrastructure deployments had been carried out mostly by operators of large networking infrastructures. Therefore, it was necessary to interview persons with corresponding experience. Thirty-nine expert and elite interviews (Kvale, 2007, p. 70; Flick, 2014a, pp. 227-232) with persons from telecommunications and cloud data centre network operators – including from Tier 1 ISPs – as well as from vendors, and systems integrators, were conducted between July 2013 and December 2015. As presented in Table 4-1, several interviewees held senior management roles at their organisations, with some simultaneously occupying leading SDN-related roles at SDOs. Some individuals, at the time of their interview, were not working at network operators, vendors or systems integrators, neither were they in senior management, but all of these, with the exception of one person, held experience working with network operators. Of the thirty-nine interviews, one person was determined not to be an expert in networking or generally in digital infrastructures (Flick, 2014a, p. 229). This person's responses were set aside from the data collected for analysis. Problems with the recording quality of two other interviews meant that they could not be transcribed for use in the data analysis.

Interviewees were contacted by email, or via the LinkedIn professional network. Research access request letters that described the research undertaking were provided to potential interviewees. The details of gaining research access to interviewees are detailed in section 4.7's explanation of the ethical considerations of this research.

The final composition of interviewees was the outcome of a combination of purposive sampling (Richards, 2005, pp. 37,41; Mabry, 2008, p. 223; Cohen, et al., 2011, pp. 156-158; Flick, 2014a, pp. 173-181; Rapley, 2014; Barbour, 2014, pp. 497-498) and snowball sampling strategies (Esterberg, 2002, pp. 93-94; Cohen, et al., 2011, pp. 158-160). The purposive sampling strategy involved selecting individuals in positions of authority and influence over the architectural direction for SDN being promoted by SDOs to network operators. The individuals in these roles held considerable and broad insight into why and how network operators were architecturally evolving their networks towards SDN infrastructures. Similarly, senior managers and networking experts at systems integrators and vendors which were involved in architecturally evolving network operators' infrastructures towards SDN infrastructures, were selected because of their experience and understanding of heterogeneous network operators' networking infrastructures. Senior managers and networking experts at network operators were selected because of their experience implementing SDN infrastructures.

There was another reason for selecting expert and elite interviewees. These interviewees were not limited to the expression of personal opinion. Given their seniority within their organisations, and for some, their seniority within the networking industry, they held significant power and influence (Kvale, 2007, p. 70; Flick, 2014a, pp. 227-232) for bringing into being the architectural transformation of existing production networking infrastructures, and the broad social and technical objectives of SDN, which are discussed in Chapter 5. As such, I projected that they would have certain insights – possibly not privileged to all – on how architectural evolution in sociotechnically ossified traditional networking infrastructures occurs.

Specific SDOs, network operators, vendors and systems integrators were selected via the purposive sampling strategy such that after gaining access, a secondary snowballing strategy could be implemented. The snowballing sampling strategy's objective was to use a selected interviewee's access to other knowledgeable individuals to secure additional interviews at the interviewee's organisation, or with contacts in the broader networking industry (Esterberg, 2002, pp. 93-94; Cohen, et al., 2011, pp. 158-160). Effectively, an interviewee's acquaintance was used to get an introduction to other reputable individuals (Cohen, et al., 2011, pp. 158-160) that might not have otherwise participated in this research. A secondary snowballing sampling strategy used was to ask personal contacts for introductions to their networking colleagues.

After a process of thematising that included specific study of the subject matter to be discussed by interviewees (Richards, 2005, pp. 24-25; Kvale, 2007, pp. 37-40; Yin, 2014, pp. 75-76), two topic guides (Gaskell, 2000; Esterberg, 2002, pp. 94-100; Kvale, 2007, pp. 57-60; Cohen, et al., 2011, pp. 415-421), of which samples are provided in Appendix B and Appendix C, were prepared for conducting semi-structured interviews (Esterberg, 2002, pp. 87-89; Kvale, 2007; Fielding & Thomas, 2008, pp. 246-247; Robson, 2011, pp. 285-286; Flick, 2014a, pp. 209,217). As explained previously, at the beginning of this research, there was an initial focus on infrastructure virtualisation, followed by a narrowing of the case to the advent of SDN infrastructures. The first nine interviews were conducted with cloud computing infrastructure experts, directed by the topic guide in Appendix C. All of those interviews, with the exception of one (Interviewee **I09**), included SDN as a topic of discussion. The remainder of interviews were conducted with networking experts, according to the topic guide in Appendix B. For clarity, as the research proceeded, the topic guides were refined to include direct questions about topics encountered in preceding interviews (Gaskell, 2000, p. 40), but the samples in Appendix B and Appendix C, include all general topics of questioning.

Table 4-1 List of Interviewees

Anonymised Alias	Type of Interviewee's Organisations	1st Interview Date	1st Interview Modality	1st Interview Length in Minutes	Anonymised Position Names
I01	Consultancy	25/07/2013	In Person	101	Sales Director at a Technology Firm
I02	Systems Integrator	12/08/2013	In Person	64	Vice President at a large virtualisation technology firm
I03	Systems Integrator/Network Operator	02/09/2013	In Person	51	High Performance Cloud Systems Architect at a technology firm with SDN products
I04	Systems Integrator/Network Operator	04/09/2013	In Person	54	Cloud Solution Architect at a technology firm with SDN products
I05	Systems Integrator/Network Operator	17/09/2013	In Person	115	CTO at a technology firm with SDN products
I06	Systems Integrator/Network Operator	18/09/2013	In Person	81	Lead in Cloud Computing at a technology firm with SDN products
I07	Systems Integrator/Network Operator	18/10/2013	Phone	25	IT Architect at a technology firm with SDN products
I08	Consultancy	07/11/2013	Phone	71	Managing Director at Cloud Consultancy Firm
I09	Systems Integrator/Network Operator	14/11/2013	Phone	15	Associate Partner at a technology firm with SDN products

I10	SDO	16/04/2014	Phone	35	Director at a SDO
I11	Network Operator	02/06/2014	In Person	69	Vice President in Networking at a Tier 1 ISP
I12	SDO/Vendor/Systems Integrator	05/06/2014	Phone	58	Senior Systems Engineer at a large Network Equipment Vendor and systems integrator
I13	SDO/Vendor/Systems Integrator	11/06/2014	Phone	55	Head of Architecture in a SDO
I14	SDO/Vendor/Systems Integrator	11/06/2014	Skype/Phone	54	Distinguished Engineer at a large network equipment vendor and systems integrator
I15	Vendor/Systems Integrator	08/07/2014	Phone	47	Senior Standards Manager at a large Network Equipment Vendor and systems integrator
I16	Vendor/Systems Integrator	10/07/2014	Phone	56	Research Manager at a large systems integrator
I17	Network Operator/Systems Integrator	11/07/2014	Phone/WebEx	52	Senior Network Architect for SDN and NFV at an ISP
I18	Vendor/Systems Integrator	11/07/2014	Skype/Phone	53	Solution Architect at a large network equipment vendor
I19	Network Operator	17/07/2014	Phone	55	Director of Core Networks at an ISP
I20	University	24/07/2014	Phone	58	Researcher in 5G networking
I21	SDO/Network Operator	29/07/2014	Phone	61	Lead in Network Architecture at a ISP

I22	SDO/Vendor/Systems Integrator	07/08/2014	Phone	65	Senior Director at a network equipment vendor and systems integrator
I23	Network Operator	08/08/2014	In Person	54	Operations Director at a Managed Service Provider
I24	SDO/Network Operator	22/08/2014	Phone	67	Director at a SDO
I25	Systems Integrator/Network Operator	17/09/2014	Phone	56	Vice President in Networking at a technology firm with SDN products
I26	SDO/University	19/09/2014	Skype/Phone	57	Researcher in SDN
I27	Network Operator	11/11/2014	Phone	52	Lead Member of Technical Staff at a Tier 1 ISP
I28	Test-bed Network Infrastructure	24/04/2015	Phone	45	Researcher in SDN
I29	SDO/Vendor/Systems Integrator	28/04/2015	Phone	53	CTO at a large network equipment vendor with SDN products
I30	SDO/Network Operator	29/04/2015	Phone	57	Director in Strategy in Networking at a Tier 1 ISP
I31	Test-bed Network Infrastructure	29/04/2015	Phone	62	Researcher in SDN
I32	University	29/04/2015	Phone	41	Researcher in SDN
I33	SDO/Network Operator	11/06/2015	In Person	57	Head of a technical committee at a SDO
I34	Network Operator	24/06/2015	Phone	49	Technology Specialist at a Tier 1 ISP
I35	Vendor/Systems Integrator	01/07/2015	Phone	50	CTO at a large network equipment vendor and systems integrator

I36	Vendor/Systems Integrator	03/07/2015	Phone	55	Product Line Manager at a large network equipment vendor and systems integrator
I37	Systems Integrator/SDN Research	24/07/2015	Phone	46	Deputy Head of R&D at a technology firm with networking products
I38	Systems Integrator	25/10/2015	Phone	61	Business Development Manager at a technology firm with system integration services
I39	Vendor	21/12/2015	WebEx	52	Solutions Architect at a large network equipment vendor

Table 4-2 Interviewees Given a Second Interview

Anonymised Alias	Type of Interviewee's Organisations	2nd Interview Date	2nd Interview Modality	2nd Interview Length in Minutes	Anonymised Position Names
I02	Systems Integrator	11/09/2013	Phone	48	Vice President at a large virtualisation technology firm
I03	Systems Integrator/Network Operator	11/09/2013	Phone	39	High Performance Cloud Systems Architect at a technology firm with SDN products
I07	Systems Integrator/Network Operator	06/11/2013	Phone	56	IT Architect at a technology firm with SDN products
I08	Consultancy	08/11/2013	Phone	27	Managing Director at Cloud Consultancy Firm
I38	Systems Integrator	01/11/2015	Phone	25	Business Development Manager at a technology firm with system integration services

4.3.5 The Interviewing Process

Of the thirty-nine interviews, thirty were conducted via telephone or similar communications means, and nine in person (Esterberg, 2002, pp. 100-102; Fielding & Thomas, 2008, pp. 252-253; Cohen, et al., 2011, pp. 439-442; Robson, 2011, p. 290). All interviews were recorded to digital audio (Esterberg, 2002, pp. 106-107; Kvale, 2007, pp. 93-94; Fielding & Thomas, 2008, pp. 257-258; Cohen, et al., 2011, p. 424). Telephone interviewing was the preferred method for conducting interviews because many of the interviewees were internationally dispersed, making in-person meetings impractical to arrange (Fielding & Thomas, 2008, pp. 252-253; Cohen, et al., 2011, pp. 439-442; Robson, 2011, p. 290).

At the beginning of each interview, I provided a brief introduction of the research, which reiterated and extended the content of interviewees' research access request letters (Esterberg, 2002, pp. 102-103; Kvale, 2007, pp. 55-56; Cohen, et al., 2011, pp. 421-422). As part of the introduction, I explained that IS research has a social science orientation rather than a technical or engineering orientation, but that I had a technical academic and professional background and therefore they could include technical details when elaborating their answers. The purpose of disclosing my technical background to interviewees (see also (Esterberg, 2002, pp. 90-92)) was threefold. First, it made interviewees aware that what I was investigating included the social practicalities of transforming traditional networks towards SDN infrastructures. Secondly, it liberated them to delve into technical details, as necessary, when trying to articulate subtleties of SDN infrastructures. Thirdly, it was intended to reduce some of the reversed power asymmetries that are inevitable in expert and elite interviewing (Kvale, 2007, pp. 14-15,70; Flick, 2014a, pp. 229-231), since they knew that their responses to questions would be under the scrutiny of an informed interviewer.

After introducing the interview, each interviewee was asked to confirm whether or not he or she wanted to proceed with the recorded interview, and only following confirmation, recording of the interview began (Esterberg, 2002, pp. 106-107; Kvale, 2007, pp. 93-94; Fielding & Thomas, 2008, pp. 257-258; Cohen, et al., 2011, p. 424). Interviews were ordered (Esterberg, 2002, pp. 96-98; Kvale, 2007, pp. 56-60) by asking some technically nuanced questions up front, in order to gain the respect of interviewees and thus further reduce, if possible, any power asymmetries that alter the format of the interview encounter (Kvale, 2007, pp. 14-15,70; Flick, 2014a, pp. 229-231), and to encourage interviewees to avoid defaulting to product marketing-styled responses.

Semi-structured interviewing provided opportunity for the experts to elaborate their answers to questions, and to introduce topics pertinent to understanding the advent of SDN infrastructures that might have risked exclusion if interviewing was too structured (Esterberg, 2002, pp. 87-89; Kvale, 2007; Fielding & Thomas, 2008, pp. 246-247; Robson, 2011, pp. 285-286; Flick, 2014a, pp. 209,217). As well, semi-structured interviewing permitted ample incorporation of probing and follow-up questions (Esterberg, 2002, pp. 104-105; Kvale, 2007, pp. 63-65; Fielding & Thomas, 2008, pp. 250-251; Cohen, et al., 2011, pp. 420-421), thus adding depth (Yin, 2014, p. 16) to the data collected for the case study.

In section 4.3.3, I explained the threat of theoretical architectural ideals to internal and construct validity (Cohen, et al., 2011, pp. 180-181,184-185,188-189; Yin, 2014, pp. 45-49). Another source of threats to internal and construct validity came from the technical solutions-oriented disposition of interviewees. It was important to ensure that interviewees were not responding to interview questions as though in the role of a technical consultant explaining to a network operator client how it is *possible* to architecturally evolve traditional networks towards SDN infrastructures on the basis of the pre-embedded features and generative capacity of available technological innovations. Indeed, what is *possible* is important, but again, sociotechnical ossification is not circumvented by the features of a technological innovation alone (Monteiro, 1998; Hanseth & Braa, 2000; Egyedi & Spirco, 2011; Sanner, et al., 2014; Grisot, et al., 2014). Therefore, in the introduction, interviewees were asked to respond to questions based on their *past* or *on-going experience and knowledge* of implementing SDN infrastructures, and I sought clarifications during interviews whenever this was not clear. Press releases and other published technical documentation were monitored and collected if confirmation that an interviewee was referring to production SDN infrastructures, was not done during the interview process. Some interviewees, depending on their responsibilities for defining SDN architecture at SDOs, were asked about how the architecture of SDN infrastructures *should be*, but these responses were only included in data collected for the second unit of analysis if deployment in production SDN infrastructures was confirmed via press releases or other published technical documentation. Any interviewee's response that was not confirmed to be about production SDN infrastructures, was explicitly set aside from data of the second unit of analysis. In this way, the interviews conducted may be classified as having proceeded under a factual interviewing format (Kvale, 2007, p. 71).

On conclusion of interviews, snowballing for additional interviewees was done (Esterberg, 2002, pp. 93-94; Cohen, et al., 2011, pp. 158-160), where appropriate.

4.4 Data Analysis

4.4.1 Introduction

The objective of the data analysis (Cohen, et al., 2011, pp. 538-540) was to find and to establish, through retroduction, the reality of the generative mechanisms sought by the research question. Practically, the data analysis, though guided by retroductive reasoning (Reichertz, 2014), involved the use of specific qualitative data analytic and technical analytic procedures to arrive at an explanation for the advent of SDN infrastructures. This section explains what analytic techniques were used, and how they were applied to answer the research question.

As stated in the preceding section, data analysis was intermingled with data collection. Preliminary data analysis was conducted in February 2013 and March 2013, and between October 2013 and November 2013 initially on infrastructure virtualisation, and then during August 2014 and December 2014 on the advent of SDN infrastructures. This analysis consisted of pre-coding of qualitative interviews (Saldaña, 2013, pp. 19-21), diagrams and notes, and methodological and early analytic memo writing (Esterberg, 2002, pp. 165-166; Richards, 2005, pp. 73-75; Gibbs, 2007, pp. 30-32; Saldaña, 2013, pp. 41-54). These were primarily handwritten and kept in notebooks, with some written in electronic documents. The main data analysis phase started in February 2015 and ended in March 2016. A review of all coded and themed data was conducted between May 2016 and June 2016. An additional a review of all coded relationships between themes was conducted between June 2016 and August 2016. The main data analysis phase was supported by the use of NVivo, a computer-assisted qualitative data analysis (CAQDAS) software application (Esterberg, 2002, pp. 176-179; Gibbs, 2007, pp. 105-123; Cohen, et al., 2011, pp. 542-546; Saldaña, 2013).

4.4.2 First Unit of Analysis

To reiterate, the first unit of analysis was of the technical details of the architecture of SDN infrastructures. Because production SDN infrastructures were inaccessible, documentation was analysed to establish the architecture of SDN infrastructures, and archival data were analysed for the architectural history of SDN infrastructures. The term "architectural history" refers to the progression of predecessor architectures in SDN innovations up to the core architecture of production SDN infrastructures. Qualitative interview data was analysed for complementary technical details of the architecture of SDN infrastructures. The analysis of documents is presented first, and then the analysis of interview data.

Documents and archival data were technically analysed using architectural analysis techniques. Architectural analysis can be described as *"the activity of discovering important system properties using the system's architectural models"* (Taylor, et al., 2010, p. 291)[18]. This activity involves investigation of structural and functional aspects of architecture (Crawley, et al., 2016, pp. 58-63). There are no well-known qualitative methods for conducting the architectural analysis required for the first unit. Thus, as stated in the description of the data collection strategy for the first unit of analysis, my technical background was of assistance in data collection, and in the analysis. Elaborating the formal particulars of architectural analysis goes beyond the scope of an information systems thesis, but I will still provide a very high-level summary of key activities of the analysis. An interested reader can refer to Taylor, et al. (2010) and

[18] Taylor, et al. (2010) use the term "system", but the definition is appropriate for this context.

Crawley, et al. (2016) for useful introductions to the formal particulars of architecture and architectural analysis.

Analysis activities included:

- Establishing the underlying infrastructure requirements for supporting the deployment of SDN infrastructures. For example, are there particular technologies, techniques, physical or software components, or management required?
- Tracing and comparing the key architectural components of SDN as an innovation, with that of traditional networking architectures. For example, are infrastructural components added, removed or re-arranged – and in what part of the network are these changes made?
- Applying modular analysis to identify what interfaces are introduced, and the functional purpose of those interfaces.
- Establishing how any new infrastructural components interact and with what they interact. For example, do they interact with existing routers, how is this interaction facilitated, what is the nature and purpose of this interaction?
- Establishing which architectural differences between SDN as an innovation and traditional networking infrastructures foundationally define SDN infrastructures, as opposed to being peripherals of implementation.
- Identifying what types of data models or data exchanges are introduced. For example, how are routers or other network devices configured in SDN infrastructures, or what new protocols are introduced to the networking infrastructures?
- Ascertaining the practicalities of instantiating architectural concepts in production SDN infrastructures. For example, if new software is required, where does it execute, and what manages it – a human, an operating system, infrastructure management and orchestration? Alternatively, what production-capable SDN products are available to network operators?

- Comparing architectural features of commercially available SDN products with SDO prescribed architectures. For example, what interfacing and protocols are supported across different products?

The aforementioned broadly describes the architectural analysis. Practically, the analysis involved searching within documentation and archival data to identify architectural details that answered the questions, and taking note of these. I did not create formal computer-based architecture models of SDN architecture. Instead I took notes, and drew diagrams. One threat to construct and internal validity came from my technical background. Similarly to the treatment of elite interviewees' responses, I had to retain a reflexive awareness (Gibbs, 2007, pp. 91-93) of the distinction between architectural details described within documentation and product features, and what was actually implemented in SDN infrastructures. Beside technical details of architecture, the analysis of archival data uncovered insights on the history and motivations for SDN as an innovation. These contributed towards the analytic history of structural conditioning (Archer, 1995, pp. 149-161,165-218; Archer, 1998) prior to the advent of production SDN infrastructures.

The analysis of interview data (for both units of analysis) was organised as a two-level *Thematic Coding* strategy (Saldaña, 2013, pp. 22-24). The first level of coding broadly categorised interviewee responses by high-level topic. The second level of coding involved applying specific coding methods that were selected based on the characteristics of the data within each high-level topic (Saldaña, 2013, pp. 22-24). The purpose of this two-level coding strategy was to ensure that data was coded in a manner that remained faithful to the context of interviewee responses. For example, what SDN is as an innovation, is different from the reasons for the advent of SDN infrastructures, but some isolated data if coded line-by-line, or at a similarly granular level, could be coded to either high-level topic. Given that there is an existing context of the data, such coding may threaten internal validity and reliability (Gibbs, 2007, p. 96). The problem was mitigated by coding more broadly by high-level topic first, and as the second level of coding progressed into later stages of analysis, searching for relationships that may exist between data in codes created under different high-level topics and coding these as relationships (in NVivo).

Analysis of interview data began with preliminary reading, which is the reading of transcripts to broadly grasp what topics had been covered in interviewees' responses (Robson, 2011, pp. 476-478; Saldaña, 2013, p. 143) about the architecture of SDN infrastructures. Preliminary reading of transcripts was followed by *Holistic Coding* (Richards, 2005, pp. 92-93; Saldaña, 2013, pp. 142-144) of interview data that categorised the data at a high-level by topic. Eight pre-defined holistic codes were used. These codes were defined *a priori* on the basis of the original aims of the lines of questioning in the interviews conducted (for both units of analysis), and on initial impressions gathered from preliminary reading (Fielding & Thomas, 2008, pp. 259-260; Gibbs, 2007, pp. 44-45). Holistic Coding was applied to the entirety of interview text, with the exception of the first nine interviews in which only responses related to SDN, or those that defined virtualisation (to be discussed in the Findings chapter) were coded (Saldaña, 2013, pp. 16-17).

The definition of each high-level code was stored within a detailed codebook (Saldaña, 2013, pp. 24-25). For each code, the criteria for identifying data that matches it, one or more examples of typical and optionally atypical qualifying data excerpts, exclusion criteria for when the code may seem to apply but does not, and optionally an example of data that meets the exclusion criteria with a reason for why it does not, was documented (Saldaña, 2013, pp. 24-25). Though this resulted in a large codebook, this level of detail was kept and continuously consulted to safeguard the analysis from threats to reliability and validity due to definitional drift of codes over time (Richards, 2005, pp. 98-100; Gibbs, 2007, pp. 98-99; Cohen, et al., 2011, p. 560). As the analysis proceeded into the second level of coding, five additional data-derived holistic codes were created. Whenever holistic codes were added, the inclusion and exclusion criteria for all existing holistic codes were reviewed and refined, and all data assigned to refined codes were re-coded.

The second level of coding involved data-driven Thematic Coding of the data coded at each high-level holistic code. All data-driven codes were added to the codebook. The objective of this stage of coding was to produce analytic themes, through multiple coding cycles that transitioned through initial coding, analytic coding and the development of themes (Esterberg, 2002, pp. 157-160; Richards, 2005, pp. 87-95;

Gibbs, 2007, pp. 38-46,71-89; Cohen, et al., 2011, pp. 559-561; Saldaña, 2013, pp. 9-14,87-91,100-105,175-183,202-206). The high-level holistic codes were coded and analysed one at a time in isolation from each other. As themes were identified, codes were continuously compared and revisited to search for relationships within and across high-level holistic codes (Gibbs, 2007, pp. 73-89,96; Saldaña, 2013, pp. 9-14,194-209). Additionally, throughout the data analysis, analytic memos and research summaries were written for the purpose of refining the developing themes and theoretical insights (Esterberg, 2002, pp. 165-166; Richards, 2005, pp. 73-75; Gibbs, 2007, pp. 30-32; Saldaña, 2013, pp. 41-54). As the analysis progressed, data models and diagrams were created and continuously refined to capture emerging theoretical insights (Esterberg, 2002, pp. 162-164; Gibbs, 2007, p. 86; Saldaña, 2013, pp. 202-204).

While coding and analysing the data assigned to each high-level holistic code, it was determined that some high-level codes were better coded using specific coding methods. Of relevance to the first unit of analysis was the application of *Versus Coding* (Saldaña, 2013, pp. 115-119) to distil how SDN infrastructures are architecturally and capably differentiated from traditional networking infrastructures. For clarity, Versus Coding was not used exclusively for the first unit of analysis. From a technical perspective, Versus Coding was used to identify what architectural changes constitutes architectural evolution that yields SDN infrastructures. From a critical realist perspective, Versus Coding contributed towards uncovering part of the explanation, from an architectural perspective, of what underpinned antecedent structural conditioning of traditional networking infrastructures, and conditions subsequent to the advent of SDN infrastructures.

Similarly to the analysis of documents, interview data was analysed for technical details of the architecture of SDN infrastructures as explained by interviewees. Indeed, analysis of interview data requires the awareness that the analysis procedure involves interpreting the interviewee's interpretation of phenomena and events (Kvale, 2007, p. 144; Cohen, et al., 2011, p. 540), but recall that the interviewees were highly technical individuals who were significantly involved in the creation of the events under study, and that interviews proceeded under a factual interviewing format (Kvale, 2007, p. 71). As such, technical details reflected the architecture found in production SDN infrastructures. Data and methodological triangulation via the use and analysis of documents, provided complementary and confirming data and findings to those of the interview data and its analysis (Mabry, 2008; Cohen, et al., 2011, pp. 195-197,299-300; Flick, 2014a, pp. 182-192; Yin, 2014, pp. 119-122). This was established by comparing the technical details of architecture in SDN infrastructures produced by the analysis of both types of data collected by the two methods (Mabry, 2008; Cohen, et al., 2011, pp. 195-197; Flick, 2014a, pp. 188-190; Yin, 2014, pp. 119-122). The analysis of documents produced more detailed technical findings which were treated as complementary to those of the interview data.

4.4.3 Second Unit of Analysis

To reiterate, the second unit of analysis was of the social objectives of the advent of SDN infrastructures entrenched in the interests of three major *types* of organisations, in the networking industry, that created networking technology, or owned networking infrastructures, and were involved in SDN technological innovation, and SDN infrastructure deployments. The details of how interview data was generally analysed will not be restated here. Instead analytic procedures that are specific to the second unit of analysis are presented in this section.

As a critical realist analysis, the second unit of analysis contributed most strongly to the broad framing of morphogenetic cycles related to the advent of SDN infrastructures. Aside from general thematic coding methods, the search for an explanation of why SDN infrastructures came about at a particular point in time, and for the sociotechnical processes of how ossified networking infrastructures were being architecturally transformed into SDN infrastructures, was assisted by the use of a *Causation Coding* method (Saldaña, 2013, pp. 163-175). Causation Coding was used to identify antecedent causally related conditions that predated the advent of SDN infrastructures, mediating conditions that catalysed the advent, and the outcome of these (Saldaña, 2013, pp. 163-175). Through a critical realist analytic lens, antecedent conditions were treated as anterior structural conditioning (Archer, 1995, pp. 149-161,165-218; Archer, 1998), mediating conditions were analysed for enabling and stimulating conditions, and outcomes provided a releasing condition for the generative mechanisms sought by the research question (Pinkstone & Hartwig, 2007, p. 458; Hartwig, 2007, p. 80; Fleetwood, 2011).

Interview data was also analysed for subsequent conditions of structuring, whom they conditioned, and how they conditioned, were analysed (Archer, 1982; Archer, 1995, pp. 165-194). A *Process Coding* method (Saldaña, 2013, pp. 96-100) was further utilised to develop understanding of the sociotechnical processes that brought about the advent of SDN infrastructures, and processes facilitating continued deployment of production SDN infrastructures. As stated in the data collection section, press releases were continuously monitored primarily for corroborating interviewee statements (Mabry, 2008; Cohen, et al., 2011, pp. 195-197; Flick, 2014a, pp. 182-192; Yin, 2014, pp. 107,119-122).

4.4.4 Conclusion of Analysis

Throughout the analysis, the case of the advent of SDN infrastructures was kept focal (Yin, 2014, pp. 55-56), with the retroductive process drawing on findings from both units of analysis to establish as real, the operation of particular causal generative mechanisms. Data analysis concluded on theoretical saturation (Eisenhardt, 1989). I present analytic arguments for the reality of these generative mechanisms in Chapter 6. Finally, there was an extended period of monitoring and collecting press releases that continued until September 2016, after data analysis was completed. The purpose was to search for potential shortcomings in the proposed theoretical contributions based on the latest developments in the networking industry.

4.5 An Account of Retroducing a Generative Mechanism

4.5.1 Introduction

This section demonstrates a retroductive process by providing the methodological account of how one generative mechanism sought by the research question was identified. Its purpose is to provide support for the research's credibility and methodological reliability (Richards, 2005, pp. 43-44; Cohen, et al., 2011, pp. 201-204; Robson, 2011, pp. 93,155,159; Yin, 2014, pp. 48-49). As already stated, following the refinement of the case, the retroductive process started by considering the existence of production SDN infrastructures, and then seeking an explanation of how they came into being. Specifically, I sought an explanation of how architectural evolution in traditional networking infrastructures to yield SDN infrastructures occurred.

To reiterate, counter to extant IS theorising on architectural evolution in digital infrastructures, in the advent of SDN infrastructures, network operators' networking infrastructures were not replaced due to the need to introduce new underlying architecture (Hanseth & Lyytinen, 2010; Grisot, et al., 2014). Further, generally for networking infrastructures, the use of gateways as a means of underlying architectural evolution, the second position of IS theorising on architectural evolution in digital infrastructures (Hanseth, 2001; Hanseth & Lundberg, 2001; Edwards, et al., 2007; Egyedi & Spirco, 2011), has been limited to problems that are narrow in scope (Monteiro, 1998), and was not the means by which SDN infrastructures came about. Architectural evolution by interconnection, which is the third position taken in IS research explaining digital infrastructure scaling and evolution (Hanseth, 2001; Hanseth & Lundberg, 2001; Edwards, et al., 2007; Hanseth & Lyytinen, 2010; Grisot, et al., 2014), is limited to deployment architecture evolution which does not change underlying architecture in digital infrastructures. Therefore, it did not provide theoretical insight into how underlying architecture in such extensively sociotechnically ossified traditional networking infrastructures was evolved. Given these shortcomings of existing IS theorising, I searched for an alternative explanation (Sayer, 2000, pp. 13-17; Easton, 2010; Wynn & Williams, 2012; Reichertz, 2014; Kelle, 2014, pp. 561-562), framed by Archer's critical realist morphogenetic approach to the transformation of structure (Archer, 1982; Archer, 1995), to ascertain how from an architectural perspective, production SDN infrastructures came about.

To be clear, the research sought an explanation of the *transformation* of networking infrastructures from an architectural perspective. The generative mechanisms sought then, were specifically limited to those whose operation were causal to the architectural transformation of existing traditional networking infrastructures as sociotechnical structures. According to Archer, what plays out from an analytically isolated temporal point of structural conditioning until a corresponding analytically isolated later temporal point of structural elaboration, *is a generative mechanism of morphogenesis* (Archer, 1995, p. 217). Thus, the process of retroduction in this research, was a retrospective search (Wynn & Williams, 2012) for the generative mechanism *of morphogenesis* through which SDN infrastructures came into being. Generative mechanisms whose operation did not transform the existing necessary internal relations of structures or introduce new necessary internal relations, remained outside of the boundaries of this research's analysis. Examples of such non-focal generative mechanisms are morphostatic generative mechanisms which operate to sustain the existence of a structure (Archer, 1995, p. 217; Elder-Vass, 2007), and generative mechanisms whose operations do nothing to transform or to reproduce a structure and are instead the *sui generis* relatively enduring ways of acting of a structure (Archer, 1982; Collier, 1994, pp. 62,107-134,138-141; Archer, 1995, pp. 135-161,172-183,217; Archer, 1998; Bhaskar, 1998a; Archer, 2000; Elder-Vass, 2007). Indeed, some other generative mechanisms that are of SDN infrastructures sui generis, are also identified in Chapter 6, but these are presented because they promote architectural evolution post-morphogenesis.

I preliminarily hinted in the Introduction chapter that *softwarisation* corresponds to *the* generative mechanism of morphogenesis whose operation (alongside the operation of a social generative mechanism of installed base cultivation, which I develop in Chapters 5 and 6) accounts causally for the architectural transformation of traditional sociotechnical networking infrastructures into SDN infrastructures. Chapter 6, identifies five morphogenetic sociotechnical generative mechanisms that, when in a particular necessary synchronic arrangement (Elder-Vass, 2007), are causal to the emergence of the softwarisation generative mechanism of morphogenesis. Here, I present a methodological overview account of how one of these morphogenetic sociotechnical generative mechanisms, Technical Disaggregation, was retroduced.

4.5.2 Retroducing Technical Disaggregation

After designating the advent of SDN infrastructures as being the phenomenon that required explanation, the next step in the process of retroduction was identifying what characterised the phenomenon of study (Easton, 2010). There were two aspects, which align to the two units of analysis. I needed to understand the technical details of the architecture of SDN infrastructures. I also needed to understand why SDN infrastructures came about at a particular point in time, and the sociotechnical processes of how ossified networking infrastructures were being architecturally transformed into SDN infrastructures.

In relation to the first unit of analysis, I needed to clearly ascertain the following:

1. What is a production SDN infrastructure?
2. What architecturally distinguished SDN infrastructures from the traditional networking infrastructures from whence they came?
3. What were the necessary (as opposed to optional) intervening architectural transformations?

Partial answers to these questions were facilitated through the data collected via the use of documents and by conducting qualitative interviewing. The analysis of the documents and qualitative interviews produced two insights.

First, it revealed that key to the architectural transformation of networking infrastructures was a change in the relationship between the forwarding and control planes in routers. In harmony with a retroductive reasoning strategy, I first drew on existing theorising (Reichertz, 2014; Kelle, 2014, pp. 561-562) in information systems which state that the architecture of digital infrastructures should be modular (Hanseth, et al., 1996; Edwards, et al., 2007; Hanseth & Lyytinen, 2010; Yoo, et al., 2010), and I preliminarily appropriated modularity's principle of loose coupling (Sanchez & Mahoney, 1996; Schilling, 2000; van Schewick, 2010, pp. 38-44; Hanseth & Lyytinen, 2010; Yoo, et al., 2010) to frame the changed relationship as a process of *decoupling*. I searched to see to what extent this principle of modularity could help to explain the advent of SDN infrastructures, and hence how architectural evolution in sociotechnically ossified digital infrastructures occurs. Second, as data analysis proceeded, I noticed that interviewees were fixated on discussing issues of changes to existing inter-organisational structures, relationships and processes, the assignment and reassignment of the roles of existing organisation types, and how incumbent network operators, network equipment vendors, and systems integrators, were positioning or repositioning themselves within the networking industry, post the advent of SDN infrastructures.

Accordingly with the logic that guides retroduction (Easton, 2010, p. 124), it became apparent from the analysis that an important part of understanding the advent of SDN infrastructures, was to be found in *why* they came about. In the absence of an answer to the question of “Why?”, the critical realist causal account and thus the explication of the generative mechanism of morphogenesis, risked being either incomplete or causal attribution could be made to wrong (Sayer, 2000, pp. 16-17), inconsequential or non-existent morphogenetic generative mechanisms (Mingers, 2004b, p. 390; Easton, 2010; Wynn & Williams, 2012).

In the preceding sections I presented two predefined units of analysis, but chronologically it was only after the initial analysis that these units of analysis were refined. This was followed by a re-analysis of all data with attention paid to the second unit of analysis, along with new rounds of data collection (Easton, 2010; Wynn & Williams, 2012; Reichertz, 2014; Kelle, 2014, pp. 561-562). As part of this re-analysis, I introduced a new holistic code (Richards, 2005, pp. 92-93; Saldaña, 2013, pp. 142-144) under which interviewees’ explanations of the reasons for the advent of SDN infrastructures were coded primarily using cycles of Causation Coding (Saldaña, 2013, pp. 163-175).

The outcome of this was twofold. The first was an analytic history (Archer, 1995, pp. 149-161,165-218; Archer, 1998), which is presented in Chapter 5 as the structural conditioning of traditional networking infrastructures as sociotechnical structures as from the negative causal mode of constraint (Hartwig, 2007, pp. 57,80,458), that explained why network operators started transforming their traditional networking infrastructures into SDN infrastructures at a particular point in time. Second, the analysis revealed that the advent of SDN infrastructures was framed as being closely associated with and facilitative of a transformation in the existing necessary internal relations of interdependence between network operators, network equipment vendors, and network systems integrators of the networking industry. Network operators and SDOs wanted to change the necessary internal relations of the networking industry – specifically the relations of interdependence between network operators, network equipment vendors and to some extent network systems integrators – and to introduce new necessary internal relations and relata via the entrance of new types of networking innovation organisations.

As such, the analysis became one of the morphogenetic cycles of two structures: networking infrastructures as sociotechnical structures considered from an architectural perspective, and the networking industry as a social structure. Additional data was collected and analysed to substantiate whether any morphogenesis of the networking industry as a social structure had taken place (Easton, 2010; Wynn & Williams, 2012; Reichertz, 2014; Kelle, 2014, pp. 561-562).

As explained in Chapter 3, generative mechanisms instantiated with structures and properties have the characteristic of intransitivity (Collier, 1994, p. 62; Bhaskar, 1998a). Therefore, the process of retroducing morphogenetic generative mechanisms whose operation led to an outcome of structural elaboration, necessitated an analysis that searched for the diachronic operation of powers not limited to local actions of any singular network operator internally innovating its networking infrastructure. Further, the exposition of the two morphogenetic cycles needed to causally account for the relationship between the transformation of the networking industry as a social structure, and the transformation, from an architectural perspective, of traditional networking infrastructures to yield SDN infrastructures. I proceeded to search with these requirements in mind for causal explanation of the transformation of multiple large network operators' networking infrastructures towards SDN infrastructures, for changes that appeared in the networking industry, and for the relationship between the two.

During this analytic process, I determined that I would need more than the explanatory capacity of modularity theory's principle of decoupling to explicate these interrelated morphogenetic cycles of structures. Though the activities of SDOs remained in the background of the case, the analytic history revealed that the continuous work of SDOs underpinned the transformation of network operators' traditional networking infrastructures towards SDN infrastructures, and was implicated in the subsequent corresponding changes that occurred within the networking industry. Accordingly, I adapted installed base cultivation as a social generative mechanism from the IS digital infrastructure literature on the formation and evolution of digital infrastructures (Monteiro, 1998; Hanseth & Lyytinen, 2010; Aanestad & Jensen, 2011; Sanner, et al., 2014; Grisot, et al., 2014; Reichertz, 2014; Kelle, 2014, pp. 561-562). Because the work of SDOs was continuous, I proceeded with the analysis, with the understanding on the basis of the analytic history and additional data collected, that the operations of any morphogenetic generative mechanisms to be found were facilitated by and acted concurrently with the operation of installed base cultivation.

The analysis also revealed that part of the outcome of installed base cultivation was the adoption of a type of architectural transformation of networking infrastructures owned by multiple large network operators which created conditions of structuring that catalysed a morphogenetic cycle of the networking industry as a social structure. (I address the issue of technological determinism at the end of Chapter 6). But given that this research wanted to go beyond the social aspects of installed base cultivation, the analysis sought specifics of one or more morphogenetic generative mechanisms whose operation produced the outcome of structural elaboration in spite of conditions of prohibitive sociotechnical ossification. These morphogenetic generative mechanisms needed to address technical aspects of the transformation of traditional networking infrastructures as sociotechnical structures, from an architectural perspective.

Initially, it seemed as if the transformation of the necessary internal relations of the networking industry directly corresponded in *type* with the architectural transformations in network operators' networking infrastructures. Namely, the aforementioned change in the relationship between the forwarding and control planes in routers, seemed to correspond with the subsequent type of changes of relations between the three types of organisations in the networking industry. I returned to modularity theory again, to determine whether the advent of SDN infrastructures could be theoretically explained by considering how the decoupling principle might have underpinned the morphogenetic cycles. Excerpts from an analytic memo provide a view of this:

August 26th, 2015 - Theory Memo - Disaggregation as a Facilitating Underlying Process

> *"Disaggregation is the term used to describe the decoupling of layers of hardware and software components in a manner that allows organisations/people to be able to sell or buy components from multiple disparate vendors and to recombine them in a modular compatible manner.*
>
> *In networking, the process of disaggregation refers to the decoupling of software that runs on networking elements from the hardware which carries out the intentions of the software. That is, the software and hardware become independently substitutable.*
>
> *The decoupling in networks and elsewhere in the datacentre goes further to allow not only organisations to purchase components from various vendors, but to also have different service contracts. …*
>
> *With disaggregation, service contracts are not necessarily tied to software and hardware. The decoupling of the control plane from the forwarding plane in SDN, is a network-specific example of a wider disaggregation process. …*
>
> *Though disaggregation can be the term used to describe the process, from a theoretical perspective, modularity theory can be a more formal way to anchor this."*

Findings from continued analysis showed, however, that neither the architectural transformation of traditional networking infrastructures, or the transformation of the networking industry, fit neatly into a broad disaggregation generative mechanism of morphogenesis. Disaggregation did not explain *how* architectural decoupling was accomplished, and the analysis had already revealed elsewhere more than the operation of a singular disaggregation generative mechanism in the circumvention of sociotechnical ossification of traditional networking infrastructures. That is, even with the concurrent operation of an installed base cultivation social generative mechanism, the principle of decoupling, drawn from modularity theory could not alone sufficiently explain the architectural transformation of network operators' traditional networking infrastructures. Disaggregation of networking functionality was certainly present, but by itself, disaggregation as a generative mechanism did not carry sufficient explanatory power to be *the* generative mechanism of morphogenesis of both morphogenetic cycles. If disaggregation was a morphogenetic generative mechanism, it was one that helped to explain (Collier, 1994, p. 109) another emergent generative mechanism of morphogenesis.

Previously, in early preliminary data analysis before the case was refined to the advent of SDN infrastructures, I had noticed elements of this problem while investigating why certain foundational principles of modularity theory seem to be violated in virtualised cloud computing infrastructures – particularly within their networks. At that time, I had appropriated insights from the computer science and software engineering literature on the nature of software and from information systems theorising on digital materiality, to make sense of what the analysis had shown. Two excerpts of analytic memos that contain early musings about elements of what I had noticed follow:

January 13th, 2013 – "Digital Fluency" or "Digitally Fluent Environments"

> *"For example,* ***a*** *virtualisation is a finite set of virtual objects, but a virtualisation is not in its truest sense a virtualisation unless it is at runtime. To illustrate, if Amazon's network virtualisation platforms went down, the data centre would go offline. But this doesn't mean that the various virtualisations are lost. These virtualisations are indeed stored as data … but as data they are virtualisations at most in a descriptive sense until they are instantiated and restored to their runtime … state."*

March 4th, 2013 – "Digital Fluency" or "Digitally Fluent Environments"

> *"A lot of information systems research is still treating digital objects as though they are solely data entities to be acted upon, by external technology. Digital Fluency however, is looking at a specific class of digital objects that intrinsically have behaviour and seriously blur the boundaries of what is data versus software as in the case of SDN virtualisation."*

The insights from this early work continued to be developed throughout the analysis without causal connection to the advent of SDN infrastructures, until I returned to this prior work to analyse whether there was any relation of those insights specifically to architecture in digital infrastructures.

Following what the ensuing analysis showed, I refined the disaggregation that characterised the architectural decoupling as a Technical Disaggregation in network operators' networking infrastructures, and searched for the reality of other morphogenetic generative mechanisms and for any interactions and relations between them and Technical Disaggregation. Chapter 6 presents extended arguments for the reality of other morphogenetic generative mechanisms that operated in a diachronic temporal sequence and in concurrency with Technical Disaggregation in a particular necessary synchronic arrangement (Archer, 1995, pp. 149-161,165-218; Archer, 1998; Elder-Vass, 2007) to exploit an inherent capacity of digital infrastructures and bring about architectural transformation. In a similar manner, I later refined the disaggregation that characterised the changes of the networking industry social structure as Social Disaggregation, a social generative mechanism of either morphogenesis or morphostasis. In Chapter 6, I explain why its morphogenetic or morphostatic designation was not definitive by the conclusion of this research.

4.6 Generalisability of Case Study Theoretical Contributions

4.6.1 Single Case Study Generalisation

Generalisation is about making arguments (Firestone, 1993) for the extension of the conclusions of a research undertaking to other research contexts (Gobo, 2008; Polit & Beck, 2010; Maxwell & Chmiel, 2014). Generalisations are always tentative arguments (Kennedy, 1979; Firestone, 1993) that aim to persuade about the suitability of research results to inform in unknown and unobserved contexts (Polit & Beck, 2010). Arguments for demonstrating the generalisability of findings include: *Empirical Generalisation, Internal Generalisation, Analytic Generalisation, Transferability, and External Generalisation.*

Empirical generalisation arguments (Maxwell & Chmiel, 2014) are premised on the assertion that generalisation can be achieved by extrapolation of findings from a representative sample to a population. Practically, empirical generalisation is usually demonstrated via application of probability theory (Kennedy, 1979; Firestone, 1993; Gobo, 2008; Maxwell & Chmiel, 2014; Yin, 2014, p. 40). Typically, it relies on identification beforehand of the population about which inferences will be made from a representative sample (Gobo, 2008; Polit & Beck, 2010).

Overlapping empirical generalisation is internal generalisation (Maxwell & Chmiel, 2014). Internal generalisation arguments follow a logic that generalisation *within* the research context studied, i.e. the case, can be achieved by extrapolation to entities not studied within the same research context (Maxwell & Chmiel, 2014).

Differently from empirical generalisation, the concern of analytic generalisation is not to demonstrate that inference to a population constitutes a valid generalisation argument. For analytic generalisation, the logic of the argument is that generalisation can be made to theory (Firestone, 1993; Yin, 2014, pp. 20-21, 40-41, 68). Theoretical insights may then illuminate contexts of research and practice that were not directly under study (Firestone, 1993; Yin, 2014, p. 41). Practically, analytic generalisation is demonstrated by developing new theory, or by making refinements to theory upon which a research undertaking was premised using support from research results (Maxwell & Chmiel, 2014; Yin, 2014, p. 41).

Transferability arguments are premised on the assertion that the responsibility for deciding the extent to which research results are generalisable to another context, is the reader's. The reader of research results considers the researcher's argumentation and thick description of the research context and research procedures (Firestone, 1993; Gobo, 2008), and decides whether there is suitable temporal and contextual similarity (Gobo, 2008) to make an appropriate transfer of knowledge to another context (Kennedy, 1979; Firestone, 1993; Gobo, 2008; Polit & Beck, 2010; Maxwell & Chmiel, 2014). Because the researcher does not know what knowledge may be subjected to transfer, when relying on transferability as the generalisation argument, the researcher must consider what type of contextual information might be important to include within a thick description (Kennedy, 1979; Firestone, 1993; Polit & Beck, 2010).

External generalisation (Maxwell & Chmiel, 2014) is a broad argument that generalisation can be made to contexts of research and practice that are outside of the research context studied. Empirical generalisation (unless it is an internal generalisation argument), analytic generalisation and transferability arguments, are external generalisation arguments (Maxwell & Chmiel, 2014).

Single case studies, such as the advent of SDN infrastructures, generalise via theoretical contribution, and therefore require an analytical generalisation argument (Yin, 2014, pp. 20-21; 40-41). This is the generalisation argument that I will seek to establish in the Discussion chapter – specifically that the research results generalise satisfactorily to theory.

4.6.2 What Kind of Theory?

According to Gregor (2006), there are at least four goals of theory in IS research: *Analysis and Description, Explanation, Prediction, and Prescription*. The objective of analysis and description is to elucidate in detail the object of research, outlining relationships between its constructs, and placing limits on the generalisability of the findings. The description excludes elaborations of causality. Explanation seeks to explicitly discover, and clarify causal relationships and processes to explain the phenomena. Prediction, on the basis of analysis, seeks to derive propositions about the future that reliably come to pass, provided that empirically established predefined conditions are satisfied. Prescription, seeks to produce guidelines for how a process is to be carried out to fulfil some purpose – such as the creation of an IT artefact.

Five types of IS theories correspond to the aforementioned goals of theory (Gregor, 2006): *Theory for Analysing, Theory for Explaining, Theory for Predicting, Theory for Explaining and Predicting, Theory for Design and Action*. These theory types have a straightforward correspondence to the goals of theory, with Theory for Design and Action being premised on the goal of prescription.

The research question of this empirical investigation sought causal explanations. Theory for explaining or *Explanatory Theory*, and theory for predicting hold different perspectives on causality, and hence are organised to different logical structures (Markus & Robey, 1988; Gregor, 2006). Theory for predicting follows a *variance model* in which the logical structure of causation is deemed satisfactorily captured by the identification of necessary and sufficient preconditions and corresponding outcomes should those preconditions be met (Markus & Robey, 1988). A logical structure of causation based on the variance model allows extrapolations of how much of a precondition is necessary to be sufficient for an outcome, unlike with a process model which does not accommodate such logic in arguments for causality (Markus & Robey, 1988). Explanatory theory follows a *process model* in which the logical structure of causation consists of an account of the processes by which an outcome is reached (Markus & Robey, 1988). With the process model logical structure of causation, it is asserted that outcomes may also fail to occur regardless of the detail of causal elaboration (Markus & Robey, 1988). The process model logical structure of causation in explanatory theory is thus harmonious with critical realism's formulation of causality and explanation in open environments. Accordingly, the proposed theoretical contributions of this research form an explanatory theory that seeks to explain, via a process model logical structure of causation (Gregor, 2006; Avgerou, 2013).

Explanatory theory explains how and why things occurred or occur as they did (or do), and elucidates causal attribution, and causal associations, relationships, and processes (Markus & Robey, 1988; Gregor, 2006; Avgerou, 2013). It is through the knowledge of attribution of causality, causal associations, relationships, and processes that explanation is achieved (Markus & Robey, 1988). Insights from explanatory theory should be new and illuminative of poorly or imperfectly understood phenomenon (Gregor, 2006). The development of explanatory theory should also demonstrate that a search for alternative explanations has been carried out (Eisenhardt, 1989; Gregor, 2006). This is in accordance with the requirements for conducting critical realist research (Sayer, 2000, pp. 13-17; Easton, 2010; Wynn & Williams, 2012) – and case study research (Yin, 2014, pp. 140-142,147-150). According to (Gregor, 2006, p. 625), the contribution to knowledge of explanatory theory is evaluated on *"plausibility, credibility, consistency, and transferability of the arguments made."* For research whose goal is explanation, the research product itself is an explanatory theory (Gregor, 2006).

4.6.3 What Constitutes a Theoretical Contribution?

The preceding section presents requirements of explanatory theory. But what are the requirements of a theoretical contribution?

Theoretical contributions should be insightful, characterised by originality, and have utility for research and practice (Whetten, 1989; Corley & Gioia, 2011). Logic rather than statements of data facilitates explanation (Whetten, 1989; Sutton & Staw, 1995), though, depending on the purpose and maturity (DiMaggio, 1995; Weick, 1995), a theoretical contribution may at the time of its proposal represent an intermediate state of understanding that may warrant a greater dependence on statements of data (Weick, 1995).

Whether considered intermediate or mature, theoretical contributions should go beyond restating extant knowledge, identifying anomalies or shortcomings in extant theory, or merely identifying new concepts (Whetten, 1989; Sutton & Staw, 1995). A theoretical contribution should add to knowledge; it should explain why an anomaly exists as it does, and should propose solutions; it should elucidate convincingly how any proposed new concept changes understanding of the object of research (Whetten, 1989). For explanatory theory, a theoretical contribution should go beyond the epistemic script of refining or affirming an existing [general] theory through deductive application to a new context (Avgerou, 2013).

In the Discussion chapter, I argue that the key attributes of explanatory theory (Markus & Robey, 1988; Gregor, 2006) and of theoretical contributions are present in this research's proposed theoretical contributions.

4.7 Ethical Considerations

4.7.1 Introduction

Decisions made during the research process are to be continuously mediated by careful consideration of ethical obligations to research participants and to users of research results (Kvale, 2007, pp. 23-32; Bulmer, 2008; Fisher & Anushko, 2008; Silverman, 2010, pp. 152-178; Cohen, et al., 2011, pp. 75-104; Flick, 2014a, pp. 48-62). Ethical considerations incorporate a justification of the need for a research undertaking, and explains its benefits and beneficiaries. That is, when articulated, they demonstrate the principle of beneficence (Fisher & Anushko, 2008, p. 96; Cohen, et al., 2011, p. 86; Flick, 2014a, p. 50). Concomitant with the principle of beneficence, ethical considerations identify beforehand the steps to be taken to minimise, where possible, the likelihood of harm to research participants, and these steps are then implemented. This is the ethical principle of non-maleficence (Silverman, 2010, p. 156; Cohen, et al., 2011, pp. 85-86; Flick, 2014a, p. 50). A clearly articulated account of a planned and deliberately employed ethical strategy throughout the conduct of a research investigation interrelates with issues of research quality (Kvale, 2007, pp. 23-32; Bulmer, 2008; Fisher & Anushko, 2008; Silverman, 2010, pp. 152-178; Cohen, et al., 2011, pp. 75-104; Flick, 2014a, pp. 48-62), and serves as an indicator of the suitability of research findings for use (Mertens, 2014).

Key ethical considerations and how they were implemented during this research are described next.

4.7.2 Informed Consent

Research participants (here specifically, expert and elite interviewees[19]) should, as far as possible, be made aware of essential information that could influence their decision

---

[19] Whether interviewees are characterised as subjects or participants (Esterberg, 2002, p. 88; Kvale, 2007, pp. 15-19) does not contribute any additional methodological clarity.

to participate in research (Fisher & Anushko, 2008; Bulmer, 2008; Silverman, 2010, pp. 155-170; Cohen, et al., 2011, pp. 77-78; Flick, 2014a, pp. 51-52,54-57). Relevant information that may, for example, be facilitative to participants' decision-making include: a declaration of the purpose of the research, an indication of the extent and type of effort required for participation, an explanation of the potential scope of use of the research findings, clarification of what access participants will have to research findings, an outline of the confidentiality and anonymisation measures that will be employed to safeguard participants and their contributions, and disclosure of whether participants will be recorded (Kvale, 2007, pp. 23-29; Bulmer, 2008; Fisher & Anushko, 2008; Silverman, 2010, pp. 155-170; Cohen, et al., 2011, pp. 77-81; Flick, 2014a, pp. 51-52,54-57).

That is to say, when individuals agree to participate in research, their consent should be given voluntarily, and on the basis of an informed decision. This is the ethical principle of informed consent (Kvale, 2007, p. 27; Bulmer, 2008, pp. 150-151; Fisher & Anushko, 2008; Silverman, 2010, pp. 155-170; Cohen, et al., 2011, pp. 77-81; Flick, 2014a, pp. 51-52,54-57).

4.7.3 Implementing Informed Consent

Although interviewees held postgraduate and research academic degrees, it could not be overlooked that most of them operated daily within non-academic commercial environments. It seemed inappropriate then, to present them with an informed consent form written in academic jargon for their signing. Taking such an approach could have led to potential research participants' concern about the extent to which the informed consent form was a legal document, and about issues of trustworthiness, since they had no prior relationship with the researcher (Cohen, et al., 2011, pp. 80-81; Marzano, 2012).

Considering that the wrong approach could unnecessarily jeopardise gaining research access (Cohen, et al., 2011, pp. 80-81; Marzano, 2012), the decision was made to combine research access request letters (Cohen, et al., 2011, pp. 81-84; Robson, 2011, pp. 399-404) with elements of informed consent.

Specifically, research access request letters explained the nature of the research by providing a summary of the research's objective using minimal academic language, and by clarifying that it was being conducted as part of a PhD research undertaking (Cohen, et al., 2011, pp. 71-84; Marzano, 2012). Initially, letters included an explanation of the amount of time required to conduct the interview, but this was later removed and instead placed within introductory emails. Potential interviewees were made aware that the right to publish the research findings would be retained by the researcher (Kvale, 2007, p. 27; Bulmer, 2008, p. 154; Silverman, 2010, p. 156; Cohen, et al., 2011, p. 83; Mertens, 2014), but they were told that if the research findings were published in an academic article, they would be provided a copy. Research access request letters clarified that the interviewee's name and associated organisation(s) would be anonymised within any published article (Kvale, 2007, pp. 26-29; Bulmer, 2008, pp. 151-153; Fisher & Anushko, 2008; Silverman, 2010, pp. 155-156,166-167; Cohen, et al., 2011, pp. 91-92; Flick, 2014a, pp. 57-59).

Research access request letters were distributed via email, initially accompanying an introductory email, but later as the research proceeded, only after the potential research participant responded to an introductory email. Introductory emails contained a summary of the nature of the research, the time required from the interviewee, and an invitation to respond if the individual had interest in participating in the research (Kvale, 2007, p. 27; Bulmer, 2008, pp. 150-151; Fisher & Anushko, 2008; Silverman, 2010, pp. 155-170; Cohen, et al., 2011, pp. 77-84; Flick, 2014a, pp. 51-52,54-57).

It is not necessary that the process of obtaining informed consent includes the production of a signed form (Bulmer, 2008; Cohen, et al., 2011, pp. 80-81; Marzano, 2012). A response confirming the desire to participate in the research, after reading the introductory email and the details of the research access request letter, was accepted as sufficiently demonstrating that the interviewee understood the terms to which he or she had consented.

Still, prior to starting each interview, ethical considerations were reiterated in the introduction (Esterberg, 2002, pp. 102-103; Kvale, 2007, pp. 55-56; Cohen, et al., 2011, pp. 421-422). It was made clear to interviewees that they were not being asked to speak officially on behalf of their organisation, but that the responses shared would be treated as their personal statements, based on their experience, and reflective of those held by major organisational types in the networking industry. Further, interviewees were told that if they happened to mention clients, these organisations' names would be anonymised (Silverman, 2010, p. 155; Cohen, et al., 2011, pp. 91-92; Flick, 2014a, pp. 57-59). By making clear that the interview contributed towards a PhD research undertaking, the main beneficiary (at least initially) was identified, but I also explained that the research was situated within wider academic IS research on digital infrastructures (Esterberg, 2002, pp. 102-103; Kvale, 2007, pp. 55-56; Cohen, et al., 2011, pp. 80,421-422). I repeated this introductory process at the beginning of *second* interviews with interviewees (See Table 4-2). Before starting interviews, each person was given the opportunity to *not* proceed with the recorded interview, i.e. they were given the opportunity to withdraw consent (Silverman, 2010, p. 155; Cohen, et al., 2011, p. 78). Given the seniority of the individuals, they could have decided to not proceed with the interview without feeling any obligation, however, none of the interviewees raised any objection to proceeding with the recorded interview.

Though an informed consent *form* was not used, the preceding demonstrates a clear implementation of informed consent, and clarifies the capacity of participants to voluntarily consent. Samples of introductory emails and research access request letters are provided in Appendix A.

4.7.4 Confidentiality and Anonymisation

Given the seniority of interviewees within their organisations, and for some, within the networking industry, it was particularly important that careful procedures were followed to ensure as far as possible, adherence to the ethical principle of non-maleficence (Silverman, 2010, p. 156; Cohen, et al., 2011, pp. 85-86; Flick, 2014a, p. 50).

During the conduct of the research investigation, all interviewee details were stored in encrypted documents, and recordings and transcriptions in password-protected locations (Silverman, 2010, pp. 155,166-167; Flick, 2014a, pp. 57-58). A third-party business was used to transcribe recorded interviews, with the exception of one which

I transcribed (Esterberg, 2002, p. 108; Kvale, 2007, p. 95; Fielding & Thomas, 2008, p. 258). The transcription business was asked to sign a non-disclosure agreement, prior to receiving interview recordings. Where possible, I edited recordings, typically at the beginning when interviewees were likely to still provide personally identifiable information, and at the end during snowballing, before delivering them for transcription. On completion of each transcription, the business was asked to confirm, in writing, that all copies of the recording and transcriptions were deleted, including being removed from recycle bins and emails.

The findings of this research, reported in the following two chapters, exclude any potentially confidential information shared by interviewees (Kvale, 2007, pp. 24,27-28; Silverman, 2010, pp. 155-156,166-167; Cohen, et al., 2011, pp. 91-94; Flick, 2014a, p. 59; Mertens, 2014, p. 512). In accordance with the terms of anonymity explained in research access request letters, interviewee's identities have been replaced by the anonymised aliases listed in Table 4-1 and Table 4-2. One interviewee did not want his name to be anonymised (Silverman, 2010, p. 167; Marzano, 2012, pp. 447-448), but I explained that the research was dissimilar to industry reports (Kvale, 2007, p. 28), to which he was accustomed, which attribute direct quotes to individuals.

An additional anonymisation step was taken to do what I will refer to as *generify* interviewee's job titles, while still representing the capacity of their role. Initially this was not a consideration, but because online search engines have made it possible to more easily definitively identify individuals, I made the decision to generify titles. The process of what I call here generification, was that if an interviewee held multiple roles (such as a position at an SDO, and at a network operator), the least identifiable role, when searched online using the information disclosed in Table 4-1, was chosen. If the interviewee was still relatively identifiable, equivalent terms were substituted into the role's title. As an illustrative example, instead of "Vice President of IP Networks", the generified role title might be "Vice President in Networking".

One limitation to the extent of control of confidentiality and anonymity (Fisher & Anushko, 2008, p. 99; Silverman, 2010, pp. 155-156; Cohen, et al., 2011, pp. 91,93; Mertens, 2014) is that, because research access request letters were written as introduced by my PhD supervisor, he knows who the interviewees are. However, the interviewees themselves received these letters, and therefore are aware of this.

4.7.5 Other Ethical Considerations

The aforementioned strategies used to alleviate threats to internal and construct validity discussed in sections 4.3.3 and 4.3.5 were treated as an issue of ethics. The strategies were formulated in the ways described, to ensure that other IS researchers could be confident of the validity of the findings, if used to support their research (Mabry, 2008, pp. 221-223; Robson, 2011, pp. 154-159; Cohen, et al., 2011, pp. 180-200; Flick, 2014a, pp. 483-486; Yin, 2014, pp. 45-49). For the same reason, interview recordings were transcribed verbatim, and the accuracy of transcriptions was verified following initial transcription, and re-confirmed throughout the data analysis to ensure that findings were based on the most faithful account of what interviewees stated (Esterberg, 2002, pp. 107-108; Kvale, 2007, pp. 94-98; Fielding & Thomas, 2008, pp. 257-258; Cohen, et al., 2011, pp. 426-427; Flick, 2014a, pp. 388-390).

Other ethical considerations include the following. As a matter of ethics, time limits agreed when gaining access to interviewee were strictly adhered to, except in cases where the interviewee agreed to prolong discussions, or to have a follow up interview. Also, because research access request letters templates were written as introduced by my PhD supervisor, whenever a letter was distributed, a copy was provided to him. Finally, to ensure reliability (Richards, 2005, pp. 43-44; Cohen, et al., 2011, pp. 201-204; Robson, 2011, pp. 155,159; Yin, 2014, pp. 48-49), the URLs of internet references included in Chapters 5 and 6 were re-verified as reachable as of September 2nd, 2016.

4.8 Conclusion

This chapter explained the research strategy and methods used to find an answer to the research question. It explained that the research design strategy was a critical realist embedded case study of the advent of production SDN infrastructures that had two units of analysis. The first unit of analysis was of the technical details of the architecture of SDN infrastructures. The second unit of analysis was of the social objectives of the advent of SDN infrastructures entrenched in the interests of three major types of organisations, in the networking industry. Data for each unit was collected via documents and qualitative interviewing, and analysed for architectural details of SDN infrastructures, and for why the advent of SDN infrastructures occurred at a particular point in time and the sociotechnical processes of how ossified networking infrastructures were being architecturally transformed into SDN infrastructures.

The chapter also included a methodological account of how one generative mechanism sought by the research question was identified, in order to provide support for the research's credibility and methodological reliability. Analytic generalisation was identified as the generalisation argument that the thesis seeks to establish, and it was explained that the proposed theoretical contributions of this research form an explanatory theory that seeks to explain, via a process model logical structure of causation. Finally, a comprehensive account of the ethical considerations and their implementation throughout the research was provided.

5 Findings

5.1 Introduction

This chapter, and the following chapter, present the findings of the case study of the advent of SDN infrastructures, as through a critical realist theoretic lens. The proposed theoretical contributions form an explanatory theory that follows a process model logical structure of causation, to delineate a process by which architectural evolution in sociotechnically ossified digital infrastructures occurs (Markus & Robey, 1988; Gregor, 2006; Avgerou, 2013). Aside from the general critical realist theoretic framing of the findings, the explanatory theory's logical structure of causation is adopted from Archer's morphogenesis of structure (Archer, 1982; Archer, 1995), which is a process model. The final explanatory theory is thus articulated in terms of the morphogenesis of structure. As the findings are presented, the correspondences with the structural conditioning, social interaction, and structural elaboration or reproduction phases of the morphogenesis of structure (Archer, 1982; Archer, 1995), are explicitly identified. This chapter begins by explaining the advent of SDN infrastructures in terms of structural conditioning and structural elaboration in two morphogenetic cycles of structures. The Analysis chapter then identifies the generative mechanisms sought by the research question, and discusses the inconclusive outcome of structural elaboration or reproduction in one of the morphogenetic cycles.

The first morphogenetic cycle, is the morphogenesis of traditional networking infrastructures owned by network operators as sociotechnical structures, whose structural elaboration yielded SDN infrastructures (Archer, 1982; Archer, 1995, pp. 149-161,165-194; Archer, 2000). Although the plural "traditional networking infrastructures" rather than the singular "traditional networking infrastructure" is used to describe this first morphogenetic cycle, recall from the description of the case that the focus of the case study is on how, *from an architectural perspective*, production SDN infrastructures came about in spite of conditions of sociotechnical ossification. Irrespective of a particular instance of a traditional network, it features the same implementation architecture in its routers, and instantiates the same core

Internet architecture. It is the change and occasion for change in these, respectively, that constitute the first morphogenetic cycle. The plural is simply used because the networking infrastructures are owned by different network operators. Having stated this, it will be shown that these architectural changes are not inconsequential technical details. They have significant implications for elaborated networks (i.e., SDN infrastructures), and they are fundamentally interrelated with the second morphogenetic cycle.

Another point of clarification is that this morphogenetic cycle included incidents of social interaction (Archer, 1982; Archer, 1995, pp. 149-161,165-194; Archer, 2000) that led to the *innovation* that is SDN, but it is network operators' production SDN infrastructures that the analysis treats as the elaborated sociotechnical structures. The social interaction that led to SDN as an innovation, are nonetheless reported within the analytic history (Archer, 1995, pp. 149-161,165-218; Archer, 1998) of this morphogenetic cycle.

The second is the early stages of the morphogenetic cycle of the networking industry as a social structure, which may yet lead to an outcome of morphostasis or morphogenesis (Archer, 1982; Archer, 1995; Archer, 2000). Within the boundaries of the case, this social structure was delimited to the necessary internal relations (Archer, 1995, p. 173; Archer, 2000; Sayer, 2000, p. 14) of interdependence between network operators, vendors and systems integrators on the provision, use, and implementation of networking products and services. These necessary internal relations were the targets of change (Archer, 1995, pp. 165-194) that aimed to transform the social structure. The objectives of the second morphogenetic cycle were to create conditions for opening the networking industry to the entrance of new innovators, and for cost-effective replacement and interchangeability of vendors' network hardware devices.

The advent of SDN infrastructures occurred within the context of a purposeful strategy for instrumenting a change of the networking industry. Thus, the morphogenesis of the networking industry, is causally related to and features in the analytic history of the morphogenesis of traditional networking infrastructures. This chapter proceeds by first presenting an analytic history of the structural conditioning of traditional networking infrastructures as from the negative causal mode of constraint (Hartwig, 2007, pp. 57,80,458). This analytic history expands on the structural conditioning of traditional networking infrastructures exerted as sociotechnical ossification, introduced at the beginning of this thesis, with key historical antecedents of SDN infrastructures, and detailed particulars of the constraining causal force of traditional networks as sociotechnical structures.

Next, enabling, stimulating and releasing conditions (Pinkstone & Hartwig, 2007, p. 458; Hartwig, 2007, p. 80; Fleetwood, 2011) identified for the generative mechanisms that promote architectural evolution in the sociotechnically ossified traditional networks infrastructures are explained. Finally, the outcome of structural elaboration in the morphogenesis of traditional networking infrastructures, namely production SDN infrastructures, and elaborated social structure of the networking industry are explained. In presenting the findings, I include an abundant number of interviewee quotations. The intention is not to overuse these, but to provide ample evidence of structural conditioning, the enabling, stimulating and releasing conditions, and structural elaboration – giving the reader an opportunity to understand the depth of these (Gibbs, 2007, pp. 97-98; Yin, 2014, p. 205).

Importantly, recall from Chapter 3 that the morphogenesis of structure is temporally continuous, but is analytically articulated as a sequential series of morphogenetic cycles. In this chapter and the next, both the temporally specific phrase "morphogenetic cycle of" and the broader terminology of "morphogenesis of" are used interchangeably to refer to the same analytically isolated morphogenetic cycles of the aforementioned structures.

Possibly a mundane detail, but for clarity it is included here, though reporting past conditions and events, much of the chapter is written in the present perfect tense[20]. This is because the case studied was of a contemporary phenomenon about which

[20] See also (British Council, n.d.).

some conditions and events (in particular of the second morphogenetic cycle) have continued into the present (at the time of this writing).

5.2 Structural Conditioning of Traditional Networking Infrastructures

5.2.1 Key Historical Antecedents

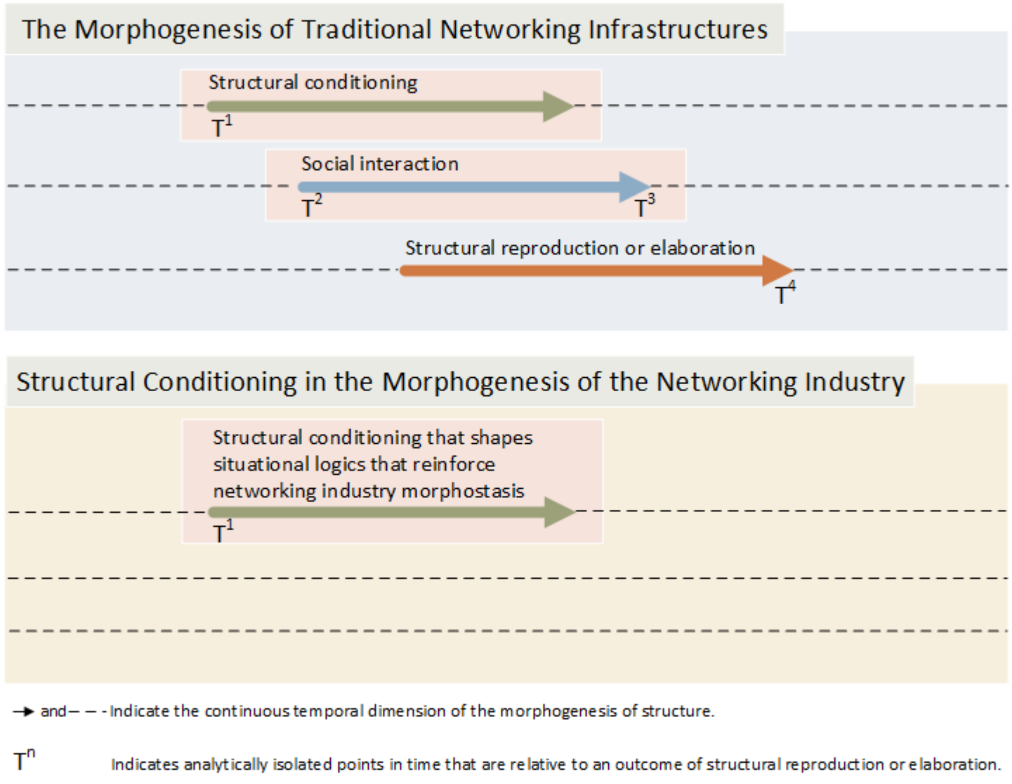

Figure 5-1 Structural Conditioning and Early Social Interaction in the Morphogenetic Cycles

As highlighted in Figure 5-1, this section details, in an analytic history, structural conditioning in the morphogenesis of traditional networking infrastructures and the networking industry[21]. It also introduces early social interactions in the morphogenetic cycle of traditional networks that led to SDN *as an innovation*, and provides an overview of early work by SDOs to promote SDN as an innovation to vendors and to network operators. Social interaction that is causal to the structural elaboration of

[21] The morphogenesis of the networking industry is only partially depicted in Figure 5-1 because it has a dependence on the morphogenesis of traditional networking infrastructures and its exposition is only fully developed by the end of the next chapter.

network operators' traditional networking infrastructures as sociotechnical structures, however, is explicated as a generative mechanism of morphogenesis in Chapter 6. The historical summary of technological antecedents that follows, does not delve into the particularities of technologies, as those technical specificities do not contribute to answering the research question. For the same reason, this section does not exhaustively enumerate technologies. Instead, key historical conditions, events and technologies are identified, and references to computer science articles that develop these in further detail are provided.

So what key historical conditions and events led to the advent of SDN infrastructures? As succinctly summarised by Interviewee **I14**, Distinguished Engineer at a large network equipment vendor and systems integrator, the difficulty of conducting academic research in networking under the technical constraints of traditional networks, was a key causal antecedent that contributed towards the advent of SDN infrastructures:

> *"One of the perspectives that drives the interest in SDN, frankly, is research. It is very hard to do research in network behaviour, when you can't change the way the network behaves. So researchers, like folks at Stanford and MIT, and Georgia Tech and Princeton and other places, were finding themselves frustrated, and so one of the drivers was to— they picked up some older work on changing the architecture and used it in a way that let them re-program the network more dynamically." –* ***Interviewee I14***

To elaborate[22], as explained at the beginning of this thesis, exploitation of the generative capacity of the Internet's five-layer core architecture, reinforced by positive network effects have led to sociotechnical ossifications in networking infrastructures. Under these conditions of ossification, it has not only become difficult to propagate

---

[22] Indeed, Archer's morphogenetic approach appropriated the word "elaborate" to indicate a theoretical construct. For clarity, where not explicitly used in relation to the morphogenetic approach in this thesis, the word "elaborate" is used without the morphogenetic connotation.

updates to the Internet's core architecture (such as with IPv6), it has become difficult and costly for computer science academics to conduct research in the area of networking when that research requires development, testing or experimentation with new networking protocols or network architectures that are fundamentally incompatible with Internet standard protocols or network architectures. Even where there has been some compatibility or interoperability, convincing network administrators or ISPs to allow these experiments to run on their networks has been a disadvantaged undertaking for academic researchers. Technical ossifications have also made it difficult to resolve known networking problems (predominantly in the area of security) that stem from network architecture that is constrained by the Internet's core architecture and its design principles – with it becoming standard networking practice to resolve these issues by violating the Internet architecture's end-to-end design principle with the introduction of *middle-boxes*[23] (Carpenter & Brim, 2002) to networking infrastructures. Complicating this, the suitability of some experimental networking protocols and network architectures for deployment in production networking infrastructures, or as recommendations for standardisation, requires evaluation on truly geographically distributed networking infrastructures, but incompatibilities with the existing core architecture of the Internet, and the risk of an experiment propagating onto the global Internet infrastructure, have been restrictive of such academic research.

Citing these aforementioned conditions as motivating factors (Chun, et al., 2003; Anderson, et al., 2005; Peterson, et al., 2006; Feamster, et al., 2014), several dedicated distributed test-bed networking infrastructures for academic research with existing and new, networking protocols and network architectures were created by various organisations. Prominent test-bed infrastructures include:

---

[23] Middle-boxes are networking equipment whose primary responsibility is to execute some network function that is not data forwarding (Carpenter & Brim, 2002). A firewall, such as Cisco's Firepower 9000 product (Cisco Systems, n.d.-b), is an example of a middle-box.

- **PlanetLab**: established by researchers from Princeton University, University of California Berkeley, and Intel Research in 2002, and provides a globally distributed test-bed (Chun, et al., 2003; Anon., n.d.; Anon., n.d.; Anderson, et al., 2005).
- **Global Environment for Network Innovations** (GENI): initiated as a project in 2006 by the National Science Foundation, and provides a distributed test-bed in the United States, (GENI, n.d.; Peterson, et al., 2006).
- **Future Internet Research and Experimentation** (FIRE): initiated as a project in 2007, funded by the European Commission and provides test-beds distributed primarily in Europe but with test-beds in Australia, USA and South Korea (FIRE, n.d.; FIRE STUDY Team, 2016).
- **Strategic Network for Smart Applications on Virtual Infrastructure** (SAVI): initiated in 2011, funded by Canada's National Sciences and Research, and provides a distributed test-bed in Canada (SAVI, n.d.; NSERC, n.d.; NSERC, 2011; Kang, et al., 2013; SAVI, 2015).

The creation of test-bed infrastructures form part of early social interactions in the morphogenetic cycle of traditional networks, that led to SDN *as an innovation*. Test-bed infrastructures have been designed to provide a generic solution for conducting diverse networking research experiments, and to isolate simultaneously executing experiments into managed, logical slices of physical hardware. As academic researchers are likely to create dissimilar, incompatible experiments, test-bed infrastructures have been engineered to be indifferent to network protocols and network architectures that run as experiments within them (Anderson, et al., 2005; Peterson, et al., 2006; Kang, et al., 2013). As such, experiments they accommodate include, clean-slate experimental networking architecture redesigns that may be entirely incompatible with traditional networks and underlying design concepts, rather than only incrementally divergent networking architectures (Anderson, et al., 2005; Peterson, et al., 2006).

Of direct relevance to the advent of SDN infrastructures, was one proposal, from researchers at AT&T, Princeton University, Carnegie Mellon University, Naval

Postgraduate School, and Reykjavik University for a conceptual clean-slate network architecture redesign called the 4D architecture (Greenberg, et al., 2005). Published in ACM SIGCOMM Computer Communication Review in October 2005[24], the proposal was received with some consternation[25] (Greenberg, et al., 2005). The authors argued that the previously mentioned networking challenges arose because existing network design couples decision-making logic for network operation (e.g., the selection of routes for data that travels through a network) with protocol logic (i.e., how routes are calculated) for interaction between network devices. They suggested a solution that reorganises the link and network layers of the Internet architecture into four layers: a *decision plane* responsible for making all network operation decisions, a *dissemination plane* responsible for connecting the decision plane to network devices, a *discovery plane* responsible for discovering network devices in the network, and a *data plane* responsible for transporting data through a networking infrastructure (Greenberg, et al., 2005). Greenberg, et al. suggested that the new architecture could be developed and tested in the GENI test-bed (Greenberg, et al., 2005).

Around the time of the proposal by Greenberg, et al., in June 2005[26] another clean-slate network architecture redesign research initiative was instantiated at Stanford University with collaborators from the University of California, Berkeley: the SANE, and Ethane projects (Casado, et al., 2006; Casado, et al., 2007; Casado, et al., 2009; National Science Foundation, TRUST, n.d.; Stanford University, SANE, n.d.; Stanford University, Ethane, n.d.). The SANE architecture was proposed as a theoretical ideal

[24] An earlier version of the proposal was published in November, 2004, In Proceedings of HotNets III (Rexford, et al., 2004).

[25] A preface for the published article, explained that the article "generated both broad consensus and wide disagreements from the reviewers," and proceeded to discuss the points of contention (Greenberg, et al., 2005).

[26] Approximate month based on the date on which the National Science Foundation grant was awarded to University of California, Berkeley. See (National Science Foundation, TRUST, n.d.).

clean-slate network architecture redesign, that similarly to the 4D architecture (but with fewer "planes"), decoupled decision-making logic for network operation from logic for transporting data through networking infrastructures, but specifically to address network security concerns (Casado, et al., 2006; Casado, 2007). Ethane, on the other hand, was proposed as a deployable, sufficiently backward compatible (relative to existing network architecture), instantiation of SANE, that included ideas from the 4D architecture, but that could be implemented in existing enterprise networking infrastructures – as opposed to only within the confines of experimental test-beds (Casado, et al., 2007; Casado, 2007).

Successful implementation in the networks of Stanford University's Computer Science department and a small business (Casado, et al., 2009), demonstrated the feasibility of, what amounted to, the super-imposition of a clean-slate network architecture redesign on top of an existing installed base of network devices and network architecture. Backward compatibility, combined with sufficiently high performance, and the ability to practically deploy the architecture in an existing production network, were the main strengths of the Ethane project (Casado, et al., 2007; Casado, 2007; Casado, et al., 2009).

Continued research premised on insights from the Ethane project, by academics at Stanford University, the University of California, Berkeley, the University of Washington, Massachusetts Institute of Technology, Princeton University and Washington University in St. Louis, and at Nicira Networks Inc., a company founded in June, 2007[27] by Martín Casado, Nick McKeown, and Scott Shenker, pioneers of the SANE and Ethane research, gave rise to two important innovations that have defined the shape of what is now commercially known as Software-Defined Networking: OpenFlow and NOX. OpenFlow was created as a protocol that leveraged pre-existing features of network devices to facilitate communication between these devices and

[27] The date of incorporation for Nicira Networks Inc. was retrieved from public company records filed at the U.S. Securities and Exchange Commission.

decision-making entities in the new clean-slate network architecture suggested by Ethane, and NOX was created as the first OpenFlow-based decision-making entity, called a *controller* (McKeown, et al., 2008; Gude, et al., 2008).

Although the formation of SDOs and standardisation processes surrounding SDN as an innovation was not the focus of this research, the role of standardisation in the advent of SDN infrastructures needs to be acknowledged. Through the efforts of the OpenFlow Consortium[28], founded in 2008, by Stanford University and the University of California, Berkeley researchers, and later the Open Networking Foundation (ONF), founded in 2011, by some of these academic researchers along with researchers from Deutsche Telekom, Facebook, Google, Microsoft, Verizon, and Yahoo!, OpenFlow-based SDN was marketed to an increasing number of network equipment vendors who added varying levels of support for the new protocol (McKeown, et al., 2008; OpenFlow Consortium, n.d.; Open Networking Foundation, n.d.-b). The backward compatibility of OpenFlow-based SDN with existing networking infrastructures gave it preferential momentum with network operators in comparison to other approaches to SDN (and prior attempts[29] to separate decision-making logic for network operation from protocol logic for interaction between network devices, or from data transmission through networking infrastructures), from the perspective of both network equipment vendors and network operators.

In addition to the work done by the ONF, other SDOs focusing on different (at times competing) aspects of SDN, such as the definition of reference architectures and software frameworks, protocol standards, information models, and the mobilisation of the network equipment vendors and network operators towards SDN adoption, have contributed towards technical refinements of SDN as an innovation. Prominent

---

[28] The OpenFlow Consortium has ceased to exist, with responsibility for OpenFlow being transferred to the Open Networking Foundation.

[29] An extended survey of predecessors, and alternative approaches to SDN can be found in (Jarraya, et al., 2014; Feamster, et al., 2014; Nunes, et al., 2014; Kreutz, et al., 2015).

SDOs with SDN standards working groups include the Telecommunication Standardization Sector of the International Telecommunication Union (ITU-T) (ITU-T, 2014), the Internet Engineering Task Force (IETF) (Boucadair & Jacquenet, 2014; Haleplidis, et al., 2015), the Institute of Electrical and Electronics Engineers (IEEE) (IEEE SDN, n.d.), and the Metro Ethernet Forum (MEF) (Metro Ethernet Forum, 2014a; Metro Ethernet Forum, 2014b; Metro Ethernet Forum, 2015). Several other SDOs have contributed to SDN-related technical innovations (See also (Kreutz, et al., 2015)).

5.2.2 Why SDN?

The preceding summarises the technical history of SDN as an innovation, and draws attention to the work led by SDOs to promote SDN as an innovation to vendors and to network operators. What it does not explain is why some network operators have shown interest in SDN, or why *at a particular point in time*, they started to show interest in SDN, especially given that earlier alternative network architecture proposals had not been well-received[30].

A quote from Interviewee **I33**, Head of a technical committee at a SDO, provided a concise high-level summary of what is expanded in the remainder of this section:

> *"Essentially, well, on the one hand, it was, in terms of research it was – there were some groups, research groups that wanted to experiment with new protocols and new mechanisms and they found that the way in which the industry was behaving was too slow for them.*
>
> *This is - maybe was the initial goal of the people that started to with that, but very soon, the people that were running data centres in cloud realised that it was ideal for them in the shape that it had four or five years ago.*

[30] Some accounts of why those proposals received low interest and deployment can be found in (Nunes, et al., 2014; Jarraya, et al., 2014; Kreutz, et al., 2015).

So, there was the combination of a few academics that were willing to make experiments and a few people in the industry that found something that it was really fitting their niche and you have the results."
– ***Interviewee I33***

Continuing the analytic history of traditional networking infrastructures' structural conditioning (Archer, 1995, pp. 149-161,165-218; Archer, 1998) as from the negative causal mode of constraint (Hartwig, 2007, p. 80), three significant conditions that were found to have impinged on (Kallinikos, 2011b) the actions of network operators, and that have contributed to their interest in SDN as an innovation, are next discussed. These are the increasing complexity and high-levels of expertise required to deliver network services to customers, inflexible networking infrastructures, and growing, uncontrollable networking infrastructure capital expenditure (Capex) and operational expenditure (Opex).

Importantly, the antecedent structural conditioning of the three feature in both morphogenetic cycles (see Figure 5-1). The overlap is unproblematic theoretically, for as Archer emphasizes, the history of antecedent structural conditioning is an analytic one (Archer, 1995, pp. 157-158,165-194). Thus, both morphogenetic cycles may include the same causal antecedent structural conditioning. In the morphogenesis of traditional networking infrastructures, antecedent structural conditioning, depicted in Figure 5-1 at analytically isolated temporal point in time T^1, have shaped circumstances that has followed through with social interaction, and structural elaboration. The same conditions of structuring, depicted at analytically isolated temporal point in time T^1 in the morphogenesis of the networking industry (Figure 5-1), have shaped circumstances that have reinforced morphostasis. In Chapter 6, I introduce structural conditioning in the morphogenesis of the networking industry, that is subsequent to the advent of SDN infrastructures.

5.2.3 Increasing Network Complexity

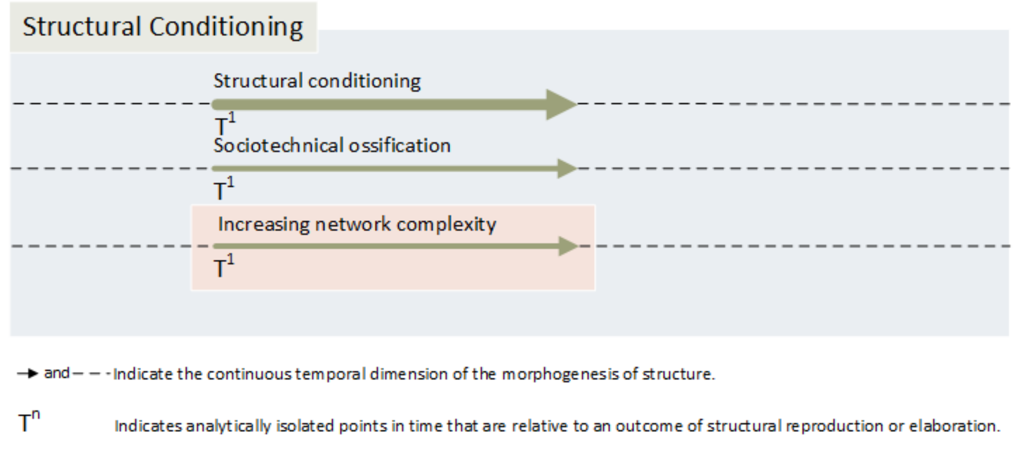

Figure 5-2 Structural Conditioning of Increasing Network Complexity

Accompanying the expansion and increasing ubiquity of the Internet (International Data Corporation, 2015; Gartner Inc., 2015a) and networks described in the Introduction and Related Literature chapters, has been an increase in networking infrastructures' complexity. Over time as the Internet and networking as a field have matured, new protocols for transmitting data through networks, identifying routes through networks, and monitoring and managing network devices have been added to Internet standard protocols at different layers of the Internet's core architecture. Several hundred Internet standard and proposed[31] standard protocols exist (Internet Engineering Task Force, 2016) – any combination of which may be active within a network operator's networking infrastructure. The use cases for networks have changed radically since the advent of the Internet. Interviewee **I36**, Product Line Manager at a large network equipment vendor and systems integrator, provided some extended commentary on this:

[31] See Housley, et al. (2011) for the distinction between the Internet standard and proposed standard maturity levels of Internet protocols. Either type of protocol may be operational within a networking infrastructure.

"Deep packet inspection of all kinds, so we have IPS (Intrusion Prevention Systems) and we have anti-virus, anti-spam … anti DOS (Denial of Service Attack) anti-everything.
You have load balancer, you have WAN (Wide Area Network) accelerator and you have gateway proxy if you want to QoS (Quality of Service) we add a new box. So the distributed architecture has become very complex - very complex to maintain…" – ***Interviewee I36***, *(Meanings of IPS, DOS, WAN, QOS added.)*

In consonance with characteristics well-articulated in the IS infrastructure literature, network operators' networking infrastructures have also followed an organic growth (Ciborra, 2000). Continued increase of core network functions (i.e., software that gives the network its infrastructural capabilities), and software applications that use these capabilities, arranged in the organic patchwork of infrastructure described, has created complexity in provisioning (i.e., creating) and managing network services offered to network operators' customers. Increased network complexity is associated with protracted service delivery times for new network services for customers, and for incremental networking infrastructure upgrades. There is a self-reinforcing complexity in which the delivery of network services requires complex automated and manual configuration of network hardware devices, and altogether perpetuates complexity in the operation of an already complex networking infrastructure. Timely provisioning of new network services has remained an elusive desire of network operators.

Using the example of provisioning virtual private networks (VPN) for customers, Interviewee **I30**, Director in Strategy in Networking at a Tier 1 ISP, for instance discussed in this relatively long quote, that impediments in inter-organisational processes for network service delivery include issues originating in overall networking infrastructure complexity:

"To give you a very concrete example today, we provide a VPN service offering and most of our major enterprise customers are usually multinational companies which means that the request for a VPN service will have to be deployed over different networks including networks that we do not operate and this usually means the establishment of discussions, negotiations with peering service providers, for example, to

> *allocate the appropriate key resource, the provider edge resource, that we need to connect the customer premises equipment that would be installed in the customer's premises that would be located for example, in Asia or America or in Africa.*
>
> *And the time it takes to actually discuss the availability of the required key resource, let alone the ability for the peering partner to actually allocate this key resource according to for example, the requirements in terms of quality of service, in terms of availability, in terms of robustness, that may have been expressed by our enterprise customer is usually quite a long time which means that the time to produce the VPN service itself and to declare that the VPN service that has been delivered is actually up and running and fully operational, usually represents several weeks if not a couple of months, depending on again, the number, the nature of functions that need to be activated, and the scope of the service itself." –* ***Interviewee I30***

Problematically, as network complexity has intensified, traditional network operation and management approaches have become less effective for network operators of large networking infrastructures. As the size of network operators' networking infrastructures and the number of user devices making use of the network has grown, previously tolerated limitations in traditional technical networking approaches to dealing with network complexity have become less tolerable. Echoing observations in IS literature on the limits of technological control (Ciborra, 2000; Kallinikos, 2011a), Interviewee **I13**, Head of Architecture in a SDO, for example commented on the unsuitability of limitations of traditional approaches to complexity, stating:

> *"The management of data communication is a huge mess, and right now, the biggest problem is that when you basically turn one of the knobs in your multitude of management protocols, you have no clue to really understand or predict what is the effect of that turning of the knob was. So it might very well be that there are so many kind of control loops integrated all together that even if you change something at one place, in the end it doesn't even matter or it breaks the whole system." –* ***Interviewee I13***

In summary, the constraining conditions of increasing networking infrastructures' complexity, aggravated by increasingly antiquated methods of technical control over that complexity, has been identified as a causal antecedent that contributed towards the advent of SDN infrastructures. Figure 5-2, thus depicts increasing networking infrastructure complexity as being part of anterior structural conditioning in the morphogenesis of traditional networking infrastructures. Increasing networking infrastructure complexity has served to reinforce morphostasis in the morphogenetic cycle of the networking industry.

5.2.4 Infrastructure Inflexibility

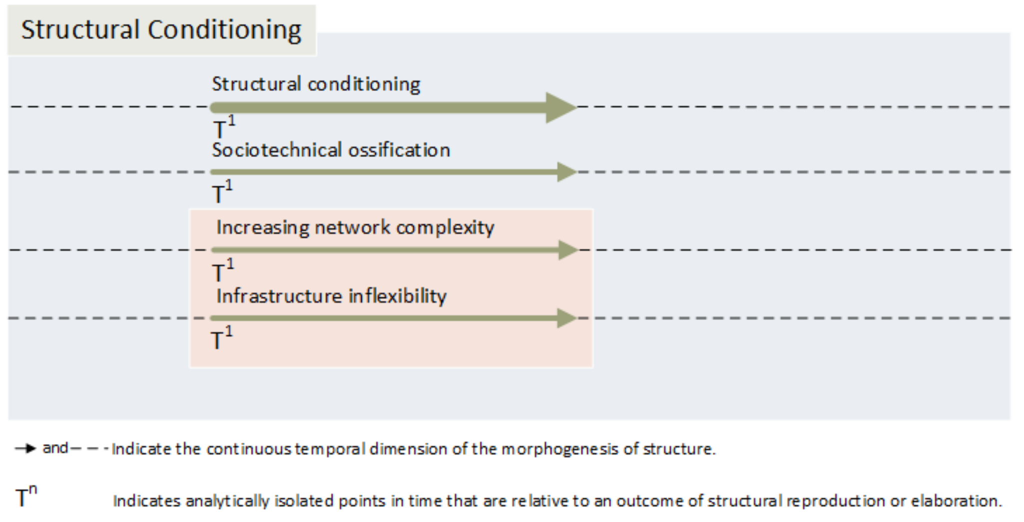

Figure 5-3 Structural Conditioning of Infrastructure Inflexibility

Concomitant with the sociotechnical ossification of networking infrastructures has been an ossification of the flexibility of these infrastructures. Infrastructure flexibility can be understood as the generative capacity (Zittrain, 2008) of a network operator's networking infrastructure relative to the creation of new network services for customers. As introduced in the related literature, network service innovations whose requirements fall outside of the generative capacity of the Internet's core architecture or that violate the de facto standard implementation architecture of network devices, are resisted. Still, there has been a long-running desire by network operators for greater degrees of configurability, and repurposing of their networking infrastructures across diverse existing and future networking infrastructure use cases. Sociotechnical ossification of traditional networks' architecture that constrain innovation activities of network operators has been another contributor to their interest in SDN as an innovation.

On evaluating characteristics of SDN's backward-compatible network architecture redesign, network operators have reasoned that SDN as an innovation is facilitative of considerably greater infrastructure flexibility than has been previously possible through other technical approaches. Quoting, Interviewee **I14**, Distinguished Engineer at a large network equipment vendor and systems integrator:

> *"I mean, when we built - one of the start-ups I have worked with - built a GGSN (General Packet Radio Service Support Node) and then we found that we had the wrong ratio between the processor power needed for the control for mobile, because mobile needs a very - a lot of control, and the data plane capacity, when we went to reuse it in a different context.*
> *So this versatility appeals both to vendors who want to build versatile products and to operators who want more flexibility in how they build and operate their networks."* – ***Interviewee I14***, *(Meaning of GGSN added.)*

As per Figure 5-3, infrastructure inflexibility is of anterior structural conditioning in the morphogenesis of traditional networking infrastructures and has served to reinforce morphostasis in the morphogenetic cycle of the networking industry.

5.2.5 Growing, Uncontrollable Capex and Opex

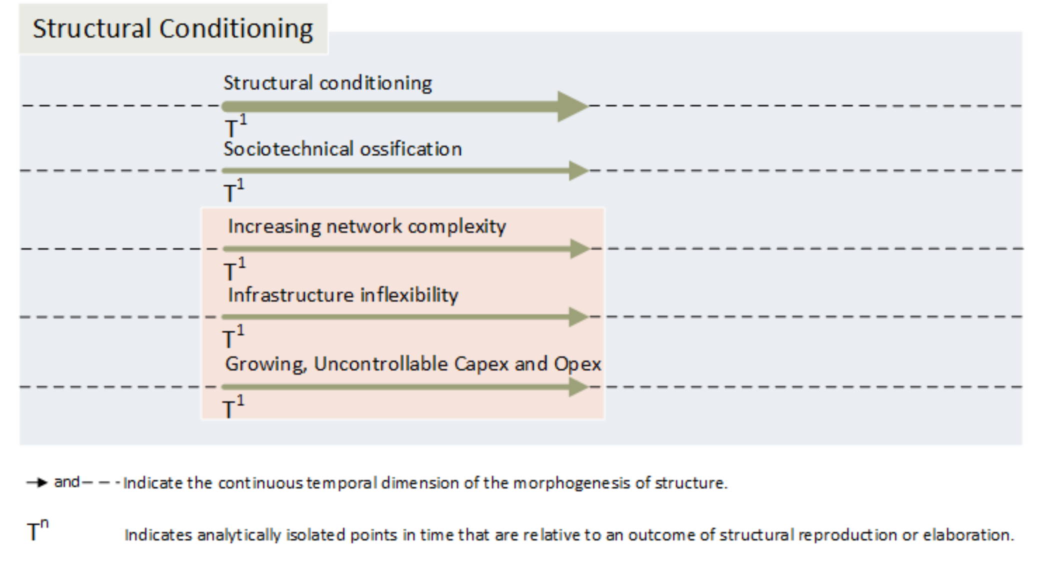

Figure 5-4 Structural Conditioning of Growing, Uncontrollable Capex and Opex

Exploitation of the generative capacity of the Internet's core architecture to realise new use cases for networking infrastructures, has led to unintended economic undesirables for network operators – in particular for large ISPs. Network operator customers of ISP network operators, such as mobile virtual network operators (MVNO), lease ISP's network capacity and via recursivity and upward flexibility of digital infrastructures, lease or repackage their allocated portion of the network capacity to their own customers. But the vast enrolment of Internet users by tenants of ISP networks (International Data Corporation, 2015; Gartner Inc., 2015a), has placed infrastructure investment demands on ISPs, which have been difficult to recoup. More than this, ISPs have been cut off from the economic value generated by their customers. They have been contending with increasing *over the top* traffic (i.e., traffic generated by their customers), and an inability to derive economic value correspondent with that increased networking infrastructure usage. Summarising this predicament, Interviewee **I11**, Vice President in Networking at a Tier 1 ISP, and Interviewee **I31**, Researcher in SDN, for example explained:

> *"I mean, we ourselves see between 60 and 70% year on year traffic growth. That's tremendously expensive to keep up with, particularly as the end of the day your Internet bill is not getting larger, right. So this is*

a situation where the - we have to do a 60 or 70% increase in traffic without actually changing the amount of money that we take in, and ultimately that's going to cause a problem." – ***Interviewee I11***

"And so, the telco's are looking at this and saying, "Wait a minute, what is our role in all of this?" because we are gradually being pushed down to being a connectivity provider and then on top of that, a lot of what we provide, we get no money for it, because of these so called, over the top services." – ***Interviewee I31***

Net neutrality rulings in the United States and European Union, have further interfered with ISPs' option of recouping costs through preferential treatment of customer network traffic (European Commission, 2015; Federal Communications Commission, 2015a)[32]. So large ISP network operators that have been disbursing immense capital and operational expenditures, building physical infrastructures, and purchasing network devices and services from vendors and systems integrators in order to support rising over the top network traffic, have developed interest in methods by which any of these costs can be reduced. Pragmatic limitations on the amount by which building physical infrastructure costs (such as fibre optics cabling) can be reduced, have contributed to ISP network operators' interests in reducing vendor and systems integration related costs. Operators of large networking infrastructures that do not own such physical infrastructures have expressed similar interest in reducing vendor and systems integration related costs. Some extended commentary is provided in this quote from Interviewee **I20**, Researcher in 5G networking:

"…the major cost is hiring the people to dig holes in the ground in the first place - those costs associated with the maintenance of networks, the running of networks are phenomenal and are often not grasped by people.

…

---

[32] The detailed USA and EU net neutrality rulings can be found in (European Parliament, Council of the European Union, 2015; Federal Communications Commission, 2015)

So finding ways to improve without having to send people out to replace every cell tower, every router, to dig up every cable, this is incredibly appealing to anyone that is managing a physical network." – ***Interviewee I20***

The preceding quote from Interviewee **I20** conveys a *projected* conclusion that has been made by several large network operators: networking infrastructures that have been architecturally transformed towards SDN's network architecture can be repurposed without complete replacement to achieve more favourable Capex and Opex. On evaluating characteristics of SDN's network architecture redesign, network operators have reasoned that the architecture is facilitative of cost-effective replacement and interchangeability of vendors' network devices in lieu of reducing physical infrastructure costs.

Notable here is that this desire by network operators is causally related to the second morphogenetic cycle (of the networking industry). While network complexity and the desire for flexible infrastructures may be somewhat addressed by projects that are entirely internal to network operators, the desire for lower Capex and Opex has brought into confrontation network operators and network equipment vendors. For lower network operator Capex and Opex, vendors must agree consistently to financial arrangements that are favourable to network operators[33]. Given the implausibility of that, network operators, and SDOs have seen the architectural transformation of traditional networks into SDN infrastructures as a strategy for instrumenting a broad change of the networking industry.

The de facto standard implementation architecture of routers co-locates forwarding and control planes inside proprietary network hardware devices, whose features and innovation cycles are decided by vendors. The value is with the hardware, over which vendors hold architectural control. Consequently, vendors have had significant control

---

[33] This is not to say that network operators cannot independently find ways to reduce expenditure.

over some economic activities associated with network operators' introduction of new networking capabilities into their infrastructures. That is, the relationship between the forwarding and control planes within the sociotechnically ossified de facto standard implementation architecture of routers has served as an architectural point of control for vendors (Woodard, 2008). As well, network operators' ability to introduce new customer network services has been dependent on vendors' product innovation cycles. As a result, the objectives of the second morphogenetic cycle were to create conditions for opening the networking industry to the entrance of new innovators, and for cost-effective replacement and interchangeability of vendors' network hardware devices. In other words, as depicted in Figure 5-4, growing, uncontrollable Capex and Opex, features in anterior structural conditioning in the morphogenesis of traditional networking infrastructures and has served to reinforce morphostasis in the morphogenetic cycle of the networking industry.

Summarising, with caution, the desire of network operators to reduce networking infrastructure costs through cross-vendor substitutability of network products as contributing to their interest in SDN for production networking infrastructures, Interviewee **I11**, Vice President in Networking at a Tier 1 ISP, stated:

> *"And it's a hope and I should have to say it's an unproven hope at this point that by saying let's make a common control plane management platform and really at this stage I'd say a lot of SDN's savings come from being a very fancy provisioning platform, really so it saves people costs from that point of view, but it requires a big capital expenditure to get to the point that you achieve those operational cost savings. So that's sort of where the networking-centric industry is coming from.*
>
> *'I am under an awfully large amount of pressure here, my margins are risible, I need to do something about that, and I need to look at every aspect of how I do everything,' and that's where NFV comes in, that's where SDN comes in and so on."* – ***Interviewee I11***

5.3 Catalysts of the Advent of Production SDN Infrastructures

5.3.1 Introduction

> *"So, Software-Defined Networking is arguably a twenty to twenty-five-year-old insight that has taken this long to find applicability."* – ***Interviewee I14, Distinguished Engineer at a large network equipment vendor and systems integrator***

Challenges of the increasing complexity and high-levels of expertise required to deliver network services to customers, inflexible networking infrastructures, and growing, uncontrollable networking infrastructure Capex and Opex are three significant conditions that have introduced circumstances of constraint (Archer, 1995, pp. 149-161,165-218; Archer, 1998) to network operators, and that have contributed to their interest in implementing SDN infrastructures. But on the positive form of causality, other social and technical factors were found to have served as enabling conditions (Pinkstone & Hartwig, 2007, p. 458; Hartwig, 2007, pp. 57-60,80; Fleetwood, 2011) of network operators' implementation of production SDN infrastructures.

Familiarity with, and insights from preceding technologies that separated decision-making logic for network operation, from logic for transporting data through a networking infrastructure is one enabling condition that has helped to catalyse the advent of production SDN infrastructures. Technical achievability combined with strong industry interest from multiple network operators, and some network equipment vendors, has been another enabling condition.

5.3.2 Conceptual Familiarity with Preceding Technologies

SDN as an innovation, includes insights from pre-existing telecommunications network technologies, and this has provided network operators (and vendors) conceptual familiarity with SDN. Conceptual familiarity with historical technologies that separated control, and forwarding has facilitated receptiveness of network operators to SDN as an innovation.

Speaking on conceptual familiarity with specific predecessors, Interviewee **I29**, CTO at a large network equipment vendor with SDN products, for example, explained:

> *"… it started out as separation of control and data planes and that's not a new idea. For example, something as old as SS7 (Signalling System No. 7) separates control and data, right this is an old control network… That's how, you know, the Telco systems work, right. It was SDN-esque because it separated the control and data. They had ForCES (Forwarding and Control Element Separation)… It's not a necessarily new idea." –* ***Interviewee I29****, (Meanings of SS7, and ForCES added.)*

However, the major objection to the 4D architecture proposal that was noted in its preface (Greenberg, et al., 2005), from which SDN as an innovation borrows architectural insights (Casado, et al., 2007; Casado, 2007), was that there had been several prior implementations of aspects of the proposal, some of which failed to contribute successfully to the fulfilment of their objectives. Given the failures of several conceptually similar predecessor technologies to either gain industry acceptance or to contribute successfully to the fulfilment of their objectives, the enabling condition of conceptual familiarity alone does not explain why these failures have been somewhat overlooked by network operators that have been implementing production SDN infrastructures.

5.3.3 Technical Achievability

Complementing conceptual familiarity has been the availability of proven facilitating technologies for realising production SDN infrastructures, and the commercial availability of production-strength SDN software.

In the interim between aforementioned early approaches for separating decision-making logic from data transmission logic in telecommunications networks such as SS7, early test-bed infrastructures such as PlanetLab in 2002, the 4D architecture proposal, and the first commercial SDN offerings for production networks such as Nicira's Network Virtualization Platform (now VMware NSX following VMware's acquisition of Nicira (VMware Inc., 2012; VMware Inc., n.d.-a)), computing as a field has undergone significant advancements, that has culminated most recently in cloud computing as has been exemplified by public cloud computing infrastructures offered by Amazon Web Services launched in 2006 (Amazon Web Services, n.d.-a), and Microsoft Azure launched in 2010 (Hauger, 2010). Network operators noted that mature, tested solutions from cloud computing to problems of performance, and efficient infrastructure utilisation could be used to circumvent challenges of achieving acceptable network performance, in network architectures with a decision-making controller entity (such as NOX or VMware NSX).

Some quotes from Interviewee **I32**, Researcher in SDN, and Interviewee **I37**, Deputy Head of R&D at a technology firm with networking products, provide details of this:

> *"So, of course, we could not have invented SDN, 20 years ago, because it was impossible then to run a network with that centralised approach. Now, in 2015, this is possible, right.*
>
> *…*
>
> *And on the other side we have now the possibility to bring this controller into more powerful servers and also into these big cloud infrastructures that we have for virtualised function and for controlling the network..."*
> – ***Interviewee I32***

> *"…and there is – has been a very important argument with virtualisation. You can implement virtual networks. You can make virtual topologies. You can multiply the number of users, isolated users existing on the same physical network. Those are key advantages of SDN. So from a point of view of the business, this is one of the main drivers behind it."* – ***Interviewee I37***

High-performance, low-cost commodity hardware has been widely utilised to build very sophisticated cloud computing infrastructures. Network operators have determined that the same can be used to implement production SDN infrastructures. Some network operators such as Google and Facebook have taken advantage of the availability of commodity network device hardware components to build SDN capable network hardware devices – bypassing the network hardware device innovation cycles and associated costs of network equipment vendors (Open Compute Project, n.d.).

Another contributor to technical achievability has been the commercial availability of standardised SDN protocols (such as OpenFlow) and decision-making controller entities (such as VMware NSX and OpenDaylight), that are performant and reliable enough for deployment in network operators' production networking infrastructures (VMware Inc., n.d.-a; OpenDaylight Foundation, n.d.; Open Networking Foundation, n.d.-e; Open Networking Foundation, n.d.-c). The work led by SDOs, such as the ONF to secure commitment from vendors to release OpenFlow-compliant network devices (Open Networking Foundation, n.d.-d), has been a notable contributor towards technical achievability. Exploitation of readily available free SDN software components has enabled network operators to gain infrastructure operational experience in a substantive rather than conceptual manner (see also (Open Networking Foundation, n.d.-a; Open Source SDN, n.d.)). Interviewee **I33**, Head of a technical committee at a SDO, for instance, elaborated:

> *"First, as I said that they came in a moment in which technology was able to deliver a reasonable performance at a reasonable price. It's not something that you can only run in a lab just for demonstrating that a certain property or theorem or whatever holds. It's something that you can run with essentially almost off-the-shelf components. So it's achievable and affordable." –* ***Interviewee I33***

Technical achievability complements the enabling condition of conceptual similarity between the innovation of SDN and preceding technologies that separated decision-making logic from data transmission logic, by contributing realisability to network operators' interest in, and conceptual familiarity with SDN. It is a significant enabling condition for the generative mechanisms that promote architectural evolution in a sociotechnically ossified context, out of which have come production SDN infrastructures. Along with the enabling conditions found, was one stimulating condition of the generative mechanisms (Pinkstone & Hartwig, 2007, p. 458; Hartwig, 2007, pp. 57-60,80; Fleetwood, 2011).

5.3.4 A Radical Change of Networking Architecture Required

The recognition by network operators that patching infrastructures is not a general solution for accommodating new customer network services, or for networking infrastructure evolution is a key stimulating condition of the generative mechanisms whose operation were causal to the advent of SDN infrastructures.

A major trigger of network operators' implementation of production SDN infrastructures, was found to have been their realisation that prevalent formal (such as introducing new networking protocols) and informal processes (i.e., internal to network operators) of extending networking infrastructure capabilities have become generally unsustainable as the type of network complexity changes. The efforts to create new protocols to solve networking problems, along with the organic internal growth of network operators' networking infrastructures driven by the creation of new network services for customers and by networking infrastructure upgrades, can be understood as a generic *methodology of patching the infrastructure*. Technical achievability has contributed realisability to network operators' interest in, and conceptual familiarity with the innovation of SDN, but the unsustainability of a methodology of patching infrastructures has been a substratum of the constraining structural conditioning, and enabling conditions out of which the advent of SDN infrastructures occurred. Under these conditions, network operators have reasoned that the transformation of their traditional networking infrastructures towards SDN infrastructures is *plausibly* a more general solution than a methodology of patching the infrastructure.

Quotes from Interviewee **I11**, Vice President in Networking at a Tier 1 ISP and Interviewee **I36**, Product Line Manager at a large network equipment vendor and systems integrator, provide some perspective on this:

> *"We can't make sticking fibre in the ground cheaper, because that costs what it costs. We can continue to put pressure on our vendors to make turning up a wavelength on that fibre once it's built, cheaper, that's one area, but there's limits to that as well. So is there a way that we can fundamentally rethink how networks are built?"* – ***Interviewee I11***

> *"Especially people like you know, the Amazons, the Googles of this world when you see, I mean, they had to deploy at that time 20 - 30,000 VMs (Virtual Machines) per day. I mean you can't - you have to radically change the way you see your IT overall but also your network.*

…

So they have [sic] to radically change the architecture and to make the system much more programmable and not configurable, because today this is where we are, we configure networks." – ***Interviewee I36***, *(Meaning of VM added.)*

The answer to the question of why some network operators started to show interest in SDN as an innovation at a particular point in time, and to implement production SDN infrastructures, is that they recognised that in response to constraining structural conditioning of traditional networks, a radical rather than incremental change in their networking infrastructures, that was capable of diminishing some structuring conditions of constraint was required, and they deemed the innovation of SDN as being facilitative of the necessary type of change. This change is facilitated through network architectural evolution – an inevitability of implementing production SDN infrastructures, and the substance of the morphogenesis of traditional networking infrastructures as sociotechnical structures.

Because vendors held some architectural control in network operators' networking infrastructures that resisted the desired network architectural evolution, SDN as an innovation was appropriated from academic research by network operators and SDOs, to play a role in changing the networking industry as a social structure. Specifically, SDN as an innovation, provided a means by which to circumvent some of vendors' architectural control in network operators' infrastructures that reinforced structural conditions of constraint unfavourable to network operators, through a type of architectural evolution in existing traditional networks that simultaneously facilitated cost-effective replacement and interchangeability of vendors' network hardware devices, and that resisted vendors' desires to regain the lost architectural control.

The explanation of how architectural evolution in sociotechnically ossified networking infrastructures occurred, thus comes out of interrelated morphogenetic cycles of traditional networking infrastructures and of the networking industry. The details of the elaborated sociotechnical structure, namely SDN infrastructure, that is the outcome of this type of architectural evolution are introduced after the following explanation of a releasing condition of the generative mechanisms sought by the research question.

5.3.5 Releasing Condition

An important releasing condition (Pinkstone & Hartwig, 2007; Fleetwood, 2011) was the work accomplished by the OpenFlow Consortium, and later the Open Networking Foundation to get vendors to make their network hardware devices SDN-compliant (Naous, et al., 2008; McKeown, et al., 2008; OpenFlow Consortium, n.d.; Open Networking Foundation, n.d.-b). This releasing condition reinforced the enabling condition of technical achievability. Early deployments of SDN infrastructures followed the work to add OpenFlow support to commercially available network hardware devices. OpenFlow-based SDN, for example was deployed, to varying degrees, by companies such as AT&T, eBay, NTT Communications, Rackspace and Google (Nicira Networks Inc., 2012a; Jain, et al., 2013; AT&T, 2013). The releasing condition is also rooted in the inherent backward compatibility of the early OpenFlow-based SDN network architecture (Casado, et al., 2007; Casado, 2007; Casado, et al., 2009).

5.4 Structural Elaboration: Software-Defined Networks Explained

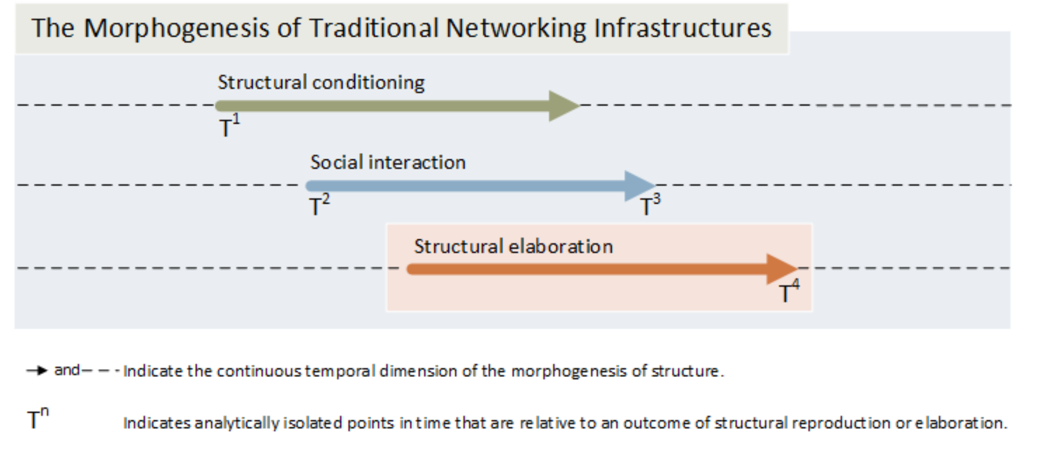

Figure 5-5 Structural Elaboration of Traditional Networking Infrastructures

5.4.1 Introduction

The preceding sections presented the structural conditioning phase of the morphogenesis of traditional infrastructures and the morphogenesis of the networking industry, preliminarily introduced early aspects of the social interaction phase of the morphogenesis of traditional infrastructures, and identified enabling, stimulating and releasing conditions for the generative mechanisms that promote architectural evolution in sociotechnically ossified traditional networks. As indicated in Figure 5-5, this section presents from an architectural perspective, the outcome of the structural elaboration of traditional networks – SDN infrastructures. It also introduces how the morphogenesis of the networking industry, which is in early stages and may yet lead to morphostasis or morphogenesis, is interrelated with the morphogenesis of traditional networking infrastructures.

Treating networking infrastructures as structures from a critical realist perspective, requires that the ontology of the elaborated structures is explained. As explained in Chapter 3, this research subscribed to a formulation of ontology in which there is a unity of structure, properties and generative mechanisms, such that the three emerge simultaneously, and none is more primary than any of the others (Fleetwood, 2009). This section preliminarily introduces, without theoretical treatment, the structure and properties of SDN infrastructures. In Chapter 6 these are formalised with critical realism, and generative mechanisms instantiated with SDN infrastructures are identified. Note that the generative mechanisms instantiated with SDN infrastructures are not synonymous with those that promote architectural evolution in traditional networks towards SDN infrastructures. But they will be identified because they feature causally in a pertinent architectural evolution that is subsequent to the morphogenesis of traditional networking infrastructures and an objective of this morphogenetic cycle. Findings necessary for identifying those generative mechanisms are included in this section.

As alluded to in Chapter 1, the elaboration of traditional networks as sociotechnical structures, involves exploitation of their existing digital materiality and upward flexibility to evolve the underlying network architecture towards the architecture prescribed by the innovation of SDN. Therefore, as part of understanding the ontology of SDN infrastructures, it is necessary to understand the role played by digital materiality in its advent. As such, pertinent details are included in this chapter, but these are explicated in Chapter 6.

Lastly, since there are multiple commercial SDN networking products, and varying styles of implementing production SDN infrastructures by network operators, the aim of this section is to identify the intransitives of the ontology of SDN infrastructures. The strategy for doing so is via explanation of the architecture of SDN infrastructures. To explain from an architectural perspective SDN infrastructures as structurally elaborated traditional networks, the differences between the architecture of traditional networks and SDN infrastructures are identified.

5.4.2 Understanding the Technical Objective of SDN

As succinctly captured by, Interviewee **I14**, Distinguished Engineer at a large network equipment vendor and systems integrator, SDN as an innovation facilitates in networking infrastructures, the ability to *apply* software that is not coupled to network hardware devices (here, routers), to the network:

> *"So what do we mean by it? We are talking about the ability to apply software that is not coupled with hardware, to the network." –* ***Interviewee I14***

More accurately, it is the application of the *outcome of some execution of external software*, to the router. In the most basic formulation, externally executed software determines network routes and applies routing information to a router, instead of the router's internal control function performing this responsibility via standardised routing protocols. More generally than route computation, the technical achievement of SDN is the ability to *introduce capabilities* to networking hardware using software that is not coupled to the hardware. The methods applied by the software may be appropriated from computer science problem domains not traditionally related to networking. Continuing with Interviewee **I14**:

> *"The idea being that software machinery written for a problem that is not tied specifically to network handling, can drive the network behaviour."* – ***Interviewee I14***

This is the overarching technical objective of SDN as an innovation. Traditional networks that have been transformed into SDN infrastructures feature this capability.

The ability to introduce capabilities to networking hardware using software that is not coupled to the hardware is facilitated by a number of architectural features. These architectural features can be understood from what happens to the control function of routers as part of the architectural evolution that transforms traditional networking infrastructures into SDN infrastructures. Importantly, note that this is architectural evolution surrounding routers' implementation architecture.

5.4.3 Decoupling of the Forwarding and Control Planes

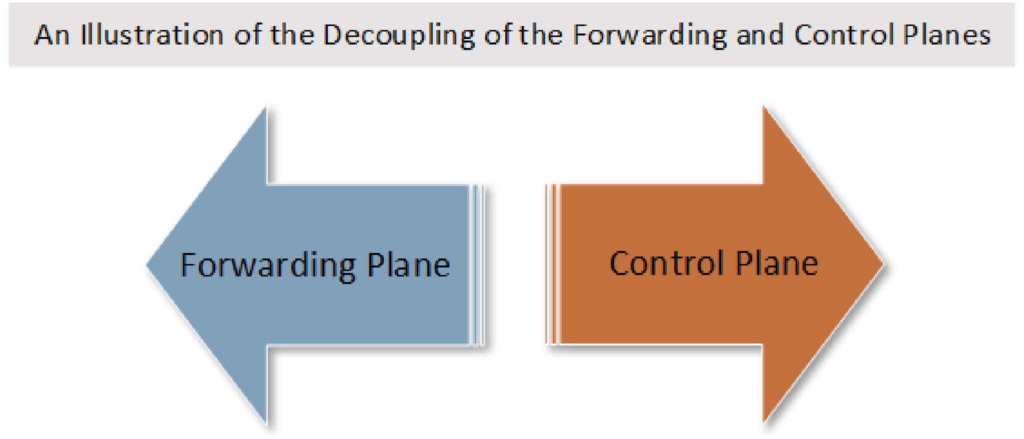

Figure 5-6 Decoupling of the Forwarding and Control Planes

The ability to apply external software to networking devices is partially facilitated by an architectural approach that decouples most or all of the control plane from its colocation in a router's hardware with the forwarding plane. The change in the relationship is visually illustrated in Figure 5-6. As explained in Chapter 2, the functions of these planes have traditionally defined the main responsibilities of routers and have been co-located within a single device. Note that this decoupling of the control and forwarding planes has been foreshadowed in the analytic history presented of the 4D architecture, SANE, Ethane, and early telecommunications networking approaches that decoupled, to differing degrees of granularity, decision-making logic for network operation from protocol logic for interaction between network devices, or from data transmission through networking infrastructures. The objective of decoupling the control and forwarding planes is to decouple responsibilities of network control (i.e., the distributed control function) from the responsibilities for data transmission.

Routers' forwarding and control functions are logically stratified into forwarding and control planes. However, it is the *physical* co-location of these planes in hardware, and not the logical separation of the functions, that is the object of this architectural approach.

5.4.4 Aggregation and Centralisation of Control Planes

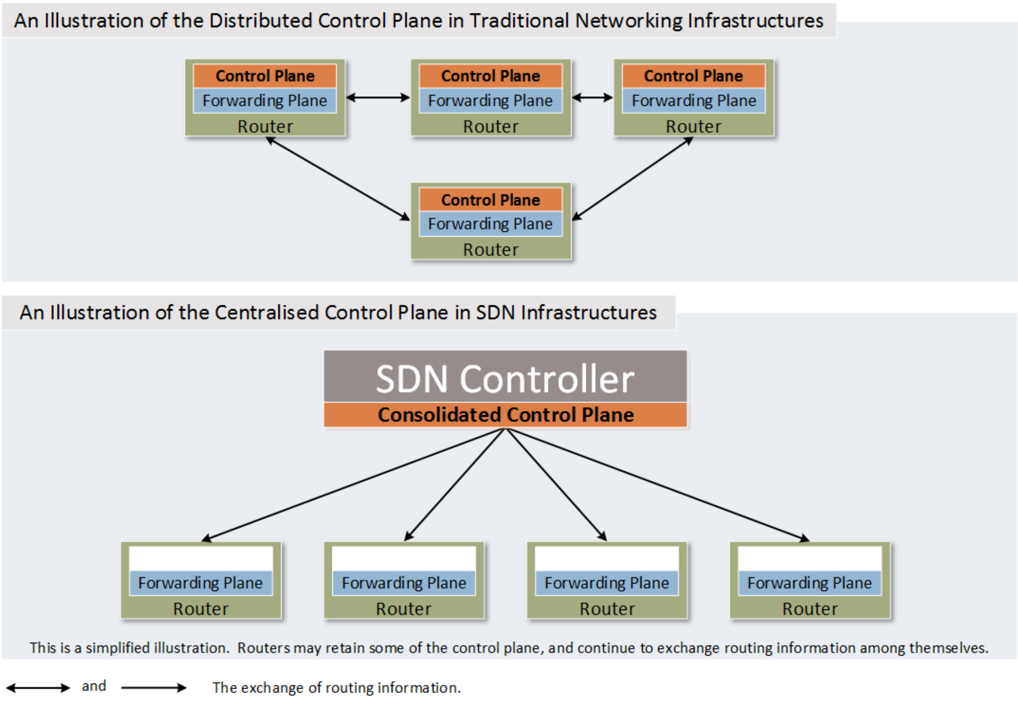

Figure 5-7 Aggregation and Centralisation of the Control Plane in SDN Infrastructures

> *"Number two is, logically centralised control - and this is very different from, 'We're splitting the control plane from the data plane, commoditising the hardware and go along our merry way'.*
> *What this means is that we are going to exploit a domain view of the network for those capabilities where it makes sense, for instance, selection of paths, determination of the aggregate … bandwidth for the services that an operator wishes to support, the ability to monitor services versus hardware."* – ***Interviewee I22, Senior Director at a network equipment vendor and systems integrator***

The control function of routers encapsulates the network *intelligence* needed to make routing decisions. As per section 2.4.4, these decisions are made on the basis of a distributed model, where each device acts in a somewhat autonomous manner, determining routes through the network via the use of distributed routing protocols. Thus, there is a logically distributed *network* control plane that is a conceptual aggregate of the control planes of all individual routers in the network (see the illustration at the top of Figure 5-7). In SDN infrastructures, there is a consolidation of routers' control plane into a logically centralised *software* entity that has a complete view of the network (illustrated in the lower diagram of Figure 5-7).

That is, control plane consolidation permits the centralised software entity to determine network behaviour without dependence on distributed routing protocols, and to act as a network control plane. This software entity, which is physically located separately from routers in the network, is called a *SDN controller*. As with the decoupling of forwarding and control planes, section 5.2.1 foreshadowed the concept of an SDN controller, referring to the decision, dissemination and discovery planes of the 4D architecture, and decision-making entities such as NOX (the first Open-Flow based SDN controller). In its basic formulation, a SDN controller applies routing information to the network's routers (i.e., application of the outcome of some execution of external software to the router).

An important corollary of the decoupling of the forwarding and control planes combined with consolidation of routers' control planes into a SDN controller, to which I return later, is that the composite functionality of routers is split into a hardware-based forwarding plane, and a software-based network control plane.

5.4.5 Programmability and Network Abstraction

SDN infrastructures are programmable.

> *"Another key aspect of SDN is about providing a wide number of abstractions which could be used by users or third parties or the operator themselves via application programming interfaces in order to program and control the function and the services of the network."* – ***Interviewee I24, Director at a SDO***

Routers' control planes are aggregated into a network control plane residing within an SDN controller and SDN controllers offer application programming interfaces (API) for making changes to the network, to client applications. Classically, these client applications are network management or infrastructure orchestration software, but may also be interacting SDN controllers or other software. Effectively, an abstraction of the physical network is offered as a programmable resource to client applications. Programmability in SDN infrastructures is not the same as the automation of manual configuration of individual routers. It is, conceptually distinct from automation and from orchestration. Commenting on the difference, Interviewee **I11**, Vice President in Networking at a Tier 1 ISP, said:

> *"So we had a lot of people ask what's the difference between automation and programmability and why are they different.*
> *To me, programmability says, I would like to be given the levers to be able to change some state or some configuration environment.*
> *Automation says, I would like a set of procedures that move those levers in reaction to my requirements, to put it very simplistically.*
> *So, programmability – and it's probably an unfortunate way to put it – programmability is just giving you the weapons. Automation is then fighting the war."* – ***Interviewee I11***

A major objective of programmability is the facilitation of increased dynamism in networking infrastructures – meaning that it becomes easier than in traditional networks to create, modify or remove network capabilities, configurations, or logical demarcations in SDN infrastructures. What constitutes a network can be *entirely programmatically defined*.

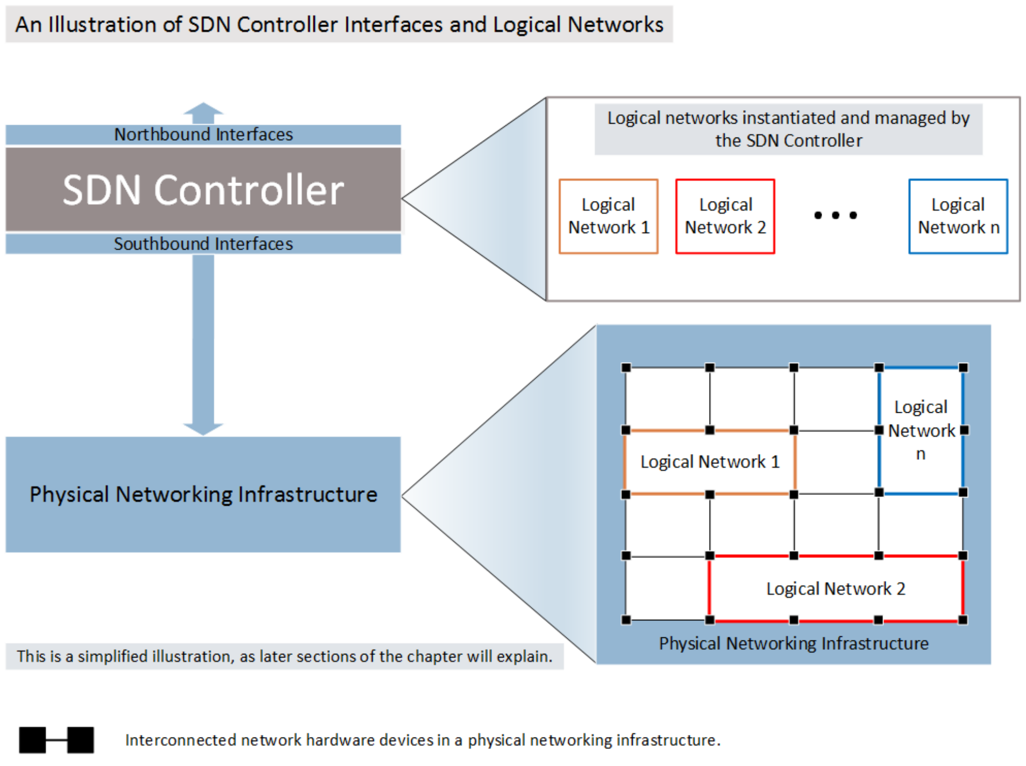

Figure 5-8 Programmability and Network Abstraction in SDN Infrastructures

One way to grasp the meaning of programmatic definition of network constitution is to understand that an SDN controller can decide which physical network device participates in the network. Traditional networks handle physical network device participation through manual command line interface (CLI) configuration of devices, or through network management systems, and the early SANE and Ethane predecessors of SDN infrastructures could exclude routers for security reasons. More than physical device network participation, the allocations of network capacity to network operators' customers are themselves *logical networks* from the perspective of customers. In SDN infrastructures, these logical networks are instantiated programmatically; they exist at a higher level of abstraction than physical devices and are managed as logical abstractions within the SDN controller[34] (see Figure 5-8).

[34] Nonetheless, these networks may include physical devices that are dedicated to a customer or to the network created.

5.4.6 Openness

Critical to SDN infrastructures is openness – open standardisation processes, open APIs, and the use of community-curated open source code. Open standardisation processes of SDOs, ensure diverse SDO membership and participation, and ensure that specifications, guidance documents and the right to implement these are available with minimal restriction. Standardised open APIs ensure that the APIs offered by an SDN controller to applications, typically called *northbound interfaces* (see Figure 5-8), is uniform and is defined by an open community of contributors, as opposed to being the proprietary offering of a vendor. It also ensures that underlying physical routers, to which the SDN controller communicates, adhere to a common standardised set of APIs, typically called *southbound interfaces* (see Figure 5-8), which also are defined by an open community of contributors and not solely by the router's vendor – as has traditionally been the case.

Quoting for instance, Interviewee **I33**, Head of a technical committee at a SDO, who provided a rationale for maintaining open northbound and southbound interfaces:

> *"And in general, communications are open by definition. I mean, IP, the IP protocol is open. You know how to implement the IP protocol. You have not to pay anyone for implementing it, and once you implement it you can connect to anyone else that is using the IP.*
> *And that applies for any kind of interconnection.*
> *…*
> *On the other hand, in SDN when you separate control and forwarding, the idea is that you require the connection - because it's a connection, it's a communication,* ***before they were inside the same box, now it's become a connection*** *- and so it's open as well.*
> *…*
> ***It's because you have separated them and there is a connection. Connections by definition are open.*** *So that is the idea behind that." –* ***Interviewee I33****, (Bold emphasis added.)*

It has been the responsibility of SDOs to put in place standardisation processes, to safeguard the openness of SDN controller northbound and southbound interfaces and to determine their emphasis on open source. Openness, then, though manifested in technical implementations in SDN infrastructures, is socially established and maintained. It is a key facilitator of the structuring conditions of opening the networking industry to the entrance of new innovators, and for cost-effective replacement and interchangeability of vendors' network hardware devices that a structural elaboration outcome in the morphogenesis of the networking industry may create. The following relatively long quote from Interviewee **I30**, Director in Strategy in Networking at a Tier 1 ISP, summarises the key rationale for openness in SDN infrastructures:

> *"And I think this is really a key fact which is probably further justified by the fact that as service providers, we absolutely need standard data models for us to be technology agnostic as much as possible. That is, regardless of the underlying networking technology, for example, in our networks we use Alcatel routers, we use Cisco routers and we use Juniper routers and all of these router technologies have their own specifics, not only in terms of configuration tasks but also in terms of supporting specific functions.*
>
> *So the point of this openness is to manipulate software development tools so that we can access any kind of underlying networking technology and basically invoke all the **elementary functions** that we need to deliver a given service, and I think one of the means to achieve that kind of objective, that is to be as technology agnostic as possible, is precisely this notion of openness." – **Interviewee I30** (Bold emphasis added.)*

Standardised open APIs, and the ability to apply external software to network devices, act as safeguards against vendor lock-in imposed through points of architectural control in the network operators' infrastructures (Woodard, 2008). They help to circumvent challenges of substituting network hardware devices, when a vendor, for commercial reasons, couples a solution for a networking problem[35] to a particular network hardware product family. The uniformity of standardised open APIs, coupled with the ability to introduce capabilities to networking hardware using software that is not coupled to the hardware, instead create *openness in the sense of the solution*. Openness in the sense of the solution describes a state of affairs in which network operators can build their networking infrastructures based on an open generic solution architecture rather than on a patchwork of vendor-specific solutions. It is openness in terms of being able to substitute vendors' network hardware devices without losing the ability to address the problem that the network hardware devices were introduced into the network to solve. Openness accommodates substitution and complementing of vendors' solutions with less need to overcome challenges of vendor lock-in because standardised open APIs abstract heterogeneity to *logically* homogenise diverse vendors' network hardware devices.

Beyond logical homogenisation of hardware, i.e. *hardware abstraction*, network operators *seek* commoditisation of network hardware. It is a diminishing of pre-embedded feature-distinctiveness between vendor's hardware devices, and it has been *projected*, by network operators, to lower Capex. Interviewee **I29**, CTO at a large network equipment vendor with SDN products, Interviewee **I10**, Director at a SDO, elaborated:

> *"In a lot of cases today, where we used to have custom hardware in the network, we don't really need it, right? Just like in a lot of places we used to have custom silicon in the compute environment. We don't really need it, right, because of the maturity of what you might consider to be COTS (Commercial Off The Shelf) hardware, you know Intel or you know AMD or even Atom. So that's the trend we are seeing. That's playing out right now."* – ***Interviewee I29***, *(Meaning of COTS added.)*

[35] Commercially available routers typically have features not strictly limited to forwarding and the execution of routing protocols coupled with them.

> *"So, in the – sort of the end game is that the infrastructure itself becomes a pool of commoditised hardware resources consisting of networking for connectivity, compute and storage."* – ***Interviewee I10***

Not surprisingly, vendors that have benefitted from the distinctiveness of their hardware have received hardware abstraction via standardised open APIs and hardware commoditisation with some consternation. SDOs supported by very large network operators have worked to convince many vendors to implement standardised open APIs such as the OpenFlow protocol, but vendors remain concerned about how this affects their ability to differentiate products, and more than this, how much hardware differentiation continues to matter in the networking industry as more network operators transform their networks into SDN infrastructures. I return to how vendors have been responding in the Analysis chapter.

The decoupling and relocation of network intelligence to outside of a closed monolithic networking device under the control of vendors, has admitted innovation by, as of this writing, uncategorised networking technology organisations, as new entrants to the networking industry, many of whom have limited their innovations to software development, exclusive of any network hardware device innovation. Comparing and contrasting closed proprietary solutions with the openness of SDN infrastructures Interviewee **I32**, Researcher in SDN, stated:

> *"So obviously, this is creating a big change in the market, because, of course, then if most of the value is there - not completely right, but, you know, a big part of the value is there - then since this is becoming fully based on software and also on an open interface, no more internal to the device in a closed platform, then, of course, you open the market opportunities to other players that were not involved, so far."* – ***Interviewee I32***

Prominent examples of new entrants include Nicira with its pioneering Network Virtualization Platform later assimilated into VMware's NSX product with several hundred deployments in production SDN infrastructures (VMware, 2016), Big Switch Networks another start-up from the SDN research at Stanford University, with its Big Cloud Fabric and Big Monitoring Fabric SDN products for data centres (Big Switch

Networks, n.d.), and Nuage Networks, a start-up venture by Alcatel-Lucent[36], with its Virtualized Services Platform for data centres and wide area networks (Nuage Networks, n.d.). Standardised open APIs have also facilitated the creation of networking innovations that extend into previously closed or proprietary areas of network hardware devices. Using a quote from Interviewee **I37**, Deputy Head of R&D at a technology firm with networking products to explain:

> *"**The point is that, there are aspects of the decision of the functioning of the protocols - the decision to be taken on the capabilities of the packet processing that were decided by the vendor and <u>not exposed across their open APIs or open protocols</u>.** This - with this perspective, the openness in SDN allows you to mangle and de-mangle all the different bits and pieces of your box. So, in the end of the day, your open API allows you to go deeper into the hardware and decide for single flows how to manage it."* – **Interviewee I37**, *(Bold and underline emphasis added.)*

But there are limits to openness.

Although an open vendor-neutral networking industry with cost-effective replacement and interchangeability of networking devices are objectives of the morphogenesis of the networking industry, everything is not open. Technical openness in SDN infrastructures does not mean that complete transparency by vendors of what competitively differentiates their network hardware devices will certainly be achieved. Rather than morphogenesis, early signs point towards a state of morphostasis wherein hardware commoditisation has remained a *desired* outcome, by network operators. At the time of this writing, with the exception of commercially available bare metal routers and white box routers that very large network operators build themselves (see also (Open Compute Project, n.d.)), hardware abstraction via standardised open APIs and externally applied software has been the main outcome.

---

[36] Alcatel-Lucent was acquired by Nokia Corporation in January 2016 (Nokia Corporation, 2016).

Experience and infrastructure investment have been entry barriers for new entrants to the networking industry. In spite of the loosening of existing boundaries of control held by vendors, and the associated entrance of new networking technology organisations to the networking industry, many network operators have considered it an unacceptable risk to place relatively unproven products into production SDN infrastructures. Summarising the concerns of network operators, Interviewee **I35**, CTO at a large network equipment vendor and systems integrator, stated:

> *"In terms of deployment I think it's a different story.*
>
> *…*
>
> *So in the end, operators cannot take so much risk, I mean, actually they cannot take any.*
>
> *And so in the end, they were very reluctant to involve this broad ecosystem that they have experimented in experimentations, in prototypes and so on, and then they ask big players to take the risk and* ***so they ask big equipment providers or big expert integrators, IT vendors to take the risk to deploy such new solutions****." –* ***Interviewee I35****, (Bold emphasis added.)*

In other words, openness has practical constraints that limit the degree to which network operators' desire for lower Capex and Opex can be satisfied and that resist morphogenesis of the networking industry. To summarise, openness is an architectural feature of SDN infrastructures, but its presence particularly at the northbound and southbound interfaces of SDN controllers was deliberately devised by network operators, and SDOs to facilitate structural elaboration of the networking industry.

5.4.7 Virtualisation as a Key Enabler of SDN Infrastructures

Availability of proven facilitating technologies was identified as an enabling condition of generative mechanisms that promotes architectural evolution out of which have come SDN infrastructures. Virtualisation is one such facilitating set of technologies. Very importantly, virtualisation techniques are already widely used in networking (e.g., for virtual local area network (VLANS), virtual private networks (VPN), link virtualisation, etc.; see also (Wang, et al., 2013)), but in SDN these have been augmented with techniques borrowed from computing – specifically from cloud computing. Virtualisation is characterised by the ability to create logical representations of underlying physical hardware at some level of abstraction above the hardware. These logical abstractions are typically a combination of [self-contained] software entities and data that are managed by a *container*[37] software entity that is responsible for administering the lifecycle (from instantiation to termination) of the software entities (Smith & Nair, 2005; Rosenblum & Garfinkel, 2005; Zhang, et al., 2010; Pearce, et al., 2013; Medina & García, 2014).

A prevalent computing instantiation of this is *hypervisor* virtualisation. Hypervisors create and manage the lifecycle of *virtual machine* abstractions of underlying physical servers such that multiple isolated virtual machines can share logical slices of the same physical server resource, and virtual machines can be migrated to different physical resources (including while they are running[38]) (Smith & Nair, 2005; Rosenblum & Garfinkel, 2005; Zhang, et al., 2010; Pearce, et al., 2013; Medina & García, 2014). Two benefits of applying hypervisor virtualisation are efficiency of physical resource utilisation, and resilience of services running within the virtual machines because migration of virtual machines allows them to survive underlying hardware failures.

There are multiple ways to utilise virtualisation in SDN infrastructures and varying virtualisation techniques have been used by network operators at different levels of SDN infrastructures. There is no minimum virtualisation or precise prescription on exactly what should be virtualised. A comprehensive technical survey of virtualisation in SDN infrastructures is beyond the scope of what is necessary to answer the research

---

[37] Depending on the specific virtualisation technique this might be referred to as a monitor (Smith & Nair, 2005; Rosenblum & Garfinkel, 2005; Zhang, et al., 2010; Pearce, et al., 2013; Medina & García, 2014). Note that as used here, the word container is not referring specifically to container-based virtualisation approaches (such as Docker).

[38] See for example, VMware's vSphere vMotion product (VMware vSphere, n.d.-b).

question. Still, a high-level description of two levels of virtualisation used in SDN infrastructures is necessary for understanding the later analysis.

First, the programmatically created logical abstractions of the physical network, whose lifecycle is managed by a SDN controller are *virtual networks* (see Figure 5-8). Depending on the SDN infrastructure, a virtual network may be constituted of:

1. Information stored in a SDN controller[39] and the forwarding tables of physical routers to logically demarcate the network or,
2. It may exist in a SDN controller, software routers (such as Open vSwitch (Open vSwitch, n.d.)), and in forwarding tables on physical routers, or
3. Purely in a SDN controller and software routers.

Virtual networks can be dynamically created, modified, destroyed or migrated across physical routers while operational. Second, virtualisation techniques may be applied to the SDN controller itself. A SDN controller may be deployed to run within a virtual machine, whose lifecycle is managed by a hypervisor that abstracts the physical servers on which the virtual machine runs. This is typically the case in network operators' production SDN infrastructures. The consolidation of routers' control planes into a logically centralised SDN controller running within a virtual machine may be understood to be a virtualisation of the control plane. As Interviewee **I14**, Distinguished Engineer at a large network equipment vendor and systems integrator, remarked:

> *"But you can centralise and you can virtualise - and virtualisation is the important part of this and not but until the advent of SDN we weren't even virtualising the control of these routers. So to some degree SDN*

[39] The architecture of SDN controllers is more sophisticated than the high-level description given, but such technical particularities do not add anything substantive for the analysis required to answer the research question.

also provides a motivation for virtualising the control, even if all we did was virtualise." – ***Interviewee I14***

Depending on the virtualisation approaches used in SDN infrastructures there may be a very tight coupling with virtualisation, but they remain distinct concepts. Clarifying the distinction between virtualisation techniques borrowed from computing and SDN infrastructures, Interviewee **I30**, Director in Strategy in Networking at a Tier 1 ISP, for instance, stated:

> *"…**the fact that you are using virtualisation techniques does not necessarily mean that you take full advantage of the SDN approach**. That is, I would say that **data centre networking has been out for quite some time and you have the ability to play with virtual machines for quite some time to do some specific tasks, and at that time there was no specific mention of SDN or even service orchestration by definition**, but **if you needed some resources and you wanted to get as independent as possible from the hardware, from that standpoint, virtualisation is really attractive** in the sense that you can easily use that kind of technique for simulation purposes or facilitating the management of some specific resources mainly CPU resources, for example, **and yet you still don't have SDN per say**." –* ***Interviewee I30****, (Bold emphasis added)*

To recapitulate, mature virtualisation techniques are a key enabler of SDN infrastructures.

5.4.8 Software-Definition of Networking Infrastructure Hardware

Together, decoupling of the forwarding and control planes, aggregation and centralisation of control planes, programmability and network abstraction and openness, produces networking infrastructure hardware *generification* by which the ability to apply software that is not coupled to network hardware devices to the network is achieved. Likewise to the transformation of the computer from specialised hardware to general-purpose hardware whose behaviour is stipulated by the software it executes, that is, the hardware is *software-defined*, the transformation of traditional networking is towards software-definition of open, generified networking infrastructure hardware. Using a summarising quote from, Interviewee **I27**, Lead Member of Technical Staff at a Tier 1 ISP:

> *"So the network is very, very kind of, let me find the correct word here, very tied with software with hardware.*
>
> *…*
>
> *So, I mean, we don't really need all this intelligence in those boxes. All we need is certain hardware that actually listen to whatever problem I put on top of it*
>
> *…*
>
> *…and by the way, those programs should not be tied to the devices themselves."* – ***Interviewee I27***

The generification of routers transforms them into simplified data forwarding devices with network intelligence consolidated within a SDN controller, making external software-definition of new network capabilities possible. Architectural evolution of routers' de facto standard implementation architecture that splits router functionality across hardware and external software, is favourable to network operators, as it releases them from conditions of constraint on network service innovation imposed by limitations of pre-packaged vendor-defined product features, and from dependence on vendor-determined network hardware devices' innovation cycles.

Regarding the ontology of SDN infrastructures, the preceding sections have identified an invariant *core SDN architecture*, but to reiterate, there are variations in SDN infrastructures and vendor SDN product implementations, including variations at a conceptual level. Consequently, there are different degrees to which these architectural features may be manifested within any network operator's SDN infrastructure.

For instance, in its beginnings, VMware NSX (originally derived from Nicira's Network Virtualisation Platform) mainly promoted the building of entirely software-based SDN infrastructures with software routers making forwarding decisions, over abstracted underlying hardware routers (i.e., a *network overlay*) that indifferently carried out the actual forwarding of data over physical distances[40] (Nicira Networks Inc., 2012a; Nicira Networks Inc., 2012b; Nicira Networks Inc., 2012c; Nicira Networks Inc., 2012d; VMware Inc., 2013a; VMware Inc., 2013b; VMware Inc., 2016). That approach to software-defined networking decoupled the control and forwarding planes, and more than this, relocated some of the *forwarding plane* responsibility of physical routers to software routers. This is the third type of virtual networks listed in the preceding section on virtualisation. In a different approach, the OpenDaylight Foundation has promoted equally the building of pure network overlay-based SDN infrastructures, SDN infrastructures with entirely physical routers (and virtual network abstractions in the OpenDaylight SDN controller), and SDN infrastructures that are hybrids of these (OpenDaylight Foundation, n.d.). That is, all three types of virtual networks listed in the preceding section.

The decoupling of the control plane may be a logical decoupling rather than a physical decoupling. For instance, many vendors have retrofitted existing routers with an implementation of a version of the OpenFlow protocol or other southbound protocol, but the physical router's control function has not been removed. The means by which this is done is through *firmware updates*. Other vendors offer bare metal routers with no preconfigured functionality beyond forwarding (such as products by Edge-Core Networks or Quanta Cloud Technology (Edge-Core Networks, n.d.; Quanta Cloud Technolgy, n.d.)).

Continuing, centralisation of consolidated router control planes into a SDN controller is conceptual rather than literal. A literal centralisation would create a single point of failure and performance congestion in a network operator's SDN infrastructure. So it is more accurate to describe the SDN controller as being *logically centralised*. *Clustering* of SDN controllers (i.e., multiple co-existing coordinated executing instances

---

[40] VMware NSX has increased support for physical SDN infrastructures via link layer software gateways to physical networks (VMware Inc., 2016), but it remains primarily a network overlay approach.

of SDN controller software that behave as one logical entity) is a common manifestation of logical centralisation. In elaborate SDN infrastructures, federation of coordinated SDN controllers is another[41]. Network operators also consider the practical balancing of local control in routers with the consolidated network control plane of the SDN controller in production SDN infrastructures to contend with SDN controller failure scenarios. For example, Interviewee **I13**, Head of Architecture in a SDO and Interviewee **I22**, Senior Director at a network equipment vendor and systems integrator expressed practical SDN infrastructure design principles for network operators:

> *"…one needs to understand that a centralised controller doesn't necessarily mean that you only have one instance of the central control entity. So centralised control as I personally, my company, and* ***[Anonymised Detail]*** *understand it, certainly allows multiple instances that can be also distributed physically and that basically together act as a central control instance or control entity, let me put it that way." –* ***Interviewee I13***

> *"Complement logical centralised control within embedded control where it makes sense and that's what the SDN architecture allows." –* ***Interviewee I22***

Notwithstanding this variety in SDN products and production SDN infrastructures, decoupling of the forwarding and control planes, aggregation and centralisation of control planes, programmability and network abstraction, openness and the use of virtualisation, is invariant across SDN infrastructures. These distinguish SDN infrastructures from traditional networking infrastructures.

I must mention briefly, that in SDN infrastructures there is always a *management plane* (typically consisting of network management or infrastructure orchestration software) that integrates via northbound interfaces with the SDN controller. Because

---

[41] See for example (Verizon, 2016).

architectural detailing of this plane was not found pertinent to understanding architectural evolution in ossified digital infrastructures, its discussion is not included.

5.4.9 Distinguishing Characteristics of SDN Infrastructures

Broad distinguishing infrastructural characteristics arise when the core SDN architecture is instantiated in SDN infrastructures. Regardless of what differentiates SDN infrastructures from traditional networks, their core responsibility as networking infrastructures remains interconnectivity and transportation of digital data from a source to its destination. Nonetheless, differentiators make it clear how SDN infrastructures are indeed the structural elaboration of traditional networks in a significant, non-trivial manner. Two general distinctive characteristics of SDN infrastructures are introduced next. These characteristics are revisited in the analysis.

5.4.10 From General-Purpose to Dynamically Tailored Infrastructure

Northbound interfaces of SDN controllers allow client applications in network operators' SDN infrastructures to dynamically create virtual networks tailored to their specific requirements at a moment in time. As a client application's requirements change over time, it can dynamically modify or destroy the virtual networks created for it. This is distinctly different from traditional notions of networking infrastructures as derived from IS research, in which a digital infrastructure is characterised as a shared general-purpose homogenous resource – homogenous in the sense that it offers the same underlying *capabilities* to all tenants of the infrastructure. Whereas a traditional networking infrastructure is a general-purpose resource shared by several client applications, a SDN infrastructure allows dynamic software-definition of virtual networks with application-specific delimited scope and capabilities, and it also allows application-specific tailoring of *shared* networking infrastructure. Higher levels of infrastructure flexibility can be achieved through granular application-specific tailoring of networking infrastructures in the control plane, and this is an inherent capability of SDN infrastructures facilitated by its core architecture.

Certainly traditional networks can be segmented (e.g., via VLANS) to host single applications, or network functions can be applied to data flows in shared infrastructures, and indeed a virtual network in a SDN infrastructure can host multiple applications, but traditional networks are premised on historical networking rationale and concepts developed that originated in the 1960s (Kurose & Ross, 2013, pp. 86 - 92) well prior to the advent of the previously mentioned technical enablers of SDN infrastructures. In contradistinction, SDN infrastructures are fundamentally amenable to fine-grained isolated tailoring and software-definition of virtual networks that are specific to the needs of tenant client applications. It is not a retrofitted or extended capability. It is inherently characteristic of SDN infrastructures.

5.4.11 From Rigid and Resistant to Change to Flexible and Accommodating of Change

Some distinctions between characteristics of traditional networking infrastructures and SDN infrastructures relate to their basis in the physical and the logical respectively. Networks have always been closely related to physical entities and concepts. The function of data transportation over geographical distances necessitates physical distribution of networks. Performance limitations of available technologies have led to a favouring of physical implementation of network functionality in hardware. With network intelligence confined to the enclosure of physical networking devices, traditional networking infrastructures have been subject to long innovation cycles of hardware and correspondingly slow network protocol standardisation processes. Commenting on physicality, Interviewee **I22**, Senior Director at a network equipment vendor and systems integrator, explained:

> *"**Today everything is built around the physical world**. It's not built around services - and by the way, when I say services, it's not just for the carrier community, the telecommunications community, even at the enterprise community, you know, what they do is assign applications to VLANs. You know it's very physical-oriented, security is - **the perimeter is really built around physical network devices - you know, creating guest LANs, using separate access points and wireless infrastructure - that kind of thing. If you think about it, it's not logical, it's not service-oriented**..." – **Interviewee I22**, (Bold emphasis added.)*

A tight coupling with the physical, renders traditional networks inherently less accommodative of change in contrast to SDN infrastructures in which physical entities and concepts matter less or sometimes not at all. Clarifying the adverb “sometimes,” being digital infrastructures, SDN infrastructures inherit characteristics from traditional networks (Star & Ruhleder, 1996; Hanseth & Lyytinen, 2010; Aanestad & Jensen, 2011); they are never entirely disassociated from some ultimate underlying physicality. In spite of this, in highly virtualised environments, such as in data centre networking, some constraints of physicality are circumvented by abstraction and the use of virtualisation techniques.

Recall that the decoupling of the forwarding and control planes combined with consolidation of routers’ control planes into a SDN controller splits router functionality into a hardware-based forwarding plane, and a software-based network control plane that encapsulates the intelligence of the network. An implication of this is that network operators can leverage well-established development and testing techniques from the field of software engineering to shorten network service innovation timescales.

Both sociotechnical ossifications of and around the Internet’s core architecture, and tight coupling of network function manifestation with physicality, contribute to difficulty in deploying new network layer protocols in traditional networking infrastructures. Production SDN infrastructures, despite being significantly technically advanced beyond early research that experimented with new networking protocols and network architectures, have retained the capacity to accommodate the deployment of new protocols. As a further matter, though restrictions on interactions between layers is relaxed in the Internet’s core architecture (van Schewick, 2010, pp. 46-47), and somewhat in violation of the end-to-end principle (Saltzer, et al., 1984) as evident with the Address Resolution Protocol (ARP), a link layer protocol implemented in switches[42] that has awareness of the network layer (Tanenbaum, 2003, pp. 450-452; Kurose & Ross, 2013, p. 494), SDN infrastructures accommodate even less strict

[42] Both switches and routers can implement link layer protocols. See also the explanation in (Kurose & Ross, 2013, pp. 506 - 508).

layering. Abstraction and virtualisation techniques in SDN infrastructures, *intrinsically* accommodate much greater degrees of layering ambivalence. To illustrate, while discussing interoperability between traditional networks and SDN infrastructures, Interviewee **I28**, Researcher in SDN, explained:

> *"…looking at the interoperating SDN and existing networks, you start,* ***for example you might want a layer two Ethernet link, but you are not, you don't have layer two connectivity to some other network. So you run it on top of IP but you use layer three to run layer two or even layer four to run layer two****. That's the lot. So what's happening* ***there's much less of a strictness in how the layers are used - becomes more tactical****."*
> *–* ***Interviewee I28****, (Bold and underline emphasis added.)*

This is not to say that such flexibility cannot be accomplished at all in traditional networks, but the emphasis is that this ambivalence, is a foundational capacity of SDN infrastructures, retained from predecessor academic research that produced SDN as an innovation, rather than a retrofitted or extended capability.

5.4.12 Summary of Characteristics of SDN Infrastructures

Variety in SDN infrastructure implementations means that characteristics of dynamically tailored infrastructure, and flexibility and accommodation of change, differ across network operators' infrastructures. Likewise to architectural features, these characteristics are always present in SDN infrastructures, but may be actualised more strongly in some instantiations than in others.

5.4.13 What is Infrastructure?

The final part of elucidating, from an architectural perspective, SDN infrastructures as structurally elaborated traditional networks and the interrelationship with the morphogenesis of the networking industry, involves clarifying what *makes up* infrastructure when, following transformation of a traditional network, function previously manifested in hardware is carried out by decoupled externally applied software – and more broadly there is a generification and subjugation of networking infrastructure hardware to software-definition. This is a question of what the term "infrastructure" refers to in the context of SDN infrastructures. Understanding this makes clear what participates in architectural evolution as well as what is not instantiating SDN architecture and therefore lies outside of SDN infrastructures, and it is relevant for elucidating the generative mechanisms sought by the research question.

The next section starts by describing the role of hardware and externally applied software in SDN infrastructures, and it is followed by a definition of what the term "infrastructure" refers to in the context of SDN infrastructures.

5.4.14 Characterising the Role of Hardware and Software in SDN Infrastructures

Several times within the preceding sections, it is stated that it is *network intelligence* that has been decoupled from physical hardware routers in SDN infrastructures[43]. Characterising then, its role in SDN infrastructures, externally applied software is the holder of network intelligence.

The geographical distribution aspect of physical networking infrastructures cannot be decoupled alongside the network intelligence that is centralised in an external software encapsulation, i.e., the SDN controller. In fact, even this externally applied software has to be geographically distributed in order to effectively manage the network. SDN controller instances must be federated and placed within relative proximity of the hardware under its control in order to manage it in a timely manner (recall the brief point on federation in section 5.4.8). Beside these requirements imposed by geographically associated physicality, there remains continued significance and necessity of hardware because the externally applied software is ultimately executed on hardware – specifically on general purpose commodity servers.

Additionally, although the compositional physicality of networking infrastructures may change due to the decoupling of network intelligence from the physical hardware and its centralisation in software, (e.g., such that certain processors, circuits, etc. are no

---

[43] "Network Intelligence" is precisely the terminology used by interviewees and in the wider networking industry.

longer needed within routers – see also (Open Compute Project, n.d.)), physicality is inalienable. So hardware, necessitated by physicality, always has a role in SDN infrastructures, but the *pre-embedded* functional scope of the hardware is being adjusted. Quoting Interviewee **I29**, CTO at a large network equipment vendor with SDN products, who discussed the role of hardware and software in SDN infrastructure:

> *"So I don't think hardware is going to go away or anything like that, because even if you have V-Switches (virtual switches) that are running on a hypervisor somewhere, they still have to run on something, right. So hardware itself is not going to go away. The question is, what's the form of that hardware?"* – ***Interviewee I29****, (Meaning of V-Switches, which are software routers, added.)*

The morphogenesis of traditional networking infrastructures may have produced architectural evolution of an ossified router implementation architecture, but performance requirements of networking infrastructures remain a practical technical constraint on what traditional networking responsibilities can be decoupled from hardware routers and centralised in externally applied software. Performance limitations of available software solutions and of commoditised hardware solutions are some reasons, from the perspective of network operators, for retaining some network intelligence and decision-making autonomy in hardware, and for the continued required role of specialised feature-rich networking hardware rather than bare-metal software-defined hardware in SDN infrastructures. From the perspective of network equipment vendors, there are other reasons which I introduce in the Analysis chapter. Discussing the role of hardware, Interviewee **I32**, Researcher in SDN, for example explained:

> *"This is the extreme possibility that you have. So soft-switching, purely soft-switching, which means general purpose hardware and then software switches on top of it and then you can peer this with hardware optimised solutions for switching.*
> *In terms of performance - gigabits per second the capacity that you have, you have is at least 2 orders of magnitude in difference at least, OK, and this difference will probably continue to be there for several years.*

So the hardware will not disappear, for sure - specialised hardware for networking…" – ***Interviewee I32***

Enabled by flexibility of the core SDN architecture, the balance of the allocation of network intelligence (and even forwarding responsibilities as with Open vSwitch and VMware NSX (Open vSwitch, n.d.; VMware Inc., n.d.-a)) to externally applied software or to hardware varies across SDN infrastructure instantiations. That is to say, mediated by network operators' tolerance of performance impact, software-definition allows the roles of hardware and externally applied software to be more fluid than in traditional infrastructures, making it less straightforward to characterise the role of hardware than of the externally applied software. Nonetheless, what can be said is that from a technical perspective, the role of hardware remains high-throughput data processing and physical network interconnectivity in SDN infrastructures.

Very importantly, I have specifically referred to the role of externally applied software, and not to the role of software in general. Software is involved throughout networks, but the focus in this chapter has largely surrounded the change in the relationship between the forwarding and control planes. Within this limited scope, I have explained the role of the software that constitutes the consolidated network control plane in SDN infrastructures: the SDN controller. Notwithstanding, recall that the forwarding and control planes are a logical stratification of routers' forwarding and control functions and that within routers, the control function is implemented as software. What this means is that, even in the absence of SDN infrastructures, within the confines of routers, hardware and software already have roles similar to those described in this section. The hardware is responsible for high-throughput forwarding, and the software is responsible for route computation – network intelligence. Quoting in support Interviewee **I30**, Director in Strategy in Networking at a Tier 1 ISP:

"Actually, route computation itself is made in software while packet processing is made by hardware in most if not all the routers implementations." – ***Interviewee I30***

The difference with SDN is that the software is decoupled from the hardware and is externally applied, while traditionally in routers, the software is physically embedded within the device. But this raises a need to further characterise hardware and software. For instance, is the control plane that was part of network hardware devices but has been decoupled and consolidated into an externally applied software entity, infrastructure, or is it a software *service* that makes use of physical infrastructure resources?

On the basis of IS theorising on digital infrastructures, the SDN controller is classified as a service within a service infrastructure that is distinct from but dependent upon a transport infrastructure of which routers are constituents (Hanseth & Lyytinen, 2010). The conceptualisation of a service and transport stratification of an aggregate support infrastructure is an analytic one derived from a logic of decomposition borrowed from modularity theory (Hanseth & Lyytinen, 2010). It is useful, and networking infrastructures are certainly support infrastructures, but as the infrastructure stratification (Hanseth & Lyytinen, 2010) is defined, its classification of the SDN controller is challenged by the controller's participation in transport-related activities designated to transport infrastructures, and by the co-located management of virtual network abstractions within the SDN controller (see Figure 5-8). This indicates that there might be complementary or alternative ways of understanding infrastructure within the scope under study.

Because it is necessary to be unambiguous about what participated in architectural evolution through which SDN infrastructures arose, I next discuss the logic by which the interviewed networking experts demarcated what hardware and software is included in the definition of infrastructure in relation to SDN infrastructures.

5.4.15 Definition of Infrastructure in Relation to Software-Defined Networks

For clarity, the way that the networking experts defined infrastructure is explained before what makes up "infrastructure" is delineated.

In accordance with IS theorising on digital infrastructure, the experts interviewed concurred that infrastructure is defined relative to the perspective from which it is considered (Star & Ruhleder, 1996). Differently still, being networking experts tasked with designing SDN products, and building complex networking infrastructures, their view of networking infrastructures is more complete than that of end users in an organisational context – a typical orientation of IS studies of digital infrastructures. Consequently, their operationalisation of the term "infrastructure," though relative, acknowledges more explicitly the ultimately underlying collective of all constituent hardware and software. Infrastructure is deemed a technical entity of which users may make use or to which they may contribute, but users are not considered included in the definition of infrastructure. This is not to say that the experts do not understand how infrastructures are implicated broadly in social reality, or how more than technical entities constitute organisational infrastructures. They understand these things, but infrastructure is distinctly identifiable and meaningful in the absence of considerations of users. Demonstrating a sociotechnical understanding that simultaneously acknowledges the distinctiveness of the technical, Interviewee **I28**, Researcher in SDN, for example, commented when defining what makes up infrastructure:

> ***"…****depends on how you want to define it.*
> *You could say a certain way - that a certain minimum set of hardware and a certain minimum set of software to operate that, is what now constitutes the infrastructure and everything above that is driven by users.* ***So maybe the infrastructure gets reduced.***
>
> *If you look at the whole thing, the whole eco system, OK, then the infrastructure gets expanded, because it became available to users that before did not have the sophisticated knowledge to be able to dabble in the network, and now it's become possible for more users now.* ***So maybe the infrastructure has expanded.*** *It depends where you draw the line." –* ***Interviewee I28****, (Bold emphasis added.)*

An articulation of infrastructure that directly engages and keeps discrete the technical is helpful, because the objective of this research is to uncover mechanisms related to architectural evolution, and architectural evolution is primarily a technical achievement. At the same time, the balanced acknowledgment of the sociotechnical nature of infrastructures attends to the practicality of changing the architecture of technical infrastructures that are intertwined with the social.

Having provided this context, the definition of what the term "infrastructure" refers to in the context of SDN infrastructures can now follow.

Infrastructure, as framed by the preceding, can be defined from different logical perspectives. Three logical perspectives are the physical perspective of infrastructure, the functional perspective of infrastructure, and the value perspective of infrastructure. None of these perspectives are primarily premised on a logic of decomposition for demarcating infrastructure.

The physical perspective discriminates and imposes a boundary on what makes up "infrastructure" on the basis of physical manifestation. For traditional networks and SDN infrastructures, the physical perspective admits only hardware, i.e., routers, fibre optic cables, base transceiver stations, etc., as constituents of infrastructure. The functional perspective is more abstract, and is the perspective preferred by interviewees for defining infrastructure. The functional perspective frames infrastructure in terms of function – what infrastructure does or is supposed to do – in abstraction from the means by which these functions are instantiated. It describes infrastructure's aggregate functionality as being physically and logically distributed across functions contained within or manifested as hardware or software. It also designates the container or manifesting entity of infrastructure functionality as being infrastructure. Here, whatever contributes infrastructure function makes up infrastructure. In other words, *the locus of infrastructure intelligence is infrastructure*. Explaining this perspective, Interviewee **I22**, Senior Director at a network equipment vendor and systems integrator, commented:

> *"So if you want to talk about a hardware switch, whether it has software, hardware or whatever, or a server, hardware, software, well, it may be a little different even though there is some embedded software there as well, you know, those are infrastructure*
>
> *…*
>
> *If we go back to the definition that I have been relying on and I've just shared with you a minute ago about infrastructure has to do with the - in essence, data planes versus control, then a virtual switch is clearly part of the infrastructure."* – ***Interviewee I22***

The suggestion is hardware, as expected, makes up infrastructure if it contributes to infrastructural function. Routers, whether specialised hardware or simplified bare-metal data forwarding devices, lie within the boundary of and make up SDN infrastructures, because they facilitate the aggregate data transmission and interconnectivity function of networking infrastructures. In the physical perspective of infrastructure, the consolidated control plane manifested as a software SDN controller is excluded from the ontology of SDN infrastructures, because of its decoupling from hardware, yielding an inherent conceptual contradiction of what is core SDN architecture. Likewise to the logic of decomposition described in (Hanseth & Lyytinen, 2010), the SDN controller is classified as an external service that leverages capabilities of physical infrastructures.

The functional perspective circumvents this inconsistency, because function is what demarcates infrastructure, and not manifestation, whether physical or logical. Thus, that the control plane is decoupled and consolidated into externally applied software, does not change the designation of its *locus* as making up infrastructure. The externally applied software is infrastructure, and in general, any software that facilitates infrastructure functionality makes up the infrastructure. Hardware and software that does not facilitate infrastructure functionality is otherwise classified. Quoting Interviewee **I17**, Senior Network Architect for SDN and NFV at an ISP in support of this point:

> *"So now we have - in the past we had only one component was the network device having the control plane and data plane, we now split that. So now we have data plane that is running on the network devices and we have control plane that is running on a centralised software controller, but this is still infrastructure.*
>
> *I mean, the network as a whole, the network as a whole has been divided into two pieces, but both are infrastructure."* – ***Interviewee I17***

The functional perspective yields a less granular conceptualisation of infrastructure than does a logic of decomposition, but it is more analytically consistent with the scope of the changed relationship between the forwarding and control planes.

To summarise, according to the experts, the functional perspective of infrastructure is preferred when demarcating what is infrastructure. The functional perspective's demarcation of what makes up infrastructure is ambivalent to the flexibility of role allocation between hardware and software. In this perspective, *the locus of infrastructure intelligence is infrastructure*. Since the locus of the aggregate function of infrastructure is a distribution of hardware and software that provide infrastructure function, the SDN controller software which encapsulates the network control plane, is equally infrastructure as are hardware routers.

Alongside the functional perspective of infrastructure, the experts discussed understanding infrastructure in terms of value. In the value perspective, infrastructure is conceptualised in terms of which of its constituents are assigned technical or business value. In the context of the advent of SDN infrastructures, the value perspective establishes the interrelationship between the two morphogenetic cycles that have been discussed to this point. Network intelligence is what is considered valuable in infrastructure, and whoever holds control over it, derives corresponding economic benefits from it. Academic research that produced the innovation of SDN changed the relationship between the forwarding and control planes of routers to achieve technical goals, but network operators' interest in the innovation of SDN in preference to other predecessor network solutions, has been underpinned by the recognition of a role played by the *relationship* between the forwarding and control planes.

As stated during the discussion of the structural conditioning of traditional networks, the de facto standard implementation architecture of routers co-locates forwarding and control planes inside proprietary network hardware devices, whose features and innovation cycles are decided by vendors. Consequently, vendors have had significant architectural control over some economic activities associated with network operators' introduction of new networking capabilities into their infrastructures. Thus, the relationship between the forwarding and control planes within the sociotechnically ossified de facto standard implementation architecture of routers has served as an architectural point of control for vendors (Woodard, 2008). Interviewee **I31**, Researcher in SDN, for example, portrayed this as control of the network layer of the Internet's core architecture:

> *"So the idea there was that people like Cisco and Juniper who have a control of the layer three of the Internet with the devices they provide – so there is this dominant feature, and it's hard to escape from them – and the notion there was that you make basically a controller, software controller control all of the critical aspects of the router and make the router just a basic hardware device."* – ***Interviewee I31***

Since SDN as an innovation changes the relationship between the forwarding and control planes, when implemented in networking infrastructures, it as a corollary reassigns the value to an external software entity. It is this reassignment of value which subverts the architectural control held by vendors that was recognised by network operators and SDOs, who accordingly appropriated the innovation of SDN from academic research and reframed its implementation in production infrastructures as a strategy for changing the networking industry. Very importantly, the explanation of architectural evolution in sociotechnically ossified digital infrastructures introduced in this thesis consequently necessarily includes key distinguishing features relative to extant IS theorising, that would not be present had subversion of vendors' architectural control and change to the necessary internal relations of the networking industry as a social structure not been an objective of network operators and SDOs.

Because the network operators that have been implementing SDN infrastructures are owners of large networking infrastructures their actions have also been meant to signal a renegotiation of their relationships with vendors. Stipulations of openness resist regression to proprietary architectural control by vendors over this external software entity, and the as yet unrealised objective of achieving hardware commoditisation reflects the reassignment of value from hardware devices to externally applied software.

5.4.16 Summary of Structural Elaboration

This section presented, from an architectural perspective, the outcome of the structural elaboration of traditional networks and how the morphogenesis of the networking industry, is interrelated with the morphogenesis of traditional networking infrastructures. To recapitulate, the advent of SDN infrastructures as structurally elaborated traditional networks, is the outcome of a strategy by network operators and SDOs, for effecting broad network industry change instrumented through specific technical means. SDN as an innovation was appropriated from academic research because its architectural features, when implemented in SDN infrastructure, subvert architectural points of control which vendors have held over network operators' networking infrastructures that have preserved structural conditions of constraint unfavourable to network operators.

From an architectural perspective, the distinguishing features of SDN infrastructures as elaborated traditional networking infrastructures are, decoupled forwarding and control planes, aggregation and centralisation of control planes, inherent infrastructure programmability and network abstraction, technical openness that promotes openness in the sense of the solution for the purpose of allowing interchangeability of vendors' products and the entrance of new innovators to the networking industry, and virtualisation as a key enabler. Each aspect of the architecture is ultimately related to a change in the relationship between the forwarding and control planes of routers. Characteristics of dynamically tailored infrastructure, and flexibility and accommodation of change are instantiated with, and distinguish SDN infrastructures. In SDN infrastructures, it is a logic of demarcation premised on a functional perspective of infrastructure that identifies the boundaries of what makes up infrastructure. A value perspective of infrastructure reveals reasons beyond the aforementioned enabling, stimulating and releasing conditions for why SDN as an innovation was appropriated by network operators from academic research and implemented as production networking infrastructures.

5.5 Summary of Findings

In summary, this chapter explained the advent of SDN infrastructures in terms of two interrelated morphogenetic cycles of structures: the morphogenesis of traditional networks as sociotechnical structures, and the morphogenesis of the networking industry as a social structure. Antecedent structural conditioning was presented and explain to be part of the analytic history of both morphogenetic cycles. This was followed by identification of enabling, stimulating, and releasing conditions for the generative mechanisms that promote architectural evolution in sociotechnically ossified traditional networking infrastructures. Finally, the structural elaboration of traditional networks and its interrelation with the early stages of the structural elaboration of the networking industry followed.

To recap, SDN as an innovation began in the context of academic research on the development of networking protocol and architecture alternatives to that of the sociotechnically ossified Internet core architecture for networking. Network operators under challenges of increasing network complexity, inflexible infrastructures, and unsustainable Capex and Opex, have recognised that a methodology of patching infrastructure cannot be maintained, and that the solution involves a radical change of the architecture in their networks. Conceptual familiarity between SDN as an innovation and preceding technologies, and technical achievability were identified as enabling conditions for network operators' transformation of their traditional networks. A recognition by network operators that a methodology of patching infrastructure is not a general solution for accommodating new customer network services, or for networking infrastructure evolution was identified as being a key stimulating condition for the advent of SDN infrastructures, helped by a releasing condition of the availability of SDN-compliant hardware.

A corollary of the initial SDN academic research is a change in the relationship between the forwarding and control planes of routers. Network operators and SDOs, recognising that, aside from facilitating software-definition of generified network hardware devices, the changed relationship occasions opportunity for a broader subversion of architectural control held by incumbent vendors that has reinforced structural conditions of constraint unfavourable to network operators, have reframed SDN implementation in production infrastructures as a strategy for changing the networking industry via architectural evolution of the ossified router implementation architecture.

Finally, the resulting SDN infrastructures that instantiate the core SDN architecture retain from the initial products of academic research the ability to accommodate new protocols, and are ambivalent to traditional notions of layering.

6 Analysis

6.1 Introduction

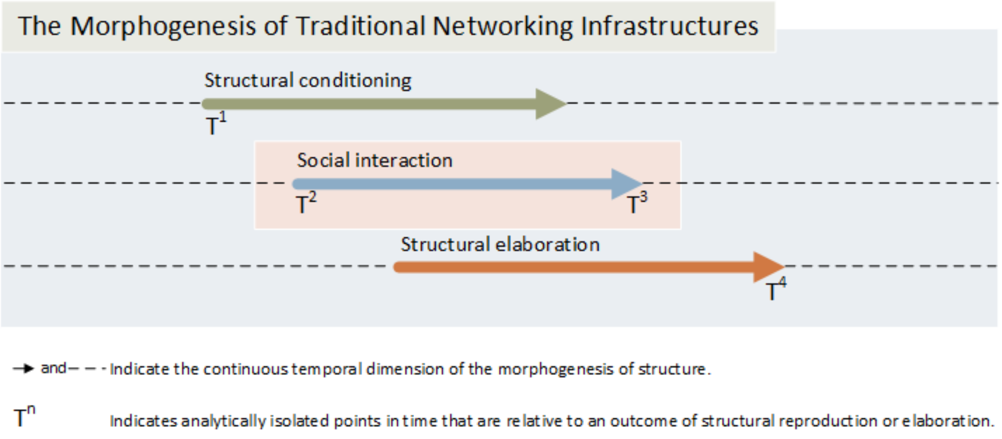

Figure 6-1 Social Interaction in the Morphogenesis of Traditional Networking Infrastructures

Having explained in detail the antecedent structural conditioning, enabling, stimulating and releasing conditions of generative mechanisms, and structural elaboration in the morphogenesis of traditional networking infrastructures, and having introduced the interrelationship with the morphogenesis of the networking industry in the preceding chapter, this chapter presents arguments for the reality of the generative mechanisms sought by the research question that were identified through a process of retroduction.

The operation of morphogenetic generative mechanisms is what causally intervenes between antecedent structural conditioning and structural elaboration in the morphogenesis of structure. Their operation constitutes the social interaction that transforms structure in the morphogenesis of structure (see Figure 6-1). As explained previously, Archer asserts further that morphogenesis and morphostasis are generative mechanisms (Archer, 1995, p. 217).

The objective of this chapter is to go beyond these high-level theoretical constructs to explicitly identify, and to explicate a set of morphogenetic generative mechanisms which promote architectural evolution in ossified digital infrastructures.

Three types of generative mechanisms were found: sociotechnical generative mechanisms, technical generative mechanisms, and social generative mechanisms (one of which promotes architectural evolution). The structures with which the technical and social generative mechanisms are instantiated are identified in this chapter, but as I alluded to at the end of Chapter 3, some incongruences between critical realism as a philosophical basis for IS research were discovered. These relate to the structure with which the sociotechnical generative mechanisms are instantiated. I elaborate the issue in the discussions in Chapter 7.

The chapter proceeds with a basic identification of retroduced generative mechanisms. It then presents detailed arguments for their reality and gradually introduces other generative mechanisms that complete the causal explanation. The collective of generative mechanisms describes a complementary insight on architectural evolution in sociotechnically ossified digital infrastructures to that of extant IS theorising on digital infrastructure architectural evolution.

Architectural evolution through softwarisation is introduced as a generative mechanism by which architecture in sociotechnically ossified digital infrastructures is evolved. This type of architectural evolution is causal to the structural elaboration of traditional networking infrastructures as sociotechnical structures, and interrelates with the morphogenesis of the networking industry in important ways. In the case of the advent of SDN infrastructures, architectural evolution through softwarisation centres around architectural evolution of the de facto standard router implementation architecture that is pervasive in traditional networking infrastructures. But there is another aspect to architectural evolution through softwarisation. Architectural evolution through softwarisation exploits the existing digital materiality of digital infrastructures to facilitate subsequent continued architectural evolution differently again from extant IS theorising on digital infrastructure architectural evolution.

As discussed in Chapter 2, IS research theorises architecture in digital infrastructures from the perspective of there being a singular architecture whose generative capacity lends itself to exploitative innovation as manifested in evolving deployment architectures of instantiating digital infrastructures. Architectural evolution through softwarisation, on the other hand yields an outcome where architecture's generative capacity is in its ability to simultaneously support multiple heterogeneous, possibly incompatible, architectures. With some limitations, digital infrastructures that instantiate this kind of ambivalent architecture do not require replacement when the need to implement heterogeneous possibly incompatible architectures arises. Though seemingly paradoxical given the state of extant information systems theorising, the particulars of this are comprehensively explained in this chapter.

The final section of this chapter, discusses the most recent stage of the morphogenesis of the networking industry, at the time of this writing. It presents the structural conditioning of elaborated traditional networking infrastructures, i.e. SDN infrastructures, on vendors.

Before proceeding a terminological clarification is necessary. Individually, generative mechanisms may be described monolithically, but they are usually complex, and only partially uncovered through empirical investigation (Bhaskar, 1998a; Sayer, 2000, pp. 10-17). Detailed explication of identified generative mechanisms, at times, involves the use of words appropriated from the English language by critical realism to connote specific philosophical concepts, making the intended meaning ambiguous. The complex generative mechanisms detailed in this chapter involve changes, and temporality (Archer, 1982; Archer, 1995, pp. 149-161, 165-194; Archer, 2000). Consequently, the term “process” is used to convey their complete description. But “process”, in critical realism, refers to *“a relation between a transfactually acting power and its consequents”* (Fleetwood, 2009) which may or may not be observed or have observability (Bhaskar, 1998a; Fleetwood, 2011). To ameliorate this ambiguity, I have employed a linguistic strategy of using “process” to delineate the *make-up* of generative mechanisms, and “event” will continue as before to connote the outcome of an actualised generative mechanism.

6.2 Basic Identification of Generative Mechanisms

6.2.1 Cultivation of the Installed Base

Although the specifics of how SDOs led cultivation of the installed base (Monteiro, 1998; Hanseth & Lyytinen, 2010; Aanestad & Jensen, 2011; Sanner, et al., 2014; Grisot, et al., 2014) of network operators, network systems integrators, and network equipment vendors towards deploying SDN infrastructures was not given dedicated study in this research due to data access restrictions, their role in the advent of SDN infrastructures was repeatedly highlighted in the preceding chapter. It was found that SDOs created specifications for protocols, established and curated open standardisation processes, produced reference architectures and information models, persuaded vendors to make some of their network hardware devices SDN-compliant – particularly OpenFlow compliant – and developed and provided SDN software and frameworks for network operators' deployment in their networking infrastructures.

In confirmation of extant IS theorising, cultivation of the installed base then, is recognised as a social generative mechanism that features causally (though in the background of this research's case study) in architectural evolution in ossified digital infrastructures.

Abstracting and relating this to the research question, architectural evolution in ossified digital infrastructures, involves cultivation of the installed base.

Generative Mechanism: A Process of Installed Base Cultivation

Installed base cultivation operates to motivate key infrastructure innovators, owners and users to transition infrastructures towards new architecture through incremental processes of familiarisation, learning through evaluation, and eventual deployment (Monteiro, 1998; Hanseth & Lyytinen, 2010; Aanestad & Jensen, 2011; Sanner, et al., 2014; Grisot, et al., 2014).

6.2.2 Technical Disaggregation

A major objective of the second morphogenetic cycle was to create conditions for cost-effective replacement and interchangeability of vendors' network hardware devices in network operators' networks. Software-definition of generified networking infrastructure hardware can be understood as the technical strategy that partially facilitates this objective. One aspect of software-definition seen in the structural elaboration of traditional networks from an architectural perspective, is the decoupling of functions that have traditionally defined the main responsibilities of routers. Logically stratified into forwarding and control planes, the tight coupling of these functions within the confines of a network hardware device, is closely associated with the value attributed by vendors to the control plane. Deemed network intelligence, the exercise of control over this control plane by vendors through the instrumentation of an architectural point of control, is technically subverted by a *disaggregation* of the tight coupling between the forwarding and control planes.

Abstracting and relating this to the research question, architectural evolution in ossified digital infrastructures such as in traditional networks, involves the identification of an ossified function and subsequent technical disaggregation of this function from whatever tightly couples it (see Figure 6-2).

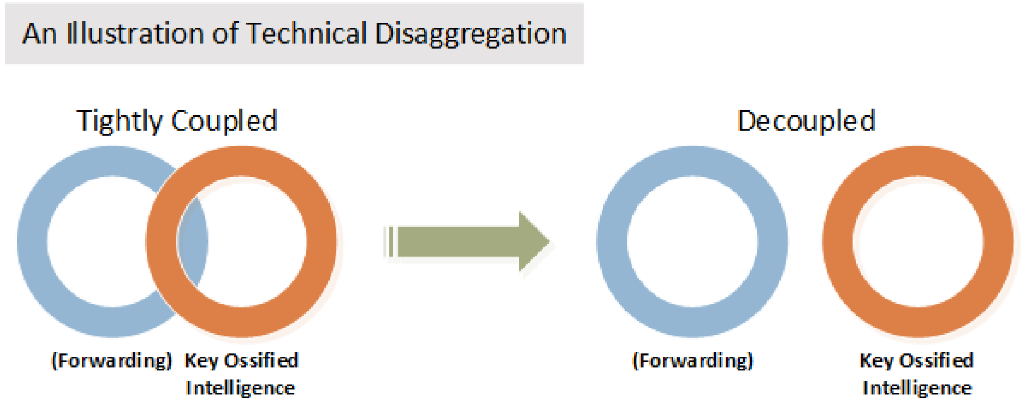

Figure 6-2 A Process of Technical Disaggregation

Generative Mechanism: A Process of Technical Disaggregation

Technical disaggregation operates to identify, and to decouple key ossified intelligence from whatever tightly couples it.

6.2.3 Intelligence and Authority Migration from Hardware to Software

Recall that from an architectural perspective, following the structural elaboration of traditional networks, SDN controllers within the resulting SDN infrastructures hold a consolidation of the control planes of routers in a network. They can be understood as holding a logically centralised aggregation of network intelligence. In traditional networks, semi-autonomously acting hardware routers encapsulate network intelligence, but it is the role of externally applied software to hold network intelligence in SDN infrastructures. Software-definition of generified networking hardware suggests that hardware is subjugated by externally applied software to indifferently carry out the actions that correspond to and fulfil the corresponding externally applied network intelligence. Regardless of how SDN's core architecture is instantiated by a particular network operators' SDN infrastructure (as discussed in section 5.4.8 on variations in SDN infrastructures), the subjugation of hardware indicates a yielding of embedded local control in deference to the authority of the externally applied software. It is a relinquishing of decision making and acceptance of externally constituted directives created by non-local intelligence. What this describes is a moving of network intelligence and authority away from network hardware devices to externally applied software.

Several interviewees discussed the theme of the transformation of traditional networks into SDN infrastructures as involving a *migration* of network intelligence from hardware to externally applied software: For example, Interviewee **I13**, Head of Architecture in a SDO, stated:

> *"This carries a bit back into something else that people also usually raise when they are talking about SDN, which is basically taking out the intelligence from the network element and moving them somewhere to some other place in the network."* – ***Interviewee I13***

Similarly, Interviewee **I24**, Director at a SDO, commented:

> *"There are different layers, but if you push the innovation to the highest level, infrastructure is a piece of - is composed by pieces of 'stupid' hardware.*
> *For example, **[Anonymised Detail]** is doing something like that. It's trying to move all the function of the radio base stations from the base stations to the cloud."* – ***Interviewee I24***

As the final example, Interviewee **I17**, Senior Network Architect for SDN and NFV at an ISP, explained:

> *"Well, for instance, if we take the data centre and our own experience in—the evolution of hardware has been from, an Ethernet infrastructure that is really delivering both your data plane and your control plane and everything is embedded in the network and it's very difficult to control it.*
>
> *…*
>
> *So now what we have in the data centre is we have simple switches and they only do IP transport between the different elements in the data centre from an infrastructure perspective, but all per-customer intelligence, the per-customer configuration - all this has been moved away from the hardware infrastructure, this is really residing in the virtual world and the virtual switches and this is what we control using SDN."* – ***Interviewee I17***

Abstracting and relating this to the research question, architectural evolution in ossified digital infrastructures, involves the migration of network intelligence from its network hardware device encapsulation to externally applied software (see Figure 6-3).

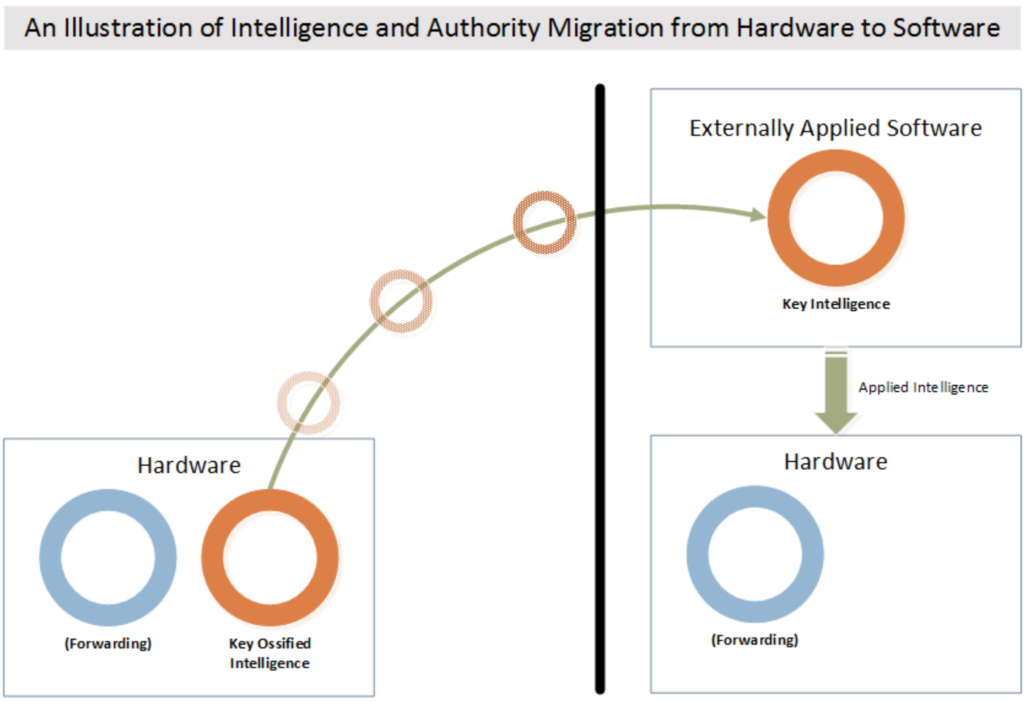

Figure 6-3 A Process of Intelligence and Authority Migration from Hardware to Software

Generative Mechanism: A Process of Intelligence and Authority Migration from Hardware to Software

Intelligence and Authority Migration operates to relocate decoupled intelligence from its hardware confinement to externally applied software.

6.2.4 Infrastructure Virtualisation

Recall that virtualisation techniques are a key enabler of SDN infrastructures, that there is no minimum virtualisation or precise prescription on exactly what should be virtualised, and that varying virtualisation techniques have been used by network operators at different levels of their SDN infrastructures. The reason for the presence of virtualisation technologies in SDN infrastructures is that there are specific requirements of software-defined networking that are technically achieved only through the use of these technologies – which continue to evolve and mature.

More abstractly than specific details of technical instantiation such as the use of particular virtualisation approaches (like hypervisors, network overlays etc.), virtualisation can be understood conceptually as a shift from the physical to the *logical* *and virtual*. While the logical connotes abstraction (which may be manifested in physical hardware, e.g. as with logical sub-divisions of physical routers[44]), the virtual is a software-*manifested* logical abstraction. Speaking on the shift towards the virtual, Interviewee **I14**, Distinguished Engineer at a large network equipment vendor and systems integrator, elaborated:

> *"So when I layout an SDN architecture or SDN components, I talk about packet forwarding and then I talk about routing software and then I talk about SDN control software, and those two are architectural components, but they are not physical components.*
>
> *Now this is not – understand, this is not revolutionary. When we've done architectures for a lot of things,* ***we talk about logical components which can be realised in various ways.***
>
> ***It is just more obvious when that realisation is virtual machines running software that gets loaded dynamically*** *- it becomes clearer, but it's actually been the way we prefer to talk about architectures for a long time.*
>
> *…*
>
> *Even infrastructure architecture,* ***we talked about it in terms of logical components and then physical realisation****.* ***<u>Well now there may be no physical realisation, that's all</u>.***
>
> *Even the forwarding stuff, some of the time it's going to be software." –* ***Interviewee I14****, (Bold and underline emphasis added.)*

Recall that the technical implementation of this shift from the physical to the logical and virtual typically includes the involvement of a container software entity that is responsible for administering the lifecycle of virtual entities (Smith & Nair, 2005;

---

[44] Partitioning of physical routers into logical systems is a pervasive feature of commercially available routers.

Rosenblum & Garfinkel, 2005; Zhang, et al., 2010; Pearce, et al., 2013; Medina & García, 2014), and that the SDN controller fulfils this role for programmatically defined virtual networks. The SDN controller, though already software, can be placed into an additional virtualisation context, namely to run within a virtual machine. Recall too that SDN infrastructures can be implemented purely in SDN controllers and software routers.

Abstracting all of this, networking infrastructures can be defined entirely in terms of the logical and virtual, or more simply – the virtual. That is, as purely *virtual infrastructures* in which, software manages the constitution of the infrastructures (see the third type of virtual networks listed in section 5.4.7.) Virtual infrastructures are an *outcome* of the *underlying* shift from the physical to the logical and virtual. To be explicit, I am saying here, that the shift from the physical to the logical and virtual, describes a sociotechnical generative mechanism that resides in critical realism's domain of the real. The shift is a sociotechnical interaction between network operator implementers of SDN infrastructures and the digital materiality of existing traditional networking infrastructures. I elaborate the role of digital materiality in architectural evolution later.

Generative Mechanism: A Process of Infrastructure Virtualisation

Infrastructure Virtualisation operates to fundamentally reframe infrastructure, via a leveraging of its facilitating digital materiality, in terms of the virtual.

6.2.5 Infrastructure Abstraction

Closely associated with the Infrastructure Virtualisation generative mechanism, which moves from the physical, to the logical, connoting abstraction, and the virtual, connoting software-manifested logical abstraction, is indeed, abstraction. It is difficult to untangle the two, but necessary to separately articulate them because they are distinct.

Abstraction can exist in the absence of virtualisation. Two points drawn from Chapter 5 help to clarify. The first comes from the objectives of the morphogenesis of the networking industry, which aim to make possible the cost-effective replacement and interchangeability of vendors' network devices in network operator's networking infrastructures. This is partially accomplished through logical homogenisation of hardware via southbound interfaces such as OpenFlow. Here, as well, is a bypassing of some traditional slow network protocol standardisation processes that are hindered by sociotechnical ossification, by relying on application level interfacing, i.e. via APIs, to provide some of the function of homogenising heterogeneous network devices that standardised network protocols do. The second comes from the morphogenesis of traditional networks, which aims to render the networking infrastructure as a pool of networking capacity, that is, as a service, to client applications. This is partially accomplished through standardised open APIs offered at the northbound interfaces of SDN controllers that hide the details of the underlying physical network, allowing API-mediated interfacing only with network abstractions. In both instances, the outcome of abstraction resides *at the interfaces*, which themselves may not necessarily be specifically virtualised[45].

Generative Mechanism: A Process of Infrastructure Abstraction

Infrastructure Abstraction operates to identify, and logically consolidate and homogenise key sources of infrastructure heterogeneity.

---

[45] That the SDN controller is virtualised does not mean that its northbound interfaces are virtualised. API virtualisation resides in a specific problem space of computing. It is a distinct undertaking. As an example, some public cloud service providers such as Amazon Web Services and Microsoft Azure provide API gateways that are virtualised (Amazon Web Services, n.d.-b; Microsoft, n.d.).

6.2.6 Summary of Basic Identification of Generative Mechanisms

The preceding basic identification of generative mechanisms isolated one social generative mechanism, and four sociotechnical generative mechanisms:

1. **A Process of Installed Base Cultivation**: operates to motivate key infrastructure innovators, owners and users to transition infrastructures towards new architecture through incremental processes of familiarisation, learning through evaluation, and eventual deployment.
2. **A Process of Technical Disaggregation:** operates to identify, and to decouple key ossified intelligence from whatever tightly couples it.
3. **A Process of Intelligence and Authority Migration from Hardware to Software:** operates to relocate decoupled intelligence from its hardware confinement to externally applied software.
4. **A Process of Infrastructure Virtualisation:** operates to reframe infrastructure, via a leveraging of its facilitating digital materiality, in terms of the virtual.
5. **A Process of Infrastructure Abstraction:** operates to identify, and to logically consolidate and homogenise key sources of infrastructure heterogeneity.

Aside from the clarifying point in the introduction that the generative mechanisms are not monolithic but complex processes, the four sociotechnical generative mechanisms should not be understood as simply a detailing of technical design processes. A non-trivial undertaking of cultivating a highly ossified installed base is a necessary facilitating substratum of these sociotechnical generative mechanisms. Further, network operators have not been designing SDN as an innovation as though a customer product offering. They have not been simply purchasing and installing SDN products into pre-existing networking infrastructures to effect an automatic upgrade of those infrastructures. They have not even been implementing SDN infrastructures for the sake of having new technological features in their networks.

As explained in Chapter 5, motivated by constraining conditions of structuring, SDN as an innovation has been appropriated from pre-existing clean-slate academic research devoid of production networking infrastructure complexity, to play a role in wider social objectives for the networking industry. Even as network operators architecturally evolve their traditional networks, they strategically decide which routers from which vendors should devolve network intelligence and authority to the externally applied SDN controller. That is, they decide which vendors' solutions should be the targets of logical homogenisation, and thus, how vendors become involved in the morphogenesis of the networking industry.

Two illuminating quotes from Interviewee **I34**, Technology Specialist at Tier 1 ISP, and Interviewee **I16**, Research Manager at a large systems integrator, convey this dynamic:

> *"Currently SDN has the potential of really driving down costs.*
>
> *…*
>
> *That's one of the things and the other big thing is that **it's a way of threatening the usual suspects and threatening their current comfort situation** so that you are pushing their prices down, you are pushing the infrastructures towards more the mode you want and it's not that you have your infrastructure with the model they want." – **Interviewee I34,** (Bold and underline emphasis added.)*
>
> *"I would say again, it's - I think it's more driven from the business requirements rather than purely from a need to change architecture or whatever.*
>
> *There's solid business rationale behind these types of technologies and I think it's that that's certainly driving— because technology has been around for years, to some extent." – **Interviewee I16***

Further, as explained in Chapter 4, the research sought to identify morphogenetic generative mechanisms. The previous sections are not merely a detailing of a technical design process. They describe a diachronic social interaction underpinned by a process of installed base cultivation, of which the outcome is morphogenesis. Archer refers to the modification of necessary internal relations of structures as being accomplished through social interaction (Archer, 1995, pp. 167-168), and simultaneously states that what plays out from an analytically isolated temporal point of structural conditioning until a corresponding analytically isolated later temporal point of structural elaboration, *is a generative mechanism of morphogenesis* (Archer, 1995, p. 217).

Preliminarily, Technical Disaggregation, Intelligence and Authority Migration from Hardware to Software, Infrastructure Virtualisation, and Infrastructure Abstraction *are the social interaction* that intervened between anterior structural conditioning and the subsequently transformed network operators' networking infrastructures. These are morphogenetic generative mechanisms.

These generative mechanisms operated to transform architecture (which is primarily technical) and I show in the development of this chapter that they necessarily exploited an inherent capacity of digital infrastructures to bring about architectural transformation, making them morphogenetic *sociotechnical* generative mechanisms. I provide additional explanation of their sociotechnical designation, and explain why they are themselves emergent, at the end of Chapter 7. Later in this chapter, I explain that these morphogenetic sociotechnical generative mechanisms, when in a particular necessary synchronic arrangement (Elder-Vass, 2007), gave rise to the emergent generative mechanism of morphogenesis by which traditional networking infrastructures were transformed to yield SDN infrastructures. I then explain that these same generative mechanisms when in a refinement of this synchronic arrangement have been implicated in the morphogenetic cycle of the networking industry. Network operators routinely innovate their internal networking infrastructures, but without connection to the modification or introduction of necessary internal relations and relata of the networking industry. Technical Disaggregation, Intelligence and Authority Migration from Hardware to Software, Infrastructure Virtualisation, and Infrastructure Abstraction, as underpinned by Installed Base Cultivation and explicated within this chapter, are therefore morphogenetic sociotechnical generative mechanisms.

Thus, the four extracted and isolated generative mechanisms of architectural evolution are to be understood as being fundamentally sociotechnical, operating to create new elaborated structures that subsequently condition a later sociotechnical context.

Still, according to critical realism, operating generative mechanisms may interact, and this interaction may give rise to new emergent generative mechanisms that are more than the resultant sum of the interactions. As well, accurate attribution of causality to mechanisms can be challenging in open systems. Therefore, the preceding basic identification of mechanisms is not enough to convince that these are sufficiently causally related to architectural evolution in sociotechnically ossified digital infrastructures. It must be supplemented with identification of generative mechanism interactions that are contingently causal to what arises, and identification of any other generative mechanisms that emerge from the interaction and synchronic relations of other generative mechanisms.

The chapter next proceeds with a gradual unfolding of the generative mechanisms identified (aside from Installed Base Cultivation), to address the identification of generative mechanism interactions, emergence of new generative mechanisms with elaborated structures, and to confirm causal attribution to generative mechanisms. For recognisability, I capitalise the names of generative mechanisms going forward.

6.3 Explicating the Generative Mechanisms

6.3.1 Relating the Sociotechnical Generative Mechanisms

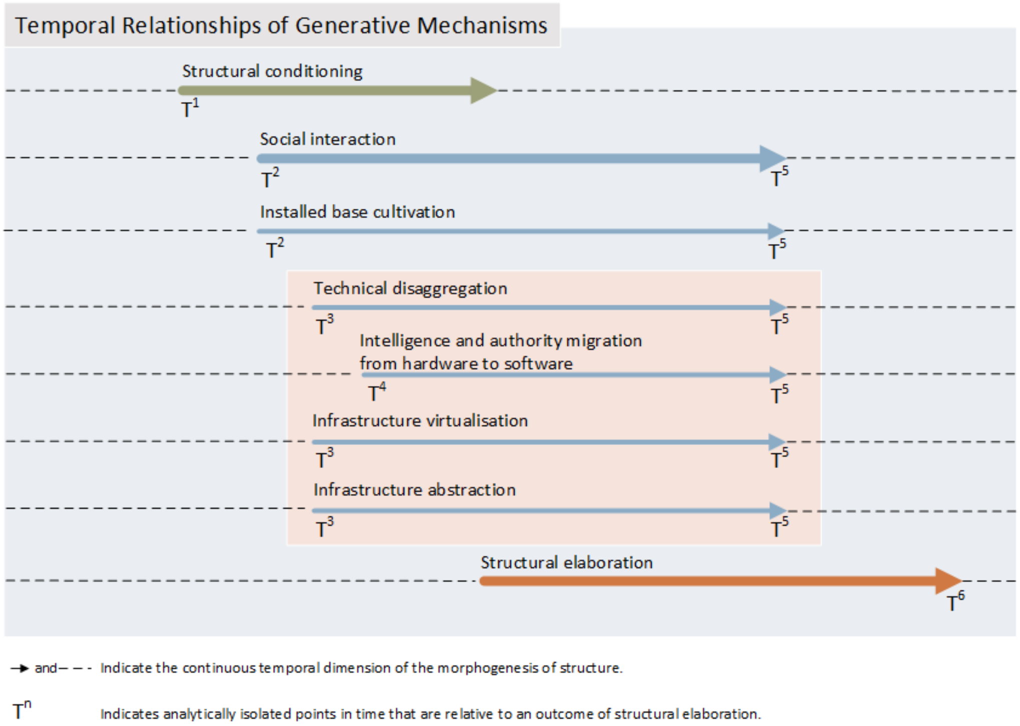

Figure 6-4 Temporal Relationships of Generative Mechanisms

Technical Disaggregation must identify, and decouple key ossified intelligence *before* Intelligence and Authority Migration from Hardware to Software *can* occur. The two generative mechanisms are temporally ordered, but it is only cumulatively that they contingently lead to architectural evolution with respect to routers' implementation architecture. To explain using an illustrative example, network operators may use the IETF's Forwarding and Control Element Separation (ForCES) (Halpern & Salim, 2010; Doria, et al., 2010) to technically disaggregate and migrate intelligence and authority from network hardware devices to a centralised controller as per SDN's core architecture, (Verizon a large ISP network operator, for example, has used a ForCES-based product from Mojatatu in parts of its networking infrastructure (Radisys Corporation, 2016; Mojatatu, n.d.)), but ForCES permits network operators to technically disaggregate without intelligence and authority migration to externally applied software (Halpern & Salim, 2010; Doria, et al., 2010; Nunes, et al., 2014) –

which does not lead to architectural evolution with respect to routers' implementation architecture. So in the context of architectural evolution, the relationship between these generative mechanisms is temporally ordered and cumulative. This is reflected in Figure 6-4, where Technical Disaggregation begins to operate at an analytically isolated temporal point in time T^3 (subsequent to Installed Base Cultivation at T^2), followed by Intelligence and Authority Migration from Hardware to Software at an analytically isolated temporal point in time T^4, in the social interaction phase.

The relationship between Infrastructure Virtualisation and other generative mechanisms is more challenging to specify, particularly considering that what lies in between the physical, and the logical and virtual, are complex processes carried out by virtualisation experts (seeking to realise business objectives of network operators) of which the outcome is software-manifested logical abstraction. The generative mechanism is not causally dependent on Technical Disaggregation and Intelligence and Authority Migration and may act transfactually (i.e., producing intermediary artefacts that are not immediately actualised in operational production infrastructures) in parallel with these generative mechanisms. Likewise, the practicality of the Infrastructure Abstraction mechanism is that it is a complex process carried out by networking and infrastructure professionals guided by business objectives to identify what heterogeneity is to be homogenised, aspects of which may occur in parallel with the outcomes of other generative mechanisms. It is a complex generative mechanism that may act transfactually in parallel with other generative mechanisms. As such, the Infrastructure Virtualisation and Infrastructure Abstraction generative mechanisms are depicted in Figure 6-4 as beginning to operate at the same analytically isolated temporal point in time T^3, at which Technical Disaggregation begins to operate.

What is certain on the basis of the basic identification of the mechanisms from the findings of Chapter 5, is that all four sociotechnical mechanisms contribute to architectural evolution by which SDN infrastructures come into being from network operators' traditional networking infrastructures. Notwithstanding the various relationships between individual generative mechanisms, it is the cumulative actualisation of all, that are contingently causal to architectural evolution in sociotechnically ossified digital infrastructures. It is by the cumulative actualisation of

Technical Disaggregation, Intelligence and Authority Migration from Hardware to Software, Infrastructure Virtualisation, and Infrastructure Abstraction generative mechanisms that, from an architectural perspective, traditional networks are incrementally elaborated and transformed into SDN infrastructures. In Figure 6-4, the cumulative outcome of the operation of the generative mechanisms is indicated at the analytically isolated temporal point in time T^5, though as per Archer's model (Archer, 1982; Archer, 1995, pp. 157,193-194), incremental structural elaboration begins prior. The architectural evolution is of the sociotechnically ossified implementation architecture of routers around which the networking industry is arranged, and it occasions core architectural evolution (i.e., the Internet's core architecture). This is partially explanatory, since, there are interactions with an additional social generative mechanism associated with the morphogenesis of the networking industry. I discuss that generative mechanism later.

These four sociotechnical mechanisms promote architectural evolution by which SDN infrastructures come into being from traditional networking infrastructures, but there remains need for an explanation of emergent characteristics of SDN infrastructures that qualitatively distinguish them from traditional networking infrastructures. These are characteristics of dynamically tailored infrastructure, and flexibility and accommodation of change. The research question seeks generative mechanisms, and given the challenge of attributing causality, critical realist research involves a search for alternative explanations (Sayer, 2000, pp. 13-17; Easton, 2010; Wynn & Williams, 2012). Therefore, it is necessary to clarify here whether these distinguishing characteristics were found to be the outcome of an already identified generative mechanism, interactions between generative mechanisms, an emergent generative mechanism, or a generative mechanism unrelated to those already identified, and how the causally responsible generative mechanisms are as well implicated in architectural evolution in sociotechnically ossified digital infrastructures.

6.3.2 Formalising the Ontology

Before proceeding to explain the emergent characteristics of SDN infrastructures, a preliminary critical realist formalisation of the ontology of SDN infrastructures is presented[46]. I revisit this formal ontology at the end of the Discussion chapter. Chapter 5 explains without theoretical treatment, the structure and properties of SDN infrastructures. A critical realist articulation formalises this into an ontology of structure, properties and mechanisms. To reiterate, the ontology to which this analysis subscribes is one of a unity of structure, its properties and generative mechanisms, such that they emerge simultaneously and none is more primary than any of the others (Fleetwood, 2009).

The structure in the ontology is the instantiated SDN infrastructure. It is both social and artefactual making it a sociotechnical structure. *Having* decoupled forwarding and control planes, *having* a consolidated centralised software-based network control plane, *being* programmable with intrinsic abstraction of the network as a service to client applications, *being* technically open, *being* intrinsically permeated by virtualisation are *properties* of SDN infrastructures. Put differently, SDN's core architecture constitutes the properties of SDN infrastructures. Archer's concept of structural emergent properties (Archer, 1995, p. 177), lends support for characterising architecture – with its technical relations and relata – as properties[47]. Generative mechanisms, complete the critical realist ontology of SDN infrastructures. These generative mechanisms which emerge with SDN infrastructures are identified next.

6.3.3 Analysing the Emergent Characteristics of SDN Infrastructures

The characteristic of dynamic [application-specific] tailoring of infrastructure is not a social capability. It is a technical capacity. It is not a pre-defined technical function to

---

[46] Importantly, this is ontology with respect to architectural evolution. I am not asserting that the entirety of the ontology of SDN infrastructures relate to architectural evolution. It is possible that generative mechanisms not related to architectural evolution in ossified digital infrastructures may exist but remain unreported because of the research's focus.

[47] Archer defined structural emergent properties as *"those internal and necessary relationships which entail material resources, whether physical or human, and which generate causal powers proper to the relation itself"* (Archer, 1995, p. 177).

be invoked. Understood as a capacity, it connotes an intrinsic ability to accommodate or to bring about a possibility that emerges when SDN infrastructures instantiate the core SDN architecture. Through the analytic lens of critical realism, the intrinsic capacity for dynamic [application-specific] tailoring of infrastructure is a technical generative mechanism of instantiated SDN infrastructures[48]. For clarity, by mentioning technical capacity, I am not equating generative capacity (Zittrain, 2008) with generative mechanisms. Generative mechanisms are about what intransitives are entailed in reality, whereas generative capacity refers specifically to a capacity to accommodate or to occasion unanticipated change. I am distinguishing between what is social and what is technical in order to make clear the generative mechanisms involved in architectural evolution in ossified digital infrastructures.

As with dynamic [application-specific] tailoring of infrastructure, flexibility and accommodation of change is not a social capability. It is a technical capacity. It connotes an ability to accommodate or to bring about a possibility. Intrinsic accommodation of much greater degrees of layering ambivalence than traditional networks is not a technical function; it is a technical capacity of SDN infrastructures. The flexibility and accommodation of change is a technical generative mechanism of instantiated SDN infrastructures.

The same logic is valid for identifying the intrinsic ability to introduce capabilities to generified networking hardware via software-definition (see section 5.4.8) as being a technical generative mechanism of instantiated SDN infrastructures.

Very importantly, there is a distinction between the sociotechnical generative mechanisms of architectural evolution that yield SDN infrastructures as elaborated traditional networks, and the technical generative mechanisms of the instantiated SDN infrastructures. The sociotechnical generative mechanisms that transform traditional networks are causal to the emergence of technical generative mechanisms that are

[48] Generative mechanisms designate *"what something does, or can do"* (Fleetwood, 2009). Thus, though technical capacities they qualify as being generative mechanisms.

instantiated simultaneously with SDN infrastructures (and properties). At the technical generative mechanisms' emergence, they are devoid of a social aspect, however as alluded to in this chapter's introduction they may actualise to contribute towards continued architectural evolution in SDN infrastructures (which I discuss later). When actualised for the purposes of continued architectural evolution in network operators' SDN infrastructures, these mechanisms are initiated into a causal role within a broader sociotechnical context.

So there are three additional generative mechanisms. They are technical, and they emerge simultaneously with SDN infrastructures (and properties). They are emergent generative mechanisms that distinctively characterise and differentiate SDN infrastructures.

1. **Technical Generative Mechanism:** Technical Capacity for Dynamic Tailoring of Infrastructure
2. **Technical Generative Mechanism:** Technical Capacity for Flexibility and Accommodating of Change
3. **Technical Generative Mechanism**: Technical Capacity for Introducing Capabilities to Generified Hardware via Software-Definition

Four morphogenetic sociotechnical generative mechanisms have been identified as promoting architectural evolution that transforms traditional networks into SDN infrastructures. SDN infrastructures that come into being are instantiated simultaneously with three technical generative mechanisms that may actualise to further continued architectural evolution in SDN infrastructures of a different kind. In what has been presented to this point, the four sociotechnical generative mechanisms do not fully explain why all characterising technical generative mechanisms that qualitatively distinguish SDN infrastructures from traditional networks emerge. To address this, I next explicate the four sociotechnical generative mechanisms through a search for alternative explanations (Sayer, 2000, pp. 13-17; Easton, 2010; Wynn & Williams, 2012) to the architectural evolution that transforms traditional networks into SDN infrastructures. The crucial role of digital materiality in this architectural evolution becomes explicit during this explanation.

6.3.4 Revisiting Technical Disaggregation, and Intelligence and Authority Migration from Hardware to Software

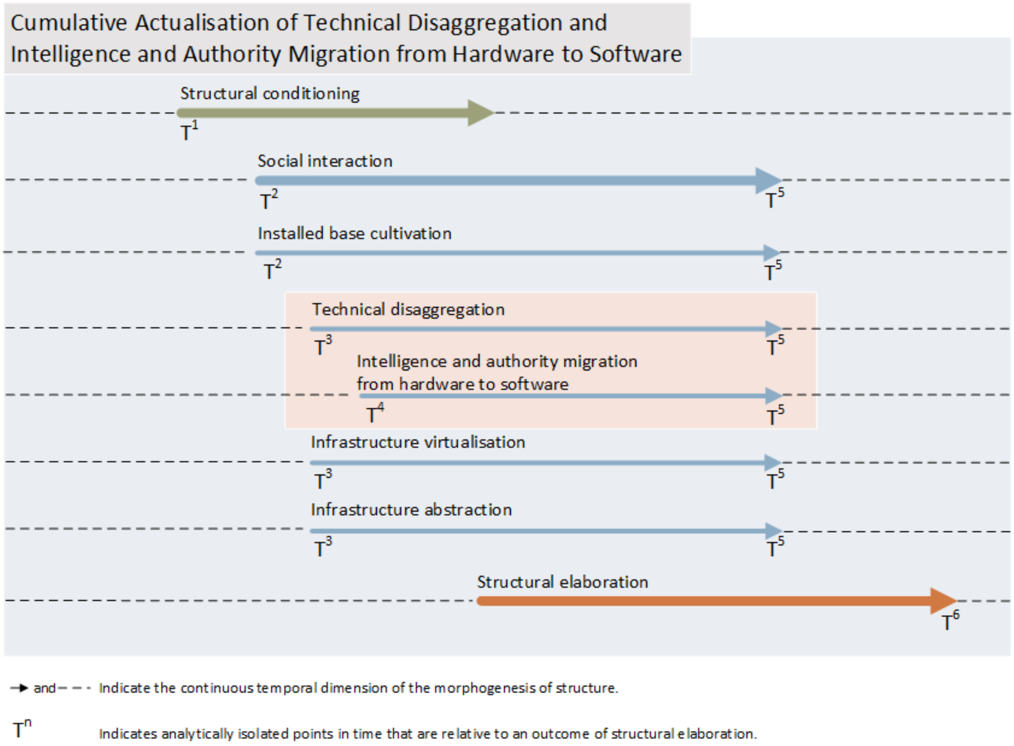

Figure 6-5 Cumulative Actualisation of Technical Disaggregation, and Intelligence and Authority Migration from Hardware to Software

Recall that in the context of architectural evolution, the relationship between the generative mechanisms of Technical Disaggregation and Intelligence and Authority Migration from Hardware to Software is temporally ordered and cumulative (see also Figure 6-5). Recall as well, that there is a preferred functional perspective of infrastructure that describes infrastructure's aggregate functionality as being physically and logically distributed across functions contained within or manifested as hardware or software, and whose demarcation logic designates the locus of infrastructure intelligence as being infrastructure. Thirdly, in Chapter 2, it was explained that the digital materiality of the technical constitution of digital entities, can be articulated in terms of the abstract elements of form, function and matter, and that digital infrastructures, such as networking infrastructures are technologically constituted by digital materiality, and permeated by binary signification.

Disaggregation and migration of network intelligence indicate the ability to identify, isolate, and disassociate function from its existing bearer, whatever its bearer's form or matter, and to transfer that function equivalently elsewhere to another bearer. The ability to identify, isolate, and disassociate function from its bearer, as indicated by disaggregation and migration, is facilitated by the means by which function is embodied in digital infrastructures: through binary signification. Again, binary signification, allows function (i.e., what actions must be carried out) in digital infrastructures to be encoded in abstraction from whatever bearer eventually executes it (Blanchette, 2011; Berry, 2011, pp. 94-97). Thus, binary signification is implicated in a looser coupling within digital infrastructures, of relationships between form, function, and matter (Kallinikos, 2012), in that the definition of function can be decoupled from and be indifferently transferrable between different bearers of its execution.

6.3.4.1 Digital Materiality in Traditional Networks and in SDN Infrastructures

So what are the relationships between form, function and matter, with respect to routers' implementation architecture, in traditional networks prior to the actualisation of the generative mechanisms of Technical Disaggregation, and Intelligence and Authority Migration from Hardware to Software? Function (collectively forwarding and control), though digitally signified, is embedded within the bearer (i.e., the router). Matter that admits the representation of binary signification constitutes the router. Form organises and imposes structure over objects of form – the matter that admits binary signification of the defined function (Crawley, et al., 2016, pp. 70-81) – to effect a logical and physical stratification (a formal relationship also constituent of form (Crawley, et al., 2016, p. 68;77)) of the matter involved in forwarding and the matter involved in control (see also (Kurose & Ross, 2013, pp. 346-348)). More plainly, the internal physical components of a router are organised accordingly to what fulfils the forwarding function, and what fulfils the control function. Together they realise the de facto standard implementation architecture of routers.

And what are the relationships between form, function and matter, with respect to routers' implementation architecture, after the actualisation of the generative mechanisms of Technical Disaggregation, and Intelligence and Authority Migration from Hardware to Software? Part of the aggregate function (i.e., forwarding and control) remains in its original bearer (namely forwarding), but form is modified because an object of form is logically or physically removed (physical components that fulfil the control function in routers), modifying the formal relationships (Crawley, et al., 2016, pp. 68,77) of form. Whether logical[49] or physical, the matter involved in the carrying out of function (forwarding and control) changes. A subset of the original bearer is instrumental to the execution of the aggregate function (physical components that fulfil the forwarding function in routers). The other part of the aggregate function is relocated to a different bearer (i.e., the SDN controller) whose form and matter requires separate explication.

A comparison of the digital materiality that underlies traditional networks and derivative SDN infrastructures indicates more than the theme of digital infrastructure architectural evolution by interconnection widely promulgated in IS literature on digital infrastructure. Something significant takes place via an exploitation of digital materiality.

[49] Refer to the logical decoupling point in section 5.4.8.

6.3.4.2 Refactoring of Infrastructure

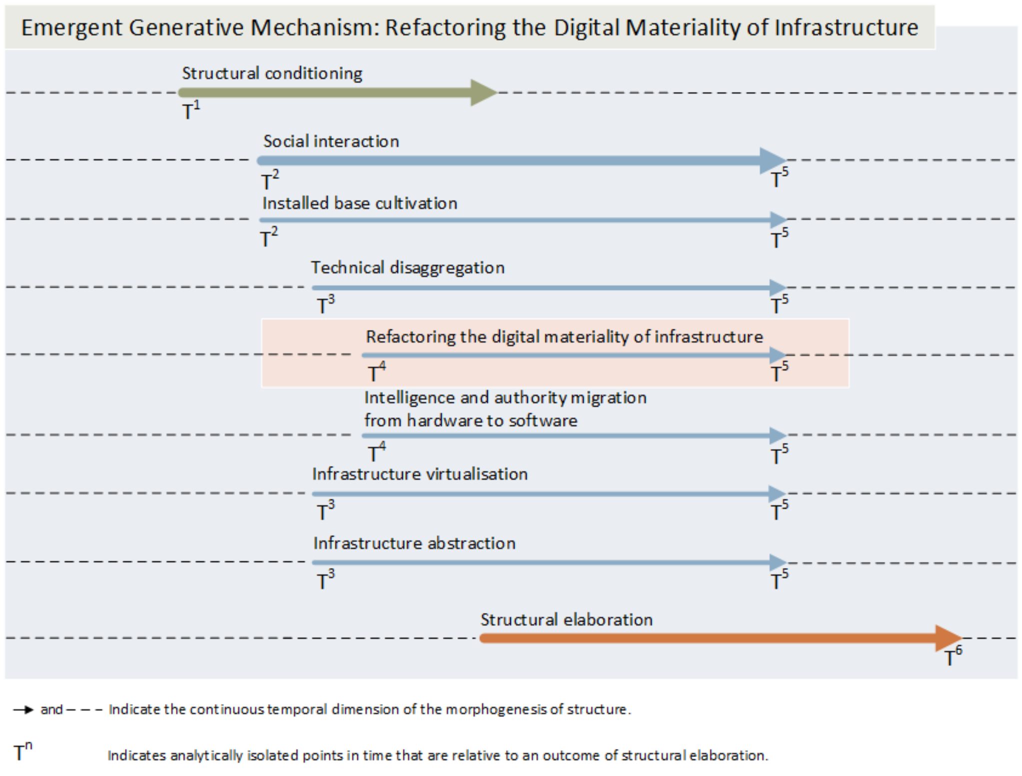

Figure 6-6 Refactoring the Digital Materiality of Infrastructure as an Emergent Generative Mechanism

The actualisation of Technical Disaggregation, and Intelligence and Authority Migration from Hardware to Software is causal to a transition in digital materiality of what constitutes network operators' networking infrastructures. They explicitly split networking infrastructures into a stratification of hardware and software. Taking the functional perspective of infrastructure that the *locus* of infrastructure intelligence is infrastructure, the two generative mechanisms interact to effect a meddling with the physicality of what is infrastructure to disaggregate some "infrastructure" out of physical form and matter and to migrate it into a "pure" software form and matter. This software form and matter bears the control function. It may be appropriate to borrow terminology from software engineering and to suggest that the two mechanisms interact to cause a *refactoring* of networking infrastructure.

In software engineering, refactoring reorganises the internal structure of software, while preserving externally observed function (Arnold, 1989; Chikofsky & Cross, 1990;

Tokuda & Batory, 2001; Mens & Tourwe, 2004). Given that the locus of infrastructure intelligence is infrastructure, refactoring changes the locus of the control function from hardware to externally applied software to invite the digital materiality of software into what is demarcated as infrastructure – while preserving the externally observed data transmission and interconnectivity functionality of the network.

The search for alternative explanations given the challenge of attributing causality, uncovered a generative mechanism that refactors the digital materiality of what is infrastructure. This mechanism actualises and operates in the cumulative interaction of the actualisations of Technical Disaggregation, and Intelligence and Authority Migration from Hardware to Software generative mechanisms. It does not exist if only one of the two mechanisms is actualised. Temporally, refactoring of the digital materiality of infrastructure begins to operate simultaneously with the operation of Intelligence and Authority Migration from Hardware to Software. Accordingly, in Figure 6-6, this new generative mechanism begins to operate at an analytically isolated temporal point in time T^4.

Mechanism: A Process of Refactoring the Digital Materiality of Infrastructure

Refactoring the Digital Materiality of Infrastructure operates to modify the matter and form that is instrumental to the execution of function.

6.3.4.3 The Form and Matter of Software

Three generative mechanisms actualise to transition traditional networking infrastructures through form and matter to yield SDN infrastructures. But what is the form and matter of software, the bearer of the control function in SDN infrastructures? To answer this question, I will employ a strategy of juxtaposing from extreme ends of the spectrum, extant thinking on what is the materiality of digital entities.

6.3.4.4 The Immateriality Perspective

The immateriality perspective attempts to respond to the seemingly ethereal qualities of software by asserting that software is composed of digital materiality that, relative to traditional notions of matter, should be characterised as immaterial. It argues that software then is immaterial (Leonardi, 2010; Kallinikos, 2012; Kallinikos, et al., 2012, pp. 8-9; Pentland & Singh, 2012, p. 289; Faulkner & Runde, 2013). It is argued that binary encoding of software makes its materiality an intangible logical one rather than a physical one (Leonardi, 2010; Kallinikos, 2012; Kallinikos, et al., 2012, pp. 8-9). The immateriality of software seems to have some support by observations on economic consequences of binary signification, where costs of perfectly reproducing software following initial production seem to vanish in contradistinction to the economics of physically reproduced products (Shapiro & Varian, 1999, pp. 20-26;84-85;93-102).

6.3.4.5 The Sociomateriality Perspective

Importantly, evaluating sociomateriality is not the focus of this research, but its juxtaposition with the immateriality perspective is illuminating for grasping the form and matter of software. To reiterate, studies of sociomateriality suggest an *"inseparability of meaning and matter"* (Scott & Orlikowski, 2014, p. 873), such that the digital entity itself is in some sense ontologically materially co-constituted in practice (Orlikowski, 2007; Orlikowski & Scott, 2008; Scott & Orlikowski, 2012; Scott & Orlikowski, 2014). Immateriality runs counter to the strongly material perspective of sociomateriality which obscures technology behind the generic term "material", to point out that certain digital technological entities – specifically software – are immaterial, and therefore are inevitably excluded or trivialised by generification of technology as "material" in a traditionally derived understanding of matter (Kallinikos, 2012). Certainly, there are scholars who have tried to reconcile the two perspectives (Leonardi, 2013), but sociomateriality does not focus on binary signification and the technical constitution of digital entities – neither is it heavily premised on these for any of its conclusions (Orlikowski, 2007; Orlikowski & Scott, 2008; Scott & Orlikowski, 2012; Scott & Orlikowski, 2014). Nonetheless, sociomateriality argues that software is not immaterial, but instead is reducible to some physicality-based "matter" manifestation (Scott & Orlikowski, 2014, p. 879). The immateriality perspective, however, argues that an emphasis on the connections that software ultimately maintains with hardware could prohibit a deeper understanding of the implications of software's foundation in mathematics and logic (Kallinikos, et al., 2012).

6.3.4.6 Shortcomings of the Sociomateriality Perspective

Both perspectives carry valid points on the form and matter of software, and yet there are significant shortcomings. The objections to sociomateriality are rooted in the way that it formulates materiality on the basis of its ontological stance. For sociomateriality, materiality is constituted through enactment[50], and thus materiality is fundamentally intermixed and inseparable from the social (Scott & Orlikowski, 2014). Materiality is a *"process of materialization that configures reality"* (Scott & Orlikowski, 2014, p. 879). Prior to sociomateriality's formulation, Archer argued from a critical realist perspective that co-constitutive instantiation of materiality is ontologically unsound (Archer, 1982; Archer, 1995, pp. 135-161; Archer, 2000), rendering several of sociomateriality's assertions on materiality fundamentally flawed. Other points of debate can be found in these articles (Mutch, 2013; Kautz & Jensen, 2013; Leonardi, 2013), but the conclusion is that the form and matter of software is not adequately accommodated by sociomateriality's formulation of materiality.

6.3.4.7 Shortcomings of the Immateriality Perspective

While it can be argued that digital materiality is different from traditional notions of matter, and it may seem reasonable to accept an immateriality constituted of logic

---

[50] To avoid ambiguity, the generative mechanism of Refactoring the Digital Materiality of Infrastructure which transitions infrastructure through form and matter does not corroborate sociomateriality's assertion that materiality as a process of materialisation through enactment. The mechanism is a process over pre-existent form, matter and binary signification, not accommodated by sociomateriality's "lived time" (Scott & Orlikowski, 2014, p. 878) rooted in its relational ontology. Binary signification and function as the object of its encoding, substantively pre-exists in a manner that transcends any local context of enactment (Kallinikos, et al., 2013).

rather than of physicality, a convincing argument cannot be made that software is truly immaterial. Notwithstanding the fundamental epistemological and ontological problems, an observation in sociomateriality that software is ultimately traceable to some matter, remains valid. Drawing on Chapter 2, the control function software in routers is stored in ROM or Flash Memory, and on execution is transferred to more volatile memory. Its definition is expressed through binary signification, ultimately represented in hardware as changing electric voltages or as physical structures (e.g., with ROM). More than this, *within* the definition of function remain the constraints of physical digital entities that execute it (Blanchette, 2011). The detection and resolution of errors originating in physical device technical failure and omissions, are written into the definition of function. These error and failure mitigation techniques hide the ways in which software remains tied to the constraints of the physical, and has left open arguments which (Blanchette, 2011) citing (Kirschenbaum, 2008) explores as the *"illusion of immateriality"*. The pervasive software engineering procedure of optimising algorithms and software structure, to contend with the limitations of the technical performance of hardware, is yet another counter-argument to the immateriality perspective.

Considering the arguments for the immateriality perspective, the issue seems to be that <u>*binary signification **of** form is being conflated with matter*</u>. Explaining the form of software, the authors Crawley, et al. (2016) write:

> *"Therefore, in software, the code (or pseudocode) is the form: It exists, it exists [sic] prior to the execution of function, and it is instrumental of function. It is implemented (written). When it is operated, this form is interpreted as an instruction that, when executed, leads to function."* –
> ***(Crawley, et al., 2016, p. 92)***

The form of software is *code*. It is an informational form (see section 2.5.3). The facilitator of the form is binary signification. I have taken care to avoid using previously the term "binary" in isolation in this analysis, so that it is clear that binary is a signification system, independent of that which it is used to signify. Binary signification is used to capture the definition of function – in the form of code. It assists a loosening of couplings between function and matter, rendering function an informational entity.

It assists a loosening of coupling which makes form (code) agnostic (to some extent) to the particular instance of processing hardware (matter) to which it is assigned. That binary is a signification system, does not mean *that* which it signifies is immaterial. For software, the relationship between form and matter may be different from physical entities, but the form of software is bound to matter that admits the representation of binary signification[51].

> *"There is a subtle difference between defining abstractions of software and information systems and defining those of a physical system, in that information itself is an abstraction. Informational form must always be stored or encoded in some physical form, and the two are a duality." –* ***(Crawley, et al., 2016, p. 92)***

6.3.4.8 Re-articulating the Form and Matter of Software

In explaining the transition of networking infrastructures through form and matter via the three generative mechanisms being explicated, I'm using the arguments from extreme ends of the spectrum on the materiality or immateriality of software to draw out what might be an issue of articulation.

Software is more than descriptions of function carried by code. Software is not merely a series of logical instructions (Kallinikos, 2012, p. 77; Kallinikos, et al., 2012, pp. 8-9) that can be executed by computers. For these instructions, written down on paper or even stored in a digital file are not software in the fullest sense. Rather, they are [descriptions of function carried by] code, at some level of abstraction, that *tends towards executability* (Harman, 2010). The logical instructions become software in the

---

[51] For clarity, I am not stating that it is, through a process of enactment, co-constituted by that matter as advocated by sociomateriality. Code, the informational form of software, is inscribed onto pre-existing matter. The matter is not in any way constituted as a consequence of code being inscribed onto it. It is a temporally prior structure (Archer, 1982; Archer, 1995, pp. 135-161; Archer, 2000).

fullest sense only after being compiled or interpreted into directly executable binary signified low-level machine code (Smith & Nair, 2005; Harman, 2010; Berry, 2011, pp. 94-98), and made available to its hardware processor for instantiation and execution of the function defined.

Put differently, software inhabits two manifestations:

1. A dormant state as a ***potentially executable*** digital object (Kallinikos, et al., 2010; Harman, 2010; Kallinikos, et al., 2013), and
2. An ***executing*** state, i.e., software in the process of running, (Wegner, 1997; Harman, 2010; Berry, 2011, pp. 94-98; Crawley, et al., 2016, p. 92) made possible by continuous changes (such as in electric voltages) in the matter facilitating its representation.

A characterisation of the form and matter of software primarily articulated in terms of immateriality does not sufficiently capture these characteristics of software. At the same time, the distinctiveness of digital materiality cannot easily be collapsed into a generic term "material" with its traditional connotations as sociomateriality, at the other end of the spectrum, attempts. What I am drawing out here using the arguments on the immateriality or materiality of software, is that software is characterised by a dual state, which transcends the limitations of discussions framed primarily by refinements and perspectives on materiality. What seems to be missing is an additional unfolding of the term "digital materiality", specifically in the context of software.

6.3.4.9 Runtime Matters

Form is *"instrumental* ***in*** *the execution of function"* (Crawley, et al., 2016, p. 68), (bold emphasis added), but additionally in software the form itself is executed (Harman, 2010; Berry, 2011, pp. 94-98; Crawley, et al., 2016, p. 92). A complete understanding of software requires more than the somewhat static aforementioned conceptions of materiality. Software requires consideration of the *runtime execution of function (carried by form)*. In the context of software, digital materiality is refined to include an additional dimension: *runtime*.

The abstract elements of digital materiality, specifically in the context of software, then are: form, function, matter and ***runtime***.

Recall that on the basis of the functional perspective of infrastructure, that software which carries out any of the aggregate function of networks is infrastructure. Since in general software inhabits two manifestations, it follows that infrastructure that is software features the same characteristic in which runtime is a significant dimension of digital materiality. Therefore, there is a requirement to consider infrastructure not just as software, but as *executing software*. This goes beyond conceptualising runtime in an infrastructure running versus infrastructure not running sense. It importantly includes degrees of behavioural variety at runtime. I return to this later in section 6.3.7.

A key analytic observation of this elaborate unfolding of the form and matter of software, is that in the context of architectural evolution studied here, Refactoring the Digital Materiality of Infrastructure not only modifies the matter and form that is instrumental to the execution of function (i.e., from hardware to software), refactoring transitions the digital materiality of infrastructure towards executing software. Runtime matters in architectural evolution in traditional networks towards SDN infrastructures.

6.3.4.10 Refining the Refactoring Mechanism

Because “matter” and “materiality” are closely related terms, a definition of digital materiality that includes “matter” as one of its abstract elements seems recursive and ambiguous. Since digital materiality as distilled here pertains to the make-up of infrastructure, the term *“digital ontology”* is a more appropriate term[52].

---

[52] Some support for this terminology can be found in an article by Kallinikos, et al. (2013) who suggest an ambivalent ontology of digital objects.

Digital Ontology: Form, Function, Matter, and Runtime

Refined Generative Mechanism: A Process of Refactoring the Digital Ontology of Infrastructure

1. *Refactoring the Digital Ontology of Infrastructure operates to modify the matter and form that is instrumental to the execution of function.*

2. *Refactoring the Digital Ontology of Infrastructure operates to transition the ontology of infrastructure towards executing software at runtime.*

6.3.5 Summary of the Explication of Technical Disaggregation, and Intelligence and Authority Migration from Hardware to Software

To reiterate, the explication of the four sociotechnical generative mechanisms is a search for alternative explanations (Sayer, 2000, pp. 13-17; Easton, 2010; Wynn & Williams, 2012) to architectural evolution that transforms traditional networks to SDN infrastructures and to account for the three emergent technical generative mechanisms.

Analysis of the Technical Disaggregation, and Intelligence and Authority Migration from Hardware to Software generative mechanisms uncovered an emergent generative mechanism that actualises and operates in the cumulative interaction of both actualised mechanisms: Refactoring the Digital Ontology of Infrastructure. The analysis produced a corollary refinement of the notion of digital materiality, renamed digital ontology, such that, in the context of software, digital ontology importantly includes a runtime dimension. The emergent generative mechanism modifies the matter and form that is instrumental to the execution of networking infrastructures' aggregate function, and transitions the ontology of the infrastructure towards executing software. The explanation also identified distinctly, infrastructure as executing software, as important for understanding architectural evolution in ossified digital infrastructures.

6.3.6 Analysis of the Parallel Actualisation of Infrastructure Virtualisation

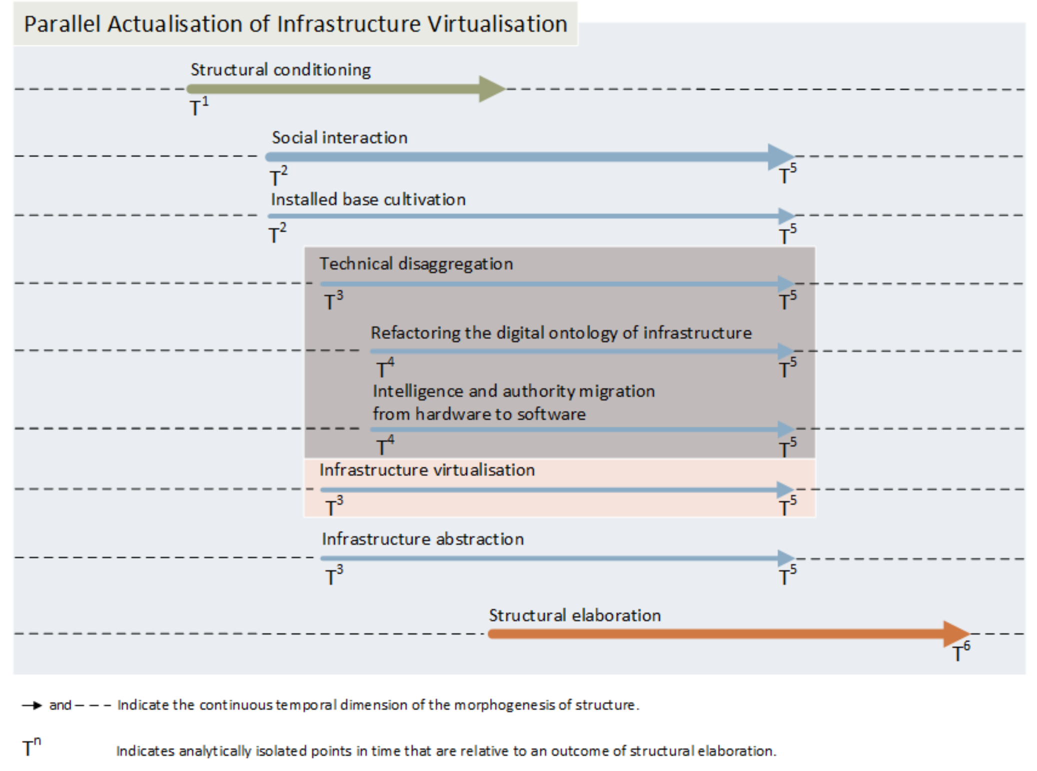

Figure 6-7 Parallel Actualisation of Infrastructure Virtualisation with Other Generative Mechanisms

To reiterate, SDN as an innovation facilitates the ability of network operators to introduce capabilities to networking hardware using software that is not coupled to the hardware. The Technical Disaggregation, Intelligence and Authority Migration from Hardware to Software, and Refactoring the Digital Ontology of Infrastructure generative mechanisms actualise to move the control function of routers to a consolidated network control plane implemented in a centralised software-based SDN controller. The SDN controller manages the lifecycle of programmatically created logical abstractions of the physical network, that is, virtual networks that are fully functional networks and may even be entirely virtual with no physical components, and is infrastructure – as executing software. The Infrastructure Virtualisation generative mechanism at some point operates in parallel with the three generative mechanisms, to fundamentally reframe infrastructure in terms of the virtual, via a leveraging of an infrastructure's facilitating digital ontology.

Continuing the explication of generative mechanisms, this section considers the parallel operation and interaction between the three generative mechanisms already explained (darkened overlay in Figure 6-7), and Infrastructure Virtualisation (highlighted in Figure 6-7) in order to establish whether these suitably explain architectural evolution and account for the technical generative mechanisms that emerge with SDN infrastructures, or whether there are more suitable explanations.

6.3.6.1 Runtime Revisited: Introducing Ephemeral Architecture

Chapter 5 explained that SDN infrastructures retain from the initial products of academic research, the ability to accommodate new protocols, and are ambivalent to traditional notions of layering. The SDN controller or the virtual networks that it manages can run networking protocols not natively supported by any underlying physical networking hardware. Protocol specified behaviour is applied to hardware via software-definition. Relative to the Internet's core architecture, a core architecture with different layering, protocols and operational semantics (such as the lack of network layer routing algorithms) can be implemented within the virtual networks. Speaking on the retained ability for doing this, Interviewee **I31**, Researcher in SDN, stated:

> *"So, one of the things that's been offered by SDN is - particularly on these layer three protocols -* ***is the ability to experiment with new protocols*** *to check if they work better and then to basically create a layer of the network…you can use the old Internet if you like, it's a best effort service, but* ***if we want to, provide, you know, capabilities with newer protocols or segregate another piece of network which will run these new protocols****..." –* ***Interviewee I31****, (Bold and underline emphasis added.)*

On the same topic, Interviewee **I17**, Senior Network Architect for SDN and NFV at an ISP, commented:

> *"The original definition of SDN as per the Open Networking Foundation, was really to have devices directly controlled by OpenFlow.* ***So really switches that don't run any routing protocols*** *and that have the*

forwarding tables configured using OpenFlow." – ***Interviewee I17****, (Bold and underline emphasis added.)*

Technical analysis of the capabilities of the OpenFlow protocol for instance, which is implemented in several vendors' physical routers as well as in software routers[53] and in commercial and open source SDN controllers, revealed that as standard operation, (in protocol versions 1.3.x) link layer data packets (Ethernet datagrams specifically) are intercepted in OpenFlow-compliant routers and processed in ways that implement behaviours reflective of protocols or network behaviour definition imposed by externally applied SDN controller software. In other words, the operational semantics of the network layer or *the entire layer* can be bypassed. This goes beyond the ability in traditional networks to use different combinations of Internet standard and proposed standard protocols described in section 5.2.3. Version 1.5.x of the protocol, allows interception of any type of data packet, and has explicit support for running experimental or new protocols (Open Networking Foundation, 2009; Open Networking Foundation, 2012a; Open Networking Foundation, 2012b; Open Networking Foundation, 2015).

Recall the quote from Interviewee **I28**, Researcher in SDN, on layering ambivalence in SDN infrastructures (see page 178). The layers that make up the core architecture implemented in virtual networks in SDN infrastructures, relationships between layers, and the protocols in use, ***are instantiated at runtime***, and as per the preceding, can be any architecture required by the network operator (or the network operator's customer). The Infrastructure Virtualisation generative mechanism facilitates a shift

[53] At the time of this research, OpenFlow versions 1.0 to 1.3.x were the most widely supported in physical routers, with support for versions 1.5.x implemented more in software routers.

from core architecture constrained by a de-facto standard physically-oriented router implementation architecture, to ***runtime defined core architecture***[54].

Runtime-defined core architecture is rooted in infrastructure as executing software. This is ***ephemeral architecture***. Ephemerality indicates an intrinsic accommodation of a possibly short-lived architecture lifecycle, or architecture that is intrinsically accommodating of continued change – *architectural change introduced at runtime*.

Runtime definition of architecture has another important implication in the architectural evolution in sociotechnically ossified digital infrastructures. It facilitates *continued architectural evolution* in SDN infrastructures that are the outcome of the morphogenesis of traditional networks which has been explained to this point. This is, continued architectural evolution post structural elaboration.

Extant IS research theorises architecture in digital infrastructures from the perspective of there being a singular architecture whose generative capacity lends itself to exploitative innovation. In contrast, through runtime definition of architecture (implemented using virtualisation technologies) the SDN core architecture accommodates multiple heterogeneous, possibly incompatible, architectures, which may evolve independently of each other *within the confines* of their instantiating virtual networks. The SDN core architecture might be understood then as a *meta-architectural enclosure* for virtual architectures that are defined at runtime. In other words, SDN core architecture's generative capacity is in its ability to simultaneously support multiple heterogeneous, possibly incompatible, architectures.

This analysis reveals that the Infrastructure Virtualisation generative mechanism promotes both architectural evolution through which SDN infrastructures come into being, and continued architectural evolution in the resulting SDN infrastructures. In the context of architectural evolution, the Technical Capacity for Flexibility and

---

[54] Runtime definition of the deployment architecture of virtual networks is the trivial instance of runtime defined architecture.

Accommodating of Change and Technical Capacity for Introducing Capabilities to Generified Hardware via Software-Definition technical generative mechanisms, actualise to promote continued core architectural evolution *via runtime software-definition of architecture*. These technical generative mechanisms therefore are not peculiarly proceeding from SDN infrastructures. *They are rooted in runtime*.

Very importantly, though the SDN core architecture is ambivalent to the ephemeral architectures it encloses, continued architectural evolution is not an inevitability within network operators' infrastructures. Interviewee **I14**, Distinguished Engineer at a large network equipment vendor and systems integrator, cautioned that the extent to which any capability of SDN infrastructures is exploited remains dependent on network operators' business objectives and on their willingness to expose capabilities as network services to customers:

> *"So the network in and of itself doesn't change from SDN but it enables it to change. So it enables you to take self-provisioning to whole new levels. You decide what packages and combinations are allowed for your customers and then they can pick whatever they want, because the software will then drive it into the network. You get a new capability into your network and you can offer it. So SDN doesn't change what your network does by itself. It allows you to offer new capabilities." –* ***Interviewee I14***

Network operators and their customers, depending on the operator's customer network services, are left to decide how, if at all, to exploit this capacity for continued core architectural evolution. Nonetheless, SDN infrastructures retain from the initial products of academic research, the ability to define ephemeral architectures at runtime, and runtime definition of architecture is facilitative of continued core architectural evolution relative to the instantiating virtual infrastructure.

6.3.6.2 Runtime Revisited: Introducing Ephemeral Infrastructure

SDN controller management of virtual network lifecycles includes the activities of programmatic creation, modification, and removal of these networks, along with the assignment and reassignment of them (as relevant) to underlying physical hardware, at runtime. These are fully functional networking infrastructures that happen to be virtual. Physical networks cannot be instantiated, modified and removed at runtime as can virtual networks. These are ***ephemeral infrastructures***. They are infrastructures and not merely data being managed and processed by a SDN controller – recall that software routers make up virtual networks. Per the functional perspective of infrastructure, the SDN controller itself is infrastructure, and depending on technical decisions taken by network operators, has varying degrees of ephemerality.

Ephemerality indicates an intrinsic accommodation of a possibly short-lived infrastructure, amenable to change *at runtime*. Future research on how sociotechnical ossifications may re-appear as a consequence of growing dependence on ephemeral infrastructures is an interesting question, but it is outside of this research's focus. The point here is that ephemeral infrastructures instantiate ephemeral architectures, and therefore carry, inherently, the capacity for continued core architectural evolution – all within the meta-architectural enclosure of SDN infrastructures. In the context of architectural evolution, the Technical Capacity for Dynamic Tailoring of Infrastructure technical generative mechanism, actualises to promote continued core architectural evolution *in the runtime software-definition of infrastructure*. This technical generative mechanism therefore is not peculiarly proceeding from SDN infrastructures. *It is rooted in runtime*.

6.3.7 Summary of the Parallel Actualisation of Infrastructure Virtualisation: Introducing Computational Ontology

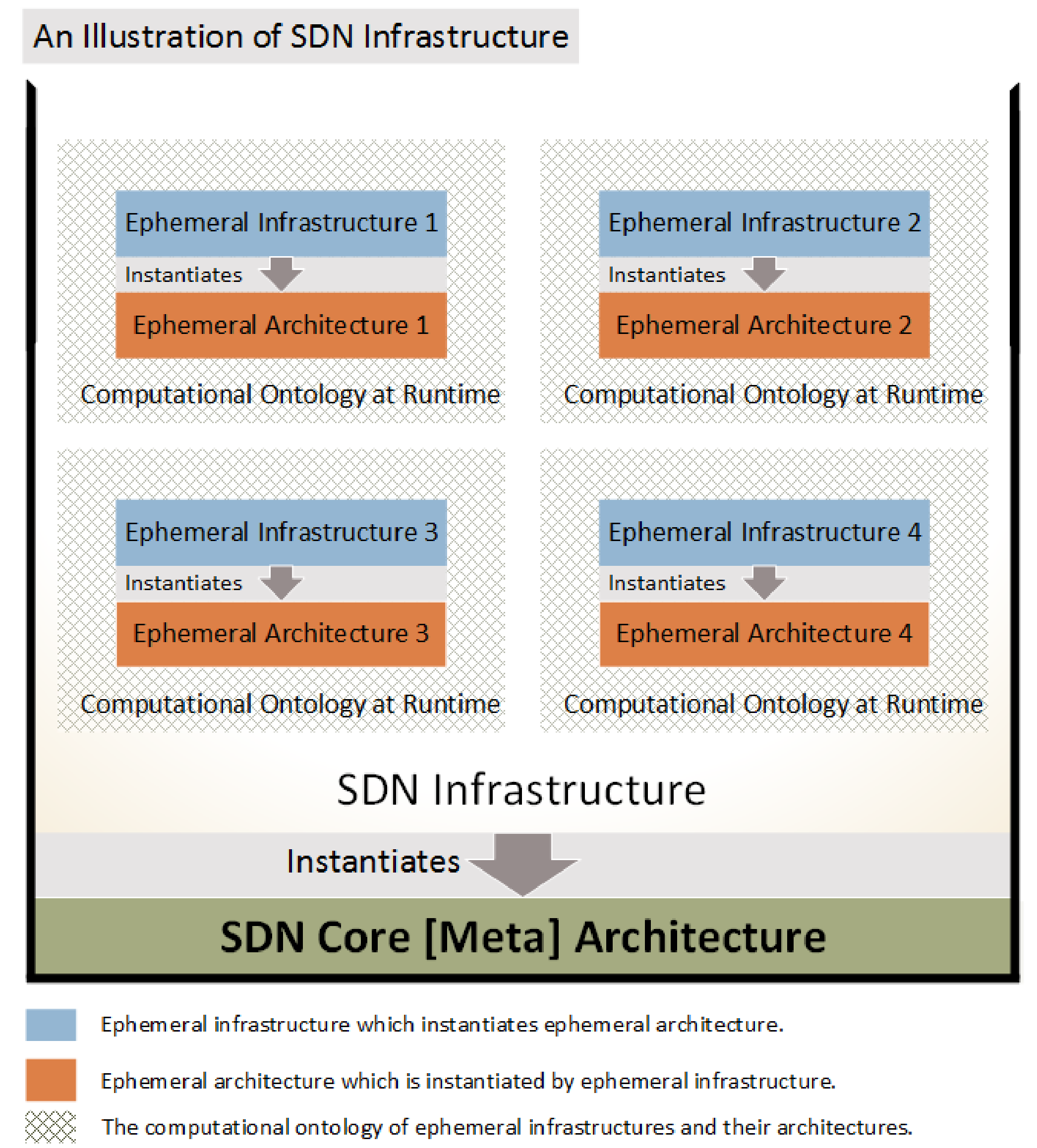

Figure 6-8 An Illustration of SDN Infrastructure

In this section I analysed the reframing of infrastructure, via a leveraging of its facilitating digital ontology that occurs in parallel with Refactoring the Digital Ontology of Infrastructure, and is the substance of the sociotechnical Infrastructure Virtualisation generative mechanism. The analysis explained that Infrastructure Virtualisation promotes both architectural evolution through which SDN infrastructures come into being, and continued architectural evolution within ephemeral virtual infrastructures instantiated within SDN infrastructures. Runtime, in the parallel actualisation of Refactoring the Digital Ontology of Infrastructure (and Technical Disaggregation, and Intelligence and Authority Migration from Hardware to Software), Infrastructure Abstraction, and Infrastructure Virtualisation generative mechanisms, underlies the Technical Capacity for Introducing Capabilities to Generified Hardware via Software-Definition technical generative mechanism.

Continued architectural evolution, is made possible by software-defined ephemeral architecture that exists solely as a runtime notion. The analysis also identified that the parallel actualisation of Infrastructure Virtualisation, yields a class of digital infrastructure that is ontologically a runtime-notion: ephemeral infrastructure. Any instance of ephemeral infrastructure instantiates ephemeral architecture, and intrinsically carries the capacity for continued core architectural evolution independently of the architecture instantiated by other ephemeral infrastructures that may co-exist within the enclosure of a network operator's SDN infrastructure's core meta-architecture.

The digital ontology of this class of digital infrastructure, ephemeral infrastructure, is a computational one. The architecture of this class of digital infrastructure is a computational one. The make-up of ephemeral infrastructure, including its architecture, is computed at runtime. It is a ***computational ontology***. Computational ontology can be understood as a [recursive][55] subset of the overall digital ontology of SDN infrastructures. Virtual networks, the ephemeral infrastructures, have form, function, they can be traced to physical manifestation, they feature within them residual constraints of physicality, and are infrastructure as executing software, but this digital ontology of the virtual network is itself, computationally produced at runtime. Virtual networks are programmatically created; the actions of software

[55] This recursion is a consequence of the intrinsic reflexivity of digital ontology, wherein digital objects mediate other digital objects (Kallinikos, et al., 2010; Kallinikos, et al., 2013). Recursion is demonstrable here, but further elaboration goes beyond the answer to the research question.

routers and logical abstractions of physical resources involved in the network are computationally produced at runtime. Figure 6-8 visually illustrates how computational ontology at runtime, ephemeral architecture and ephemeral infrastructure, and the notion of a core meta-architecture, feature in SDN infrastructures.

For clarity, note that computational ontology at runtime is unrelated to and incompatible with sociomateriality's conception of materiality as a process of materialisation (Scott & Orlikowski, 2014). Computational ontology at runtime is not a process or an enactment. It is not in a process of becoming. It is the outcome of prior computations of software, whose operations cannot always be entirely simplified to straightforward explanations in terms of receiving input (originating in the social) and producing corresponding output (in the technical). The scope of technical details is well beyond what is relevant to answering the research question, but for completeness I highlight, here, a remark by Interviewee **I14**, Distinguished Engineer at a large network equipment vendor and systems integrator, on autonomous machine learning:

> *"**[Anonymised Detail]** also has hoards of machine learning. It's just what they do.*
>
> *Because they had a programmable network they could program machine learning to their network traffic engineering and have it – and program it with – and show it patterns that they had used and the effects that they had gotten and then have it adjust the network and observe what happens.*
>
> *They have demonstrated that they are getting better traffic engineering and better network utilisation than could have been achieved by automation - by manual or conventional policy based tools." –* ***Interviewee I14***

What does computational ontology help to explain? Computational ontology manifested at runtime underlies the Technical Capacity for Flexibility and Accommodating of Change technical generative mechanism that distinguishes SDN infrastructures from traditional networking infrastructures. Computational ontology at runtime accommodates a scope of networking infrastructure runtime behavioural variety and redefinition via software-definition of architecture, and of network capabilities (i.e., the Technical Capacity for Introducing Capabilities to Generified Hardware via Software-Definition technical generative mechanism) that is broader in degree than that accommodated in the pre-delimited scope of flexibility tightly coupled to physical hardware typical of traditional networks. Discussion of some consequences of computational ontology at runtime in the morphogenesis of the networking industry follows the elaboration of sociotechnical generative mechanisms.

6.3.8 Analysing Infrastructure Abstraction

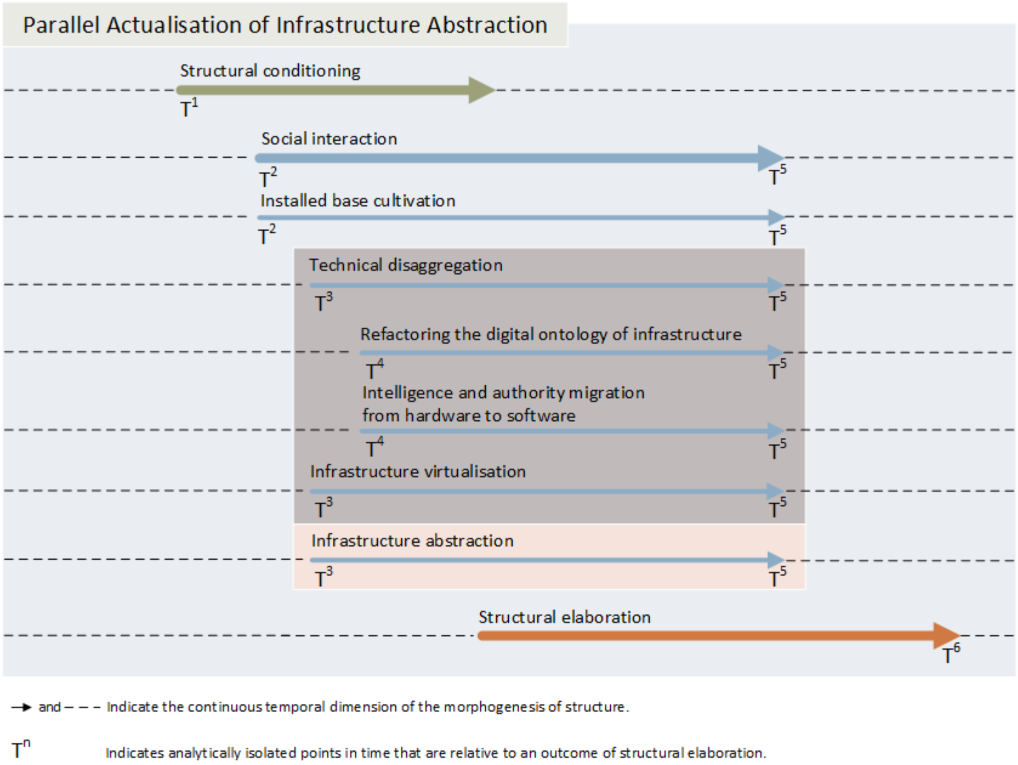

Figure 6-9 Parallel Actualisation of Infrastructure Abstraction with Other Generative Mechanisms

The Infrastructure Abstraction and Infrastructure Virtualisation generative mechanisms remain closely related and interact to bring about ephemeral architecture. The relationship with Infrastructure Virtualisation also contributes towards providing a more complete explanation of the technical generative mechanisms that emerge with SDN infrastructures. Logical homogenisation as realised by the operation of the Infrastructure Abstraction generative mechanism, brings into compliance heterogeneous infrastructural entities with the capacity to realise ephemeral architecture uniformly across the infrastructure - though the ephemerality is rooted in runtime. Thus, the operation of Infrastructure Abstraction causally features in the emergence of the Technical Capacity for Introducing Capabilities to Generified Hardware via Software-Definition technical generative mechanism.

Exposing the network as a generic resource through standardised open APIs, permits client applications to dynamically define and configure network behaviour of shared networking infrastructure or virtual networks at runtime that are application-specific, and possibly short-lived. In other words, Infrastructure Abstraction underlies the Technical Capacity for Dynamic Tailoring of Infrastructure technical generative mechanism, but instantiation of tailored ephemeral infrastructures is attributed to the shift from the physical to logical and virtual – i.e., Infrastructure Virtualisation.

Analysis of the sociotechnical Infrastructure Abstraction generative mechanism did not reveal anything significant to the architectural transformation of traditional networking infrastructures that could be causally attributed to concurrent interactions with other mechanisms (Figure 6-9).

6.4 Summary of the Sociotechnical Generative Mechanisms

6.4.1 Review of the Analysis

The analysis so far, has focused on the sociotechnical generative mechanisms that are closely associated with the morphogenesis of traditional networks. Two perspectives were discussed. The first identified and explained generative mechanisms that promote architectural evolution in sociotechnically ossified digital infrastructures (traditional networking infrastructures) by which SDN infrastructures come into being. The second explained generative mechanisms that promote continued architectural evolution in SDN infrastructures.

Four sociotechnical generative mechanisms, operating alongside the social generative mechanism of Installed Base Cultivation, that architecturally transformed traditional networks into SDN infrastructures were identified:

1. **A Process of Technical Disaggregation:** operates to identify, and to decouple key ossified intelligence from whatever tightly couples it.
2. **A Process of Intelligence and Authority Migration from Hardware to Software:** operates to relocate decoupled intelligence from its hardware confinement to externally applied software.
3. **A Process of Infrastructure Virtualisation:** operates to reframe infrastructure, via a leveraging of its facilitating digital ontology, in terms of the virtual.
4. **A Process of Infrastructure Abstraction:** operates to identify, and to logically consolidate and homogenise key sources of infrastructure heterogeneity.

Three emergent technical generative mechanisms that characterise SDN infrastructures as qualitatively distinct from traditional networks and thus arising from sociotechnical generative mechanisms of architectural evolution were identified:

1. **Technical Capacity for Dynamic Tailoring of Infrastructure**
2. **Technical Capacity for Flexibility and Accommodating of Change**
3. **Technical Capacity for Introducing Capabilities to Generified Hardware via Software-Definition**

These technical generative mechanisms, act transfactually, but may actualise as part of continued architectural evolution in SDN infrastructures[56]. The explication of the sociotechnical generative mechanisms confirmed that their actualisation additionally promotes continued architectural evolution in SDN infrastructures. The analysis also confirmed causal attribution to sociotechnical generative mechanisms, and their interaction for the emergent characteristics that are distinct to SDN infrastructures.

---

[56] Indeed, an argument can be made that a perspectival switch (Hartwig, 2007, pp. 344-345) may render these enabling conditions for later sociotechnical generative mechanisms that produce new architectures. That however, lies outside of this case's boundary and is left to future research.

Through the analysis of the interactions between the Technical Disaggregation, and Intelligence and Authority Migration from Hardware to Software generative mechanisms, an additional emergent mechanism was identified:

A Process of Refactoring the Digital Ontology of Infrastructure:

1. Operates to modify the matter and form that is instrumental to the execution of function.
2. Operates to transition the ontology of infrastructure towards executing software and to include runtime.

There may be additional generative mechanisms, but these seem to sufficiently cover both accounts of architectural evolution. The detailed explication of the generative mechanisms, in a search for alternative explanations (Sayer, 2000, pp. 13-17; Easton, 2010; Wynn & Williams, 2012), has established, *as ontological*, the reality of the sociotechnical generative mechanisms in the domain of the real, and that they are sufficiently explanatory of architectural evolution in sociotechnically ossified infrastructures. A final social generative mechanism, however, completes the explanation. I discuss it following this summary.

The explanation of the sociotechnical generative mechanisms produced some important proposed corollary theoretical insights that are left to the Discussion chapter.

6.4.2 Introducing Softwarisation

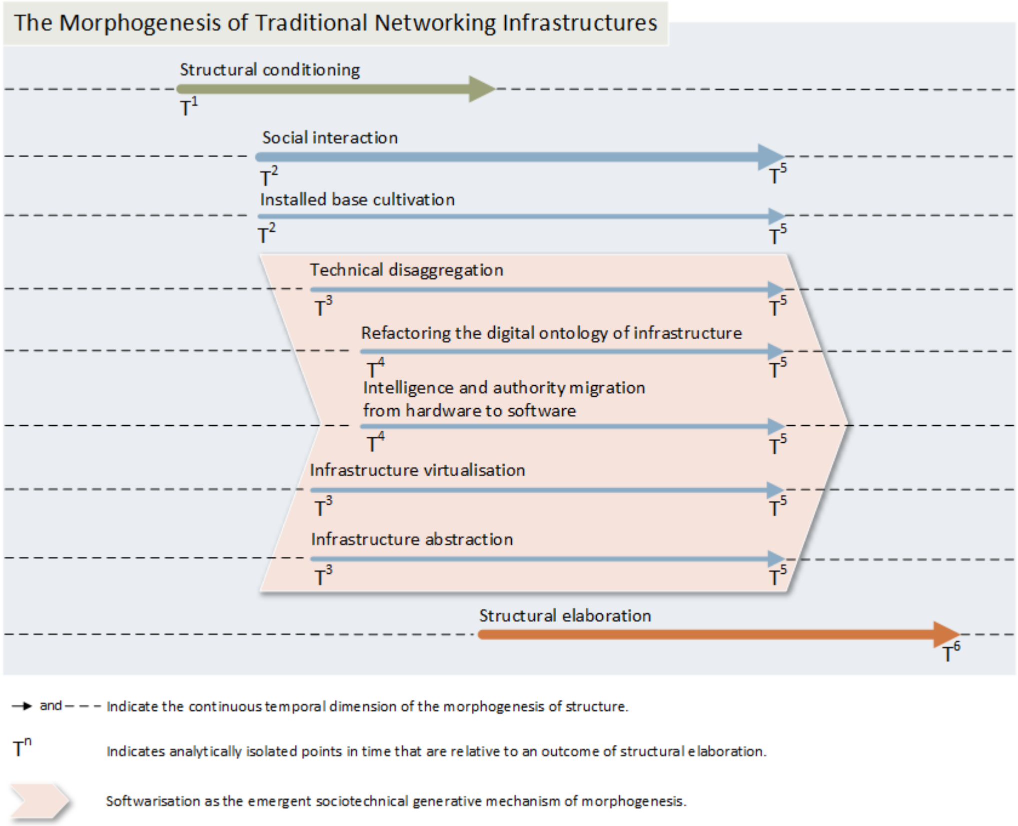

Figure 6-10 Softwarisation as the Sociotechnical Generative Mechanism of the Morphogenesis of Traditional Networking Infrastructures

The sociotechnical generative mechanisms that promote architectural evolution through which SDN infrastructures came about, can be understood as a process of *Softwarisation*. The term "softwarisation" is an industry buzzword typically used to connote the movement towards software-based infrastructures, as with cloud computing and SDN infrastructures. I have borrowed the term, and assigned it this formal elaboration in terms of critical realist sociotechnical generative mechanisms. Softwarisation is *the* sociotechnical generative mechanism of morphogenesis (Archer, 1995, p. 217) that is emergent out of necessary synchronic relations (Elder-Vass, 2007) of the Technical Disaggregation, Intelligence and Authority Migration from Hardware to Software, Refactoring the Digital Ontology of Infrastructure, Infrastructure Virtualisation, and Infrastructure Abstraction sociotechnical generative mechanisms that operate in the diachronic sequential and concurrent temporality explained in this chapter and summarised in diagram Figure 6-10.

Characteristic of the necessary synchronic relations is that all sociotechnical generative mechanisms are arranged around and thus operate with respect to the same identified point of sociotechnical ossification within digital infrastructures that is to be architecturally evolved. That is, the generative mechanism of morphogenesis (Archer, 1995, p. 217) which transforms digital infrastructures emerges only on the basis of these synchronic relations (Elder-Vass, 2007). The relationship between Softwarisation and the morphogenesis of the networking industry is discussed next, and I additionally discuss Softwarisation in Chapter 7.

6.5 On the Morphogenesis of the Networking Industry

6.5.1 Introduction

Having identified and analysed generative mechanisms of the morphogenesis of traditional networks, the analysis now proceeds to their relationship with the morphogenesis of the networking industry.

To reiterate, the objectives of the morphogenesis of the networking industry have been to subvert points of architectural control (Woodard, 2008) held by vendors that are manifested within an ossified de facto standard router implementation architecture, to create conditions for opening the networking industry to the entrance of new innovators, and for cost-effective replacement and interchangeability of vendors' devices. Because network intelligence is what is considered valuable in infrastructures, the tight coupling and encapsulation of the forwarding and control planes within hardware routers has served as an architectural point of control in network operators' networking infrastructures for vendors.

6.5.2 Softwarisation Revisited: Softwarisation for Social Disaggregation

Taking the value perspective of infrastructure, Softwarisation can be additionally clarified. The objectives of the morphogenesis of the networking industry are instrumented through Softwarisation *with respect to* a sociotechnically ossified architectural point of control *to which value has been assigned.* (This is a refinement to the necessary synchronic relations of the sociotechnical generative mechanisms (Elder-Vass, 2007) from which Softwarisation emerges.) Softwarisation is a sociotechnical generative mechanism by which these points of control are subverted. Softwarisation of sociotechnically ossified architectural control points in networking infrastructures corresponds with the broader social objective of *social disaggregation* of ossified necessary internal relations of interdependence between network operators, vendors and systems integrations on the provision, use, and implementation of networking products and services that sustain financially unfavourable conditions for network operators. Similarly to the strategy described in (Katz & Shapiro, 1994; Shapiro & Varian, 1999, pp. 196-203), in the Softwarisation generative mechanism of morphogenesis, socially defined and maintained openness (via open standardisation processes, standardised open APIs, and community-curated open source code) aims to sustain the social disaggregation of these internal relations once disaggregated.

Social Mechanism: A Process of Social Disaggregation

Social disaggregation breaks up ossified social relationships between network equipment vendors, systems integrators and network operators to create conditions for an open vendor-neutral networking industry that admits the entrance of new innovators, and facilitates cost-effective replacement and interchangeability of network equipment vendors' hardware devices.

Importantly, I have not generified the description of the Social Disaggregation generative mechanism to remove references to the networking industry because at the conclusion of this research it was only plausibly identified as a social generative mechanism. I elaborate on this in the following section.

6.5.3 Structural Conditioning and the Morphogenetic Cycle of the Networking Industry: An Explication of the Social Disaggregation Generative Mechanism

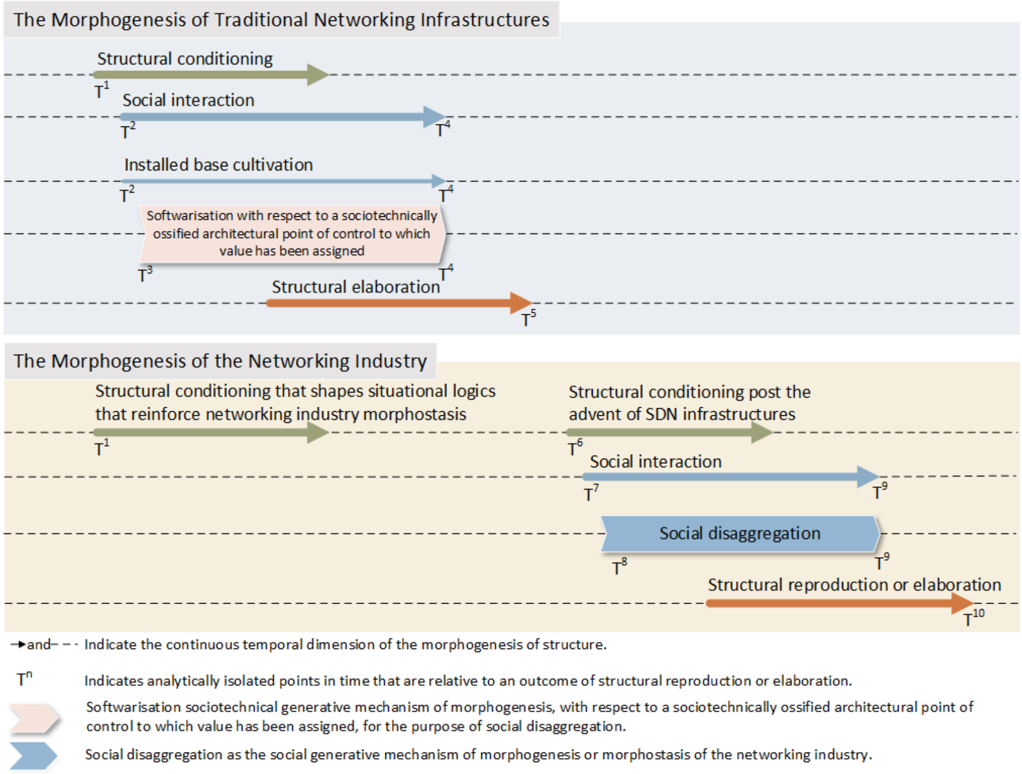

Figure 6-11 Relationship Between the Morphogenesis of Traditional Networking Infrastructures and the Morphogenesis of the Networking Industry

Recall from Chapter 3 that social structure that is the outcome of prior morphostasis or morphogenesis, subsequently conditions later human actions that either reproduce or transform it (Archer, 1982; Archer, 1995, pp. 149-161, 165-194; Archer, 2000).

With there being only a very small percentage of all networks that are SDN infrastructures, given its technological recency, the implications of Softwarisation with respect to the sociotechnically ossified architectural point of control, to which value has been assigned, in networking infrastructures for the purpose of Social Disaggregation, are not yet fully known. More than this, the morphogenetic cycle of the networking industry may yet lead to morphostasis. Therefore, although through analysis I have identified Social Disaggregation as a plausible social generative mechanism, its detailed explication and how it broadly relates to architectural evolution in sociotechnically ossified digital infrastructures remains limited. As such, in Figure 6-11, the analytically isolated temporal point in time at which the Social

Disaggregation generative mechanism begins to operate is indicated at T^8 and not simultaneously at T^7 with the social interaction phase, to accommodate the possibility of there being social interaction that is prior to its operation (similar to Install Base Cultivation's anteriority at T^2 to Softwarisation which begins to operate at T^3 in the morphogenesis of networking infrastructures). Notwithstanding, there have been reactions by incumbent network equipment vendors, systems integrators, and network operators to early structural conditioning of SDN infrastructures post their advent (i.e., onwards from the analytically isolated temporal point in time T^6, in Figure 6-11), that serve as implicit admissions on their part of there being morphogenetic implications for the networking industry of Softwarisation with respect to a sociotechnically ossified architectural point of control to which value has been assigned.

Analysis of interviewee responses produced a characterisation of Softwarisation for the purpose of Social Disaggregation, as an opportunity from the perspective of network operators, and as both a threat and an opportunity from the perspective of networking equipment vendors.

From the perspective of network operators, Softwarisation for the purpose of Social Disaggregation has been an opportunity to transform traditional networks to accomplish the objectives of the morphogenesis of the networking industry, while addressing issues of increasing network complexity, the desire for flexible infrastructure, and the desire to lower Capex and Opex. Some examples of this since the beginning of this research include, NTT Communications' Softwarisation of its core network and its commercial Enterprise Cloud offering to enterprises (International Data Corporation, 2014; NTT Communications, 2015; NTT Communications, n.d.; Gartner Inc., 2015b; Current Analysis Inc., 2016), AT&T's Domain 2.0 initiative (AT&T, 2013) and its Network on Demand customer offerings of software-defined network capabilities such as Managed Internet Service on Demand, Ethernet on Demand, and

FlexWare (formerly Network Functions on Demand)[57] (AT&T, 2014; AT&T, 2015a; AT&T, 2015b; AT&T, 2015c; AT&T, 2015d; AT&T, 2015e; AT&T, 2015f; AT&T, n.d.-b; AT&T, 2016a; AT&T, 2016b), Google's B4 WAN (Jain, et al., 2013), Verizon's SDN infrastructure (Verizon, 2016) and customer offering of Virtual Network Services[58] (Verizon, n.d.; Verizon, 2016), and revelations that Microsoft's cloud offering, Azure, is a SDN infrastructure (Russinovich, 2015; Greenberg, Albert, 2015; Subramaniam, 2016).

Standardisation and logical homogenisation of hardware threatens commoditisation of network hardware devices, undercutting the need for specialised networking equipment sold by vendors, and challenging vendors' existing business models. Interviewee **I28**, Researcher in SDN, remarked:

> *"**So it's every single equipment vendor, be it Cisco, Alcatel, Ericsson, Huawei, OK? Everyone of those is affected**. Depending on their degree of incumbency and dependence on legacy networks, that will determine the degree of impact - but all of them get - **because the business model is changing from one where they sold very expensive boxes to one in which they have to survive selling software**. So that's a complete transformation of the business." – **Interviewee I28**, (Bold and underline emphasis added.)*

Expanding on the threat to vendors when network operators' actions use the implementation of SDN infrastructures to favour commoditised hardware, Interviewee **I22**, Senior Director at a network equipment vendor and systems integrator, stated:

---

[57] AT&T's FlexWare is a realisation of Network Functions Virtualisation (NFV), but AT&T's implementation is built on SDN infrastructure.

[58] Likewise to AT&T's FlexWare, Verizon's Virtual Network Services is a realisation of Network Function's Virtualisation (NFV), but it is built on SDN infrastructure.

> *"But the point I am making is that, if you fall victim to that, it's a race to the bottom.* ***No one's going to survive when the operator drives us into the ground.***
>
> *Or, if you look at what Google and Amazon and the giants are doing, they just bypass the vendor community altogether, and design their own hardware, because they can produce it at scale, get the same ODMs to squeeze out the margin that an OEM, that an operator - excuse me, that a network equipment manufacturer might get and there that is the tactic they are doing."* – ***Interviewee I22****, (Bold and underline emphasis added.)*

Interviewee **I11**, Vice President in Networking at a Tier 1 ISP, however, raised a notable point in this relatively long quote, to which I return later, that the social objectives of the morphogenesis of the networking industry may threaten both vendors and network operators:

> *"So, how do I know that the organisation that I choose to invest in today and provide equipment or software services and all of that sort of thing, is A) going to be able to keep up with everything that's going on in the industry over the next five or six years or in fact going to be able to exist if they haven't actually thought about their future and how they fit into this whole environment? So that is what worries me.*
>
> *Cloud service providers don't worry about that. Your switch looks the same as everybody else's switch. Your hardware I'd change every – from a server point of view – I'd change every nine months anyway if I am Google, and every twelve months if I am Facebook then every eighteen months if I am Azure.*
>
> ***'So I actually do not care whether you are around today or tomorrow,' and that is a very - that's also a very concerning attitude, because eventually, if you keep on burning through organisations and driving them into the ground and destroying vendors and so on, and consolidation can sort of absorb that for a bit, but eventually you will end up with zero choice or everybody broke, and none of those are***

> ***desirable situations**." – **Interviewee I11**, (Bold and underline emphasis added.)*

Responses to the perceived threat of Softwarisation for the purpose of Social Disaggregation among incumbent vendors has been varied, and has included, strategic acquisitions, creation of technical alternatives, and even dissemination of confusing information. For instance, though a member of the ONF, Cisco, a major vendor and long-term leader in networking solutions (Gartner, 2016), following its acquisition of Insieme Networks (Cisco Systems, 2013a; Cisco Systems, 2013b), offered its Application Centric Infrastructure (ACI) product as a technical solution to major business concerns addressed by SDN as an innovation, along with OpFlex, which it positioned as an alternative southbound interface to the ONF's OpenFlow protocol (Cisco Systems, 2013c; Cisco Systems, 2014c; Cisco Systems, 2014d). The offering however, did not decouple the forwarding and control planes or centralise the control plane into an SDN controller, and for that matter addressed a different class of problem, namely policy definition in networks, which is not perfectly synonymous with the ability to introduce networking capabilities to networking hardware via software-definition – but at the time it promoted a lack of clarity about what SDN as an innovation was. In another incident, involving the same vendor, Cisco purchased Tail-F, a supplier of network hardware device management and controller software to AT&T's Domain 2.0 initiative (AT&T, 2013), after not being initially named as a supplier (AT&T, 2014) – ensuring a continued influential presence within the network operator's SDN infrastructure (Cisco Systems, 2014a; Cisco Systems, 2014b).

On the other hand, Softwarisation for the purpose of Social Disaggregation has been treated as an opportunity by some vendors open to repositioning themselves commercially around SDN as an innovation, and to pursuing new types of networking innovation not possible prior to the advent of SDN infrastructures. Interviewee **I32**, Researcher in SDN, explained:

> *"Of course, this is creating some problems to the device manufacturers, because basically this is allowing other players to enter into the market. So, of course, the classical manufacturers of networking devices are challenged by this novelty. However, of course, **they are already moving***

***in the direction of repositioning in order to be able to provide value, not now, of course, not only in the physical devices, but also on the eco-system of solutions that they can provide to their customers**." – **Interviewee I32**, (Bold and underline emphasis added.)*

Examples of this since the beginning of this research include, Juniper who since its acquisition of Contrail Systems (Juniper Networks Inc., 2012) has developed a portfolio of SDN products that include data centre and WAN SDN controllers deployed by large network operators (such as AT&T (Juniper Networks Inc., 2015)), Arista that has created a portfolio of SDN software and SDN compatible hardware products[59] (Arista Networks Inc., n.d.-a), Radysis whose SDN products have been incorporated into Verizon's SDN infrastructure (Radisys Corporation, 2016), and Brocade who appropriated the open source OpenDaylight SDN controller (OpenDaylight Foundation, n.d.; Brocade Communications Systems Inc., n.d.), and offers a portfolio of SDN software and SDN compatible hardware products.

The degree to which the social objectives of the morphogenesis of the networking industry occurs remains to be seen. However, the preceding analysis of preliminary findings do seem to corroborate that vendors, system integrators, and network operators, recognise Social Disaggregation of ossified social arrangements as catalysed by Softwarisation.

I contend on the basis of what the analysis revealed, that the actions of vendors betray their recognition that the morphogenesis of traditional networks through Softwarisation to yield computational ontology at runtime makes it difficult to re-ossify routers' implementation architecture as a hardware architectural point of control in a manner that preserves networking industry social relations that are favourable to them. On the other side, SDOs and network operators framed the advent of SDN

[59] Arista's market claims to have been founded "to pioneer and deliver software-driven cloud networking solutions" (Arista, Networks Inc., n.d.-b), should not be misunderstood as describing SDN solutions. SDN as an innovation post-dates Arista's founding.

infrastructures as part of a sociotechnical strategy for creating a vendor-neutral open networking industry, recognising that Softwarisation and computational ontology at runtime creates a challenge to vendor-preferential re-ossification. Concomitant with Softwarisation is the migration of value to software, and a converse devaluation of hardware (though not for all hardware). If Social Disaggregation of ossified social relations in the networking industry that leads to structural elaboration was to ensue, vendors would be forced to follow the value that is migrated from hardware to software, and to accordingly adjust their business models around software and to compete with independent software vendors (recall the quote by Interviewee **I28** on page 242). Drawing again on Cisco as an example, in August 2016, the vendor announced a restructuring (costing 5,500 employees their jobs) that aims to transition the organisation from dependence on its network hardware device sales towards software-based products and services in response to corresponding changes in the networking industry – including those changes created by the actions of service provider network operators (Cisco Systems, 2016; Reuters, 2016).

The significance of the concern to vendors can be understood when considering that I have largely focused on what happens to the control function. But recall that the forwarding function, implemented in hardware, can be implemented in software routers in highly virtualised environments such as in data centre networking – and that it has been this way since the advent of the first commercial SDN products. The choice between making commoditised hardware (if still economically favourable) and emphasizing software development and services, or having an installed base of hardware from which sufficient economic value can no longer be derived due to repurposing through software-definition, by network operators, for use differently from how the network equipment vendor intends, has not been an easy one for vendors to face.

In spite of the recognition of the implications of Softwarisation for Social Disaggregation, and recognition of the role of computational ontology at runtime, Social Disaggregation is not a straightforward conclusion either. Recall that experience and infrastructure investment have been entry barriers for new entrants to the networking industry. Softwarisation may just be used as a threat as per the quote from

Interviewee **I34** on page 202 – with less intention of making the types of industry changes that could in time harm network operators, as per the quote from Interviewee **I11** on page 243. In the transition towards SDN infrastructures, some network operators have also taken actions that preserve and reproduce the existing social structure of the networking industry. Using AT&T and Verizon as exemplars, these network operators have through Softwarisation transformed their networking infrastructures, but offer their business customers ephemeral infrastructures, such as virtual software routers, that are virtualisations of leading incumbent vendors' routers (AT&T, 2015a; AT&T, 2015b; AT&T, 2015d; AT&T, n.d.-b; AT&T, 2016b; Verizon, n.d.; Verizon, 2016).

Notable as well, following the circumvention, through Softwarisation, of the architectural point of control instrumented in the de facto standard implementation architecture of routers, has been what appears to be instrumentation by some network operators of an architectural point of control to be held by them. Again drawing on the actions of AT&T and Verizon, both have introduced a hardware-based customer premises device, that through computational ontology at runtime assumes an ephemeral infrastructural identity (be it a virtual software router, virtual firewall or other network component)[60] that is software-defined (AT&T, n.d.-b; Verizon, n.d.). Whether this leads to a new type of ossification between customers and their network operators remains to be seen.

Sophisticated explication of Social Disaggregation as a social generative mechanism of either morphogenesis or morphostasis requires dedicated research that extends beyond the scope of this research's question. Nonetheless, the contribution of this research is its identification as a social generative mechanism that features in architectural evolution in sociotechnically ossified digital infrastructures. Additionally, what can be stated with certainty is that Softwarisation, runtime, and computational

[60] See footnotes 57 and 58.

ontology at runtime feature consequentially in complex and significant social issues that transcend the purely technical.

6.5.4 Summary of the Morphogenesis of the Networking Industry

This section of the analysis identified Social Disaggregation as a plausible social generative mechanism of morphogenesis or morphostasis. Social Disaggregation is catalysed by Softwarisation with respect to a sociotechnically ossified architectural point of control to which value has been assigned, but it is a complex social mechanism (Avgerou, 2013) that requires further elaboration in future research. Softwarisation, runtime, and computational ontology at runtime pervade the complexity of Social Disaggregation.

6.6 A Word on Technological Determinism and Voluntarism

Note that although the findings reveal that network operators and SDOs have been using technical methods to facilitate network industry change, the strategy cannot be dismissed as being technologically determinist. Network operators and SDOs understand that technology is necessary but not sufficient to effect the desired social change (see also (Lyytinen & Rose, 2003)). They also recognise the necessity of the social generative mechanism of Installed Base Cultivation. Neither can the appropriation of the technological outputs of academic research by network operators and SDOs be construed in any way as being voluntarism. Both classifications are extremes. Counter to these classifications, the findings show that network operators, SDOs – and vendors – have a more balanced conceptualisation of Softwarisation for the purpose of Social Disaggregation. Additionally, in accordance with critical realism's formulation of causality in open environments, the findings reveal that morphogenesis of the networking industry is not an inevitability. (See also (Leonardi & Barley, 2008; Wyatt, 2008) who argue for balance regarding issues of technological determinism and voluntarism.)

6.7 Summary of Analysis

This chapter presented analytic arguments for the reality of the generative mechanisms sought by the research question. The operation of a Softwarisation generative mechanism of morphogenesis which emerges out of the synchronic arrangement of five sociotechnical generative mechanisms, was shown to be what causally intervenes between antecedent structural conditioning of traditional networks and the advent of SDN infrastructures. Underpinning the morphogenetic sociotechnical generative mechanisms, is a social generative mechanism, Installed Base Cultivation. Three technical generative mechanisms that are rooted in runtime that emerge with SDN infrastructures were identified as promoting continued core architectural evolution post structural elaboration. Finally, the relationship between the advent of SDN infrastructures, and the early stages of the operation of a Social Disaggregation social generative mechanism of either morphogenesis or morphostasis in the morphogenetic cycle of the networking industry, was presented.

7 Discussion

7.1 Proposed Theoretical Contributions

7.1.1 Answer to the Research Question: Architectural Evolution through Softwarisation

The research question asked:

Which mechanisms promote architectural evolution in sociotechnically ossified digital infrastructures?

The answer developed through analysis, is that Softwarisation, a sociotechnical generative mechanism of morphogenesis that is emergent out of the synchronic relations (Elder-Vass, 2007) of five sociotechnical generative mechanisms, promotes architectural evolution in ossified digital infrastructures. Post-morphogenesis, three technical generative mechanisms that are rooted in runtime, namely, a Technical Capacity for Flexibility and Accommodating of Change, a Technical Capacity for Introducing Capabilities to Generified Hardware via Software-Definition, and a Technical Capacity for Dynamic Tailoring of Infrastructure, may emerge to promote continued core architectural evolution in structurally elaborated digital infrastructures.

Softwarisation is always with respect to some identified point of sociotechnical ossification within digital infrastructures. Softwarisation leverages the digital ontology of sociotechnically ossified digital infrastructures (such as traditional networking infrastructures), exploiting in particular the dimension of runtime, to yield digital infrastructures (such as SDN infrastructures), in which runtime underlies the definition of architecture and of continued architectural evolution.

In the case of the advent of SDN infrastructures, in the morphogenesis of traditional networks, the Softwarisation generative mechanism leveraged the digital ontology of (bearers in) traditional networks, particularly exploiting the dimension of runtime to circumvent sociotechnical ossifications that resist architectural evolution. The runtime dimension of digital ontology was exploited via firmware updates[61] to subvert ossifications rooted architecturally in routers to make them defer at runtime to SDN controllers. Runtime is exploited through SDN controllers that create virtual ephemeral infrastructures that instantiate ephemeral architectures which being runtime notions can be updated at runtime to admit continued architectural evolution within the confines of their instantiating ephemeral virtual networks.

Formally, architectural evolution through Softwarisation involves a Process of Technical Disaggregation which operates to identify, and to decouple key ossified intelligence from whatever tightly couples it, and cumulatively with a Process of Intelligence and Authority Migration from Hardware to Software, relocates the decoupled intelligence from its hardware confinement to externally applied software, modifying – through a Process of Refactoring the Digital Ontology of Infrastructure – the matter and form that is instrumental to the execution of function, to transition the ontology of infrastructure towards executing software at runtime. A Process of Infrastructure Virtualisation, operating in parallel reframes infrastructure, via a leveraging of its facilitating digital ontology, in terms of the virtual, to transform architecture into an ephemeral runtime notion instantiated by ephemeral infrastructures amenable to continued core architectural evolution. A Process of Infrastructure Abstraction operates to identify, and to logically consolidate and homogenise key sources of infrastructure heterogeneity which obstruct social motivators for infrastructure architectural evolution. Softwarisation with respect to sociotechnically ossified architectural points of control *to which value is assigned* may catalyse Social Disaggregation, a social generative mechanism that resists regression to the initial state of sociotechnical ossification of a digital infrastructure.

[61] See section 5.4.8.

7.1.2 Generative Mechanisms that Promote Architectural Evolution in Ossified Digital Infrastructures

The analysis identified six morphogenetic sociotechnical generative mechanisms, three technical generative mechanisms that act transfactually but may actualise to promote continued architectural evolution in ossified digital infrastructures, and two social generative mechanisms - one of which underpins the morphogenetic sociotechnical generative mechanisms (i.e., Installed Base Cultivation).

The Six Sociotechnical Generative Mechanisms:

1. **A Process of Softwarisation** as emergent out of the synchronic relations of:
    1. **A Process of Technical Disaggregation**
    2. **A Process of Intelligence and Authority Migration from Hardware to Software**
    3. **A Process of Refactoring the Digital Ontology of Infrastructure**
    4. **A Process of Infrastructure Virtualisation**
    5. **A Process of Infrastructure Abstraction**

The Three Technical Generative Mechanisms that promote continued architectural evolution:

1. **Technical Capacity for Dynamic Tailoring of Infrastructure**
2. **Technical Capacity for Flexibility and Accommodating of Change**
3. **Technical Capacity for Introducing Capabilities to Generified Hardware via Software-Definition**

The Two Social Mechanisms, of which, Installed Base Cultivation underpins the Softwarisation sociotechnical generative mechanism of morphogenesis:

1. **A Process of Installed Base Cultivation**
2. **A Process of Social Disaggregation**

7.1.3 Corollary Theoretical Contributions

The explication of the sociotechnical generative mechanisms produced some important proposed corollary theoretical insights. Re-articulation of digital infrastructures as executing software, led to an extension of digital ontology in the context of software to include runtime. Attention to runtime, led to identification of a class of architecture that is ontologically a purely runtime-notion: ephemeral architecture. This ephemeral architecture facilitates the architectural ambivalence that underlies continued core architectural evolution within the confines of virtual ephemeral infrastructures as seen in SDN infrastructures. The analysis identified a class of digital infrastructures that instantiates ephemeral architecture, and is ontologically a purely runtime-notion: ephemeral infrastructure. Continued architectural evolution in ephemeral infrastructures is made possible by software-definition of their ephemeral architectures.

Another insight has been the identification and articulation of a subset of digital ontology that is computed at runtime, namely computational ontology. Computational ontology at runtime underlies, ephemeral architecture, ephemeral infrastructure, and the Technical Capacity for Flexibility and Accommodating of Change technical generative mechanism that, relative to traditional networking infrastructures, is distinctive of SDN infrastructures. Ephemeral infrastructures inherently accommodate continuous ontological re-constitution, including core architectural evolution, and more strongly they exist constituted of computational ontology primarily for this purpose. SDN infrastructures and the purpose of their advent do not fulfil their technical or social objectives in the absence of computational ontology.

7.2 Evaluating the Theoretical Contributions

7.2.1 Introduction

Single case studies generalise via theoretical contribution, and therefore require an analytical generalisation argument (Yin, 2014, pp. 20-21; 40-41). This is the generalisation argument that I seek to establish in the subsequent sections of this chapter – specifically that the research results generalise satisfactorily to theory. Additionally, I argue that the key attributes of explanatory theory (Markus & Robey, 1988; Gregor, 2006) and of theoretical contributions are present in this thesis' proposed theoretical contributions.

7.2.2 How have the Requirements of Explanatory Theory been met?

The research objective was to uncover how architectural evolution in sociotechnically ossified digital infrastructures occurs. The analysis identified the Softwarisation sociotechnical generative mechanism of morphogenesis that along with three technical generative mechanisms promote two types of architectural evolution: a transformative architectural evolution, and an architectural evolution that is a continuously accommodated capacity of an inherently ambivalent architecture. It identified a social generative mechanism, Social Disaggregation, to which causality for modifying ossified social relations in the networking industry is contingently attributed.

According to (Gregor, 2006, p. 613), explanatory theory in IS, *"links the natural world, the social world, and the artificial world of human constructions."* The highly technical character of SDN infrastructures demanded an analysis that included tracing carefully matter involved, carefully addressing causality in social reality, and through detailed technical analysis, distilling general concepts relevant to answering the research question. On this foundation, the resulting proposed theoretical contributions could clearly define the meaning of the technical, social and sociotechnical designations of each uncovered generative mechanism, and the contingently causal relationships between Softwarisation, points of architectural control to which value has been assigned, and Social Disaggregation.

The analysis went beyond mere identification of categories and themes in the data (Kelle, 2014) to detail what intervenes between initial conditions and outcomes, as is typical of a logical structure of causation based on a process model (Markus & Robey, 1988; Avgerou, 2013). Through detailed analytical arguments, relationships that exist between the identified generative mechanisms, how interactions between actualised generative mechanisms are contingently causal to particular outcomes, and how these outcomes feature causally in the processes by which architecture evolves in ossified digital infrastructures, were uncovered. The explanation covered, temporality and interaction of actualised generative mechanisms, which generative mechanisms act transfactually and when, the consequences of the actualisation of the otherwise transfactually acting generative mechanisms in the context of architectural evolution and their relationships with other actualised generative mechanisms, how the sociotechnical generative mechanisms relate to the Softwarisation generative mechanism of morphogenesis, why Softwarisation with respect to a sociotechnically ossified architectural point of control to which value has been assigned contributes to the Social Disaggregation generative mechanism (i.e., computational ontology at runtime makes it difficult to re-ossify routers' implementation architecture as a hardware architectural point of control in a manner that preserves networking industry social internal relations that are favourable to vendors), and complexity and contingency of the Social Disaggregation generative mechanism. Elucidation of generative mechanisms served to demonstrate the adequacy of the causal explanations derived from the findings, in preference to alternative explanations.

In summary, the key attributes of explanatory theory (Markus & Robey, 1988; Gregor, 2006) are present. There is a defined theoretical construct to which the theory relates that delimits the scope of the proposed theoretical contributions: architectural evolution in sociotechnically ossified digital infrastructures. Theoretical constructs of the proposed theoretical contributions are explained: what each generative mechanism is, runtime, and computational ontology at runtime. Causal attribution, and causal associations, relationships, and processes are described and explained.

Next, I juxtapose the proposed theoretical contributions with extant IS literature on digital infrastructures to argue that they fulfil the requirements of theoretical contributions, as introduced in Chapter 3.

7.2.3 Complementing Architectural Evolution by Interconnection with Architectural Evolution through Softwarisation

The Introduction and Related Literature chapters explained that in IS research, digital infrastructure architecture is articulated as though it can be completely captured by modularity theory, and consequently core architectural evolution has been articulated by evolution via interconnection in a similar manner to deployment architectural evolution (Hanseth, 2001; Hanseth & Lundberg, 2001; Edwards, et al., 2007; Hanseth & Lyytinen, 2010; Grisot, et al., 2014). This modularity-confined understanding of architectural evolution suggests core architectural evolution via transitory gateways accompanied by an exercise of cultivating the sociotechnical installed base of digital infrastructures. I questioned, however, whether modularity theory alone holds sufficient explanatory power for architectural evolution in digital infrastructures.

Architectural evolution through Softwarisation suggests that modularity may be important (as evidenced by standardised northbound and southbound SDN controller interfaces), but the role of digital ontology, in particular the runtime dimension, features strongly in implementation and core architectural evolution in digital infrastructures. Architectural evolution through Softwarisation reveals that digital ontology admits more than changes to the data structure and functionality of digital entities, post their production and distribution (Manovich, 2002, pp. 27-48; Kallinikos, et al., 2010; Yoo, et al., 2010; Kallinikos, et al., 2013; Henfridsson, et al., 2014). Softwarisation exploits the digital ontology of digital infrastructures to promote architectural evolution that yields digital infrastructures in which runtime underlies the definition of architecture. The analysis also revealed that the runtime dimension of the digital ontology of digital infrastructures can be exploited, comparably to changing the constitution and functionality of digital entities, to promote core architectural evolution. These observations fall outside of what architecture by interconnection premised on modularity theory, can capture, thus extending understanding of architectural evolution in digital infrastructures, and illuminating an unarticulated role of digital ontology in architectural evolution in digital infrastructures.

Against the distinctiveness of the theoretical contributions, it could be argued that the implementation of SDN infrastructures, could be understood as a strategy for introducing modularity's substitutability principle into specific areas of networking infrastructures, and more widely to re-modularise boundaries in the networking industry. Still, runtime and computational ontology at runtime make it too narrow to summarise the advent of SDN infrastructures solely in terms of modularity theory. Runtime and computational ontology at runtime emphasize dynamic continuous ontological re-constitution (digitally) that extends to core architectural evolution, and feature consequentially as distinct issues within complex and significant broader social issues. They make apparent phenomena unreported by modularity-based theoretic analysis, and therefore require separate articulation as distinct explanatory theoretic tools.

Architectural evolution through Softwarisation challenges and extends the notion of upward flexibility (Tilson, et al., 2010b) to apply to core architectural evolution. Upward flexibility in IS literature on digital infrastructure has been described as an inherent accommodation in digital infrastructures for making use of lower layers in ways that may have been unanticipated at the time that those layers were created (Benkler, 2000; Zittrain, 2008; Tilson, et al., 2010b; Yoo, et al., 2010; Hanseth & Lyytinen, 2010). On the other hand, in the advent of SDN Infrastructures, Softwarisation actualised to migrate the control function to software that is externally applied to the lower layers of digital infrastructures, but additionally the applied software may impose different operational semantics and core architecture layering on infrastructure network hardware devices[62], and through computational ontology at runtime, create and evolve ephemeral core architectures for ephemeral virtual infrastructures. Outside of the results of this research, there is no finding in the IS

---

[62] See the brief discussion on this in section 6.3.6.1.

literature on digital infrastructures reporting that upward flexibility via the digital ontology of digital infrastructures can be exploited to modify core architecture in an existing digital infrastructure.

The research went beyond citing challenges of core architectural evolution and the associated alluding to the need for cultivation of a resistant installed base (Monteiro, 1998; Hanseth & Lyytinen, 2010; Aanestad & Jensen, 2011; Sanner, et al., 2014; Grisot, et al., 2014), to directly engage architectural evolution including contending with highly technical details, in order to explain architectural evolution. As a result, the proposed theoretical contributions broaden understanding of implementation and core architectural evolution in sociotechnically ossified digital infrastructures. Architectural evolution through Softwarisation transcends the simplicities of deployment architectural evolution (Hanseth, 2001; Hanseth & Lundberg, 2001; Edwards, et al., 2007; Egyedi & Spirco, 2011; Sanner, et al., 2014; Grisot, et al., 2014), and adds more explanatory utility than core architectural evolution by recombination (Bygstad, 2010; Henfridsson & Bygstad, 2013) – which is mostly an extension of architectural evolution by interconnection.

7.2.4 Extending Digital Ontology: Acknowledging the Ephemeral

Digital ontology in the context of software includes a runtime dimension. This is a key proposed theoretical contribution of this research. Explicit acknowledgement of software in the act of execution broadens what can be seen of core architectural evolution in digital infrastructures, and what can be seen of digital infrastructures themselves.

The observations made by the authors of (Kallinikos, et al., 2010; Kallinikos, et al., 2013) about the dependence of digital data on computational processing for its manifestation, and, through computational processing, the sustenance and modification of particular practices, in some ways convey themes that allude to computational ontology at runtime. However, computational ontology at runtime transcends the scope of the *operands* of function – analogous to the data processed by software routers – to include software in execution – such as software routers in execution that are part of ephemeral virtual infrastructures.

In the context of software, in contradistinction to data (i.e., digital data objects), it is at runtime that the distinctiveness of digital ontology becomes apparent. Yoo, et al. hint at runtime in their concept of doubly distributed innovation within layered modular architectures, in that the designation of component versus standalone product (such as in their Google Maps example) is distributed across firms, disciplines, communities etc., but more so, realised at runtime (Yoo, et al., 2010). Wegner's juxtaposition of statically analysable algorithms and the dynamism of executing interaction systems brings this point out more sharply (Wegner, 1997). He argues that while the form of software (i.e., the definition of an algorithmic function in code) can be analysed, and on that basis behavioural outcomes deterministically predicted, interaction systems are more complex and not in the same way reducible to static analysis and prediction. They necessitate runtime execution (Wegner, 1997). Similarly, here, architecture definition and instantiation in digital infrastructures are rendered runtime notions. It is from this perspective that the traditional IS conception of digital infrastructure as a perpetual collective of potential awaiting exploitation, is extended to include infrastructure as a runtime notion. More than this, that runtime and computational ontology at runtime can feature consequentially within complex and significant social issues, becomes perceptible.

Dynamically tailored, client application-specific, possibly transient, virtual networking infrastructures, describe digital infrastructures in ways not reported in IS research. The IS literature on digital infrastructures addresses shared, perpetual infrastructures (Star & Ruhleder, 1996; Hanseth, et al., 1996; Hanseth, 2001; Edwards, et al., 2007; Hanseth & Lyytinen, 2010; Tilson, et al., 2010b), but there have not been attempts to explain this kind of granular ephemeral conception of digital infrastructures. This may be partly because in the IS literature's treatment of digital infrastructures, concerns such as infrastructure formation as the outcome of the resolution of local and global tension, installed base inertia and cultivation, embeddedness and sociotechnical intertwining, adaptability to social changes, generative capacity and end user content production, control, and others have been used extensively to characterise digital infrastructures (Star & Ruhleder, 1996; Hanseth, et al., 1996; Hanseth & Monteiro, 1997; Ciborra, 2000; Hanseth & Lundberg, 2001; Orlikowski, 2007). See also (Nielsen & Aanestad, 2006; Ribes & Finholt, 2009; Sanner, et al., 2014; Rodon & Silva, 2015).

These concerns seek out what endures while implicitly excluding the ephemeral of digital infrastructures. Runtime and computational ontology at runtime, suggest that the perpetual of infrastructures can be complemented with acknowledgement of the ephemeral of infrastructures to facilitate better understanding of social phenomena such as Social Disaggregation.

Runtime and computational ontology at runtime, also underlie a new conceptualisation of the generative capacity of architecture which this research introduces to the IS literature on digital infrastructures. Instead of generative capacity of a singular architecture (Zittrain, 2008), generative capacity is extended to an architectural capacity to simultaneously support multiple heterogeneous, possibly incompatible, architectures. It is generative capacity for the creation and evolution of core architectures, rather than only generative capacity for new end-user usage scenarios and for digital infrastructures' expansion (deployment architecture evolution). Importantly, each architecture encapsulated by an ambivalent meta-architecture, maintains the generative capacity of a singular architecture[63] as per extant IS understanding. Thus, the proposed theoretical contributions make apparent that there are at least two types of generative capacity of architecture.

Very importantly, the exposition of runtime and computational ontology at runtime in digital infrastructures, demonstrates that alongside the *"when"* of infrastructures (Star & Ruhleder, 1996), the *"what"* of infrastructures remains equally significant. The analytic process which produced these proposed theoretical contributions required tracing highly technical details in order to distil ontological invariants that are independent of the confines of usage and yet implicated in complex and significant social events.

[63]With virtualisation, it is technically possible for one of these architectures to serve as a meta-architecture to others, creating multiple levels of indirection or recursion, but this falls outside of the scope of this research.

7.2.5 Digitalisation, Softwarisation and Social Disaggregation

What is the relationship between digitalisation (Tilson, et al., 2010b; Yoo, et al., 2010), Softwarisation and Social Disaggregation? Is Digitalisation and Softwarisation referring to the same thing?

According to Tilson, et al. (2010b, p. 2), digitalisation can be understood as a *"sociotechnical process of applying digitizing techniques to broader social and institutional contexts that render digital technologies infrastructural,"* where digitizing transforms analogue aspects of infrastructures (Manovich, 2002, pp. 18-48; Tilson, et al., 2010b; Yoo, et al., 2010) to conform to the requirements of binary signification. The process of digitalisation therefore alters the form and matter of infrastructures that underlie these social and institutional contexts (Tilson, et al., 2010a; Tilson, et al., 2010b), and similarly to Softwarisation, has been implicated in changes to social structure around infrastructures (Yoo, et al., 2010; Tilson, et al., 2010b).

Architectural evolution through Softwarisation exploits the *pre-existing* digital ontology of digital infrastructures, and therefore is temporally subsequent to digitalisation as it is defined in IS literature. Softwarisation is more than digitalisation. It is emergent out of the synchronic relations (Elder-Vass, 2007) of five distinct sociotechnical generative mechanisms that are expressed at a higher level of semantics than the strategic sociotechnical application of digitizing techniques. That is, the processes that describe these generative mechanisms' make-up, are centred on higher-level concepts (i.e., disaggregation, intelligence and authority migration, virtualisation, abstraction, and refactoring) that presume digitalisation as foundational. In the context of the advent of SDN infrastructures, Social Disaggregation cannot be understood as a corollary of the application of digitizing techniques to broader social and institutional contexts. Relative to the advent of SDN infrastructures, Social Disaggregation is facilitated by the aforementioned higher-level concepts.

7.2.6 Summary

This section used an analytic generalisation argument to demonstrate that the proposed theoretical contributions are in fact an explanatory theory. It showed how the key attributes of explanatory theory are present, and it evaluated the contributions to knowledge by demonstrating, through comparisons with extant IS literature on digital infrastructures, how the theoretical contributions enlighten an imperfect understanding of architectural evolution in sociotechnically ossified digital infrastructures, and an imperfect understanding of digital infrastructures themselves.

7.3 Scope of Generalisability

Details of the conditions under which a theory holds, complete its description (Eisenhardt, 1989; Firestone, 1993; Gregor, 2006; Polit & Beck, 2010; Kelle, 2014). The explanatory theory produced by this research is a middle-range theory, applicable specifically within the context of architectural evolution in sociotechnically ossified digital infrastructures. I make no claims about the generalisability of the theory to outside the scope of architectural evolution.

Softwarisation is always with respect to some identified point of sociotechnical ossification within digital infrastructures that is to be architecturally evolved. If the point of sociotechnical ossification is a point of architectural control to which some value has been assigned it may lead to a form of Social Disaggregation (of morphogenesis or morphostasis). The Technical Capacity for Flexibility and Accommodating of Change and the Technical Capacity for Introducing Capabilities to Generified Hardware via Software-Definition technical generative mechanisms, actualise to promote continued core architectural evolution via runtime software-definition of architecture. The Technical Capacity for Dynamic Tailoring of Infrastructure technical generative mechanism, actualises to promote continued core architectural evolution in the runtime software-definition of infrastructure. These three technical generative mechanisms are not peculiarly proceeding from SDN infrastructures. They are rooted in runtime and thus may exist in other types of digital infrastructures that are the outcome of Softwarisation. It has already been confirmed in IS literature on digital infrastructures that generally, Installed Base Cultivation causally features in architectural evolution in digital infrastructures (Monteiro, 1998; Hanseth & Lyytinen, 2010; Aanestad & Jensen, 2011; Sanner, et al., 2014; Grisot, et al., 2014; Reichertz, 2014; Kelle, 2014, pp. 561-562).

I contend on this basis, that the theoretical contributions hold outside of the context of networking infrastructures, and are applicable to digital infrastructures in general. Notwithstanding, the use of the theory in future research (such as to analyse the advent of cloud or NFV infrastructures, for example) might refine more definitively the theory's scope of generalisability.

7.4 A Criticism of Critical Realism for IS Research

Finally, at the end of Chapter 3, I mentioned that somewhat contradictory to arguments that have proposed critical realism as a suitable philosophy for IS research (Mutch, 2002; Carlsson, 2004; Mingers, 2004a; Mingers, 2004b; Smith, 2006; Carlsson, 2012), some incongruences were discovered during the undertaking of this research. The objective of this research was to identify generative mechanisms that promote architectural evolution in sociotechnically ossified digital infrastructures. A combination of social generative mechanisms (one of which promotes architectural evolution), technical generative mechanisms and sociotechnical generative mechanisms were identified. The question however, is with what structure are these generative mechanisms associated? For the technical generative mechanisms, the answer is relatively straightforward: they are of digital infrastructures. For the social generative mechanism (Installed Based Cultivation), it is of the networking industry. Specifying the structure with which the sociotechnical generative mechanisms are instantiated, in a manner that maintains philosophical consistency with critical realism, is more challenging.

To understand why, recall that according to critical realism, though structure and human actions partake in an interplay that leads to morphostasis or morphogenesis, they are ontologically independent, and are analytically distinguishable on the basis of the historicity and pre-existence of structure (Archer, 1982; Archer, 1995; Archer, 2000). Archer asserts that morphogenesis and morphostasis are generative mechanisms (Archer, 1995, p. 217). Human mediation of the generative mechanisms of a purely social structure leads to little concern when the outcome is morphostasis. It is the morphogenesis of structure that is of issue.

This research found that it is a *combination* of human agency mediating the powers of types of networking industry organisations such as network operators, systems integrators and vendors, and digital infrastructures' accommodation of change via exploitation of their digital ontology that constitute the sociotechnical generative mechanisms which promote architectural evolution in digital infrastructures. Similarly formulated sociotechnical generative mechanisms have been found in other critical realist IS research on digital infrastructures (Bygstad, 2010; Henfridsson & Bygstad, 2013). Through a perspectival switch (Hartwig, 2007, pp. 344-345), the accommodation of exploitation of their digital ontology may be understood as a power or technical generative mechanism of digital infrastructures to accept change. If the morphogenetic generative mechanisms are constitutively human agency and technical generative mechanisms, then per critical realism, an argument can be made that it is really human agency in an interplay with existing structure, meaning that these sociotechnical generative mechanisms are not generative mechanisms as defined by critical realism.

To resolve this, it is necessary to consider networking industry organisations such as network operators, systems integrators and vendors as social structures (with generative mechanisms and properties) that are necessarily internally related with networking infrastructures, to form second order emergent strata (Archer, 1995, pp. 202-218), such that though these organisations' generative mechanisms are only efficacious through the agency of humans, it is via *necessary internal relations* (Archer, 1995, p. 173; Archer, 2000; Sayer, 2000, p. 14) between the *generative mechanisms* of these organisations and the *generative mechanisms* of digital infrastructures that the sociotechnical generative mechanisms *emerge* (Collier, 1994, p. 110; Archer, 1995, pp. 14,172-179; Sayer, 2000, p. 14; Elder-Vass, 2007) initially acting *transfactually*. Notably, individuals by themselves do not have the power to architecturally evolve sociotechnically ossified networking infrastructures. These generative mechanisms belong to network industry organisations. Thus the sociotechnical generative mechanisms identified, are indeed generative mechanisms and not human agency in an interplay with existing structure.

But then, of what structure are they?

The particulars of the case of the advent of SDN infrastructures were focused on architectural evolution which, though it occurs in a social context, is of networking infrastructures. This precludes attributing the generative mechanisms solely to "sociotechnical" digital infrastructures, where the term "sociotechnical" and its boundaries are ambiguously defined. Rather, the "social" is of network industry organisations such as network operators, systems integrators, vendors and SDOs, and the “technical” is of the networking infrastructures. Therefore, the structure with which the sociotechnical generative mechanisms are instantiated, is a second-order emergent sociotechnical structure of internally related network industry organisations and networking infrastructures[64].

In a critical realist IS article published after this thesis’ analysis was completed, some IS researchers faced with the same challenge of identifying the structure to which sociotechnical generative mechanisms belong, similarly proposed a *“techno-organizational context”* as the structure[65] (Bygstad, et al., 2016, p. 87). Beside their narrower scope of technology within a single organisational context, there is a significant difference between what I have stated here and the findings of Bygstad, et al. (2016). It can be argued that by following Volkoff et al. (2013) to bring together affordance theory and critical realism, Bygstad, et al. (2016) misapprehended the interplay of human agency and structure as sociotechnical generative mechanisms. This is precisely because of the narrow scope of the organisational context, which is different from the scope of digital infrastructures (Hanseth, et al., 1996; Monteiro, 1998; Edwards, et al., 2007; Hanseth & Lyytinen, 2010). Specifically, individuals might not be mediating the generative mechanisms of organisations when exploiting the flexibility afforded by a technology in a micro-context within an organisation, and as

---

[64] That the Softwarisation sociotechnical generative mechanism emerges out of the synchronic relations of the five sociotechnical generative mechanisms does not, per emergence, necessitate another separate emergent structure with which Softwarisation alone is instantiated – neither is the existence of such an emergent structure proscribed by anything in critical realist emergence (Archer, 1982; Collier, 1994, pp. 107-134,138-141; Archer, 1995, pp. 135-161,172-183; Archer, 1998; Archer, 2000; Morgan, 2007a; Elder-Vass, 2007). In this research, however, no other structure aside from the second-order emergent sociotechnical structure of internally related network industry organisations and networking infrastructures, was uncovered.

[65] Rose, et al. (2004) also proposed an explanation of sociotechnical structure though not using critical realism.

stated before, individuals by themselves do not have the power to architecturally evolve sociotechnically ossified networking infrastructures. In support, Volkoff and Strong (2013, p. 823) admit that generative mechanisms are a broader concept than affordances. Affordances belong to a micro-context (Volkoff & Strong, 2013, p. 823). The formulation of sociotechnical generative mechanisms as emergent from necessary internal relations (Archer, 1995, p. 173; Archer, 2000; Sayer, 2000, p. 14) between the generative mechanisms of organisations and the generative mechanisms of digital infrastructures, as I have explained here, remains at the level of generative mechanisms, thus avoiding the problem in Bygstad, et al. (2016) altogether.

Having answered the question of with what structure are the generative mechanisms associated, the incongruences that challenge arguments that have proposed critical realism as a suitable philosophy for IS research (Mutch, 2002; Carlsson, 2004; Mingers, 2004a; Mingers, 2004b; Smith, 2006; Carlsson, 2012), become apparent. In the morphogenesis of structure, generative mechanisms instantiated with structure feed back onto that structure to transform it – though in an interplay with deliberate human actions (Archer, 1982; Archer, 1995). The findings of this research suggest that what transforms networking infrastructures are synchronically arranged morphogenetic generative mechanisms that are not fully instantiated with networking infrastructures – the very structure whose morphogenesis is in focus. Those generative mechanisms are emergent, and instantiated with a second-order emergent sociotechnical structure. The detailed explication of digital ontology in this thesis renders ambiguously designating this structure as the digital infrastructure using broad "sociotechnical" definitions, an inadequate solution.

In an atheoretical practical sense, this is not a problem – nor does it matter. But the inconsistency, rooted in the absence of dedicated consideration of digital technology in formative critical realism writings (Mutch, 2002, p. 488), is a significant criticism of the morphogenetic approach and broadly of critical realism[66]. As such, though useful for causal explanation, the suitability of critical realism as a philosophy for IS research remains open to further investigation.

[66] Based on problems in both the morphogenetic and TMSA accounts of the transformation of structure.

8 Conclusion

8.1 Summary of Theoretical Contributions

The objective of this research, was to develop an explanatory theory of how architectural evolution in sociotechnically ossified digital infrastructures occurs. This thesis introduced *architectural evolution through Softwarisation* as an explanatory theory for how architectures evolve in digital infrastructures, that is complementary to extant IS theorising on the subject. Architectural evolution through Softwarisation exploits the digital ontology of digital infrastructures in ways not reported prior to this point in IS literature on architecture in digital infrastructures.

Softwarisation is an emergent sociotechnical generative mechanism of morphogenesis and is underpinned by the social generative mechanism of Installed Base Cultivation. Softwarisation is always with respect to a point of sociotechnical ossification within digital infrastructures to which if value is assigned, a Social Disaggregation generative mechanism of morphogenesis or morphostasis may trigger and actualise. Post-morphogenesis, three technical generative mechanisms that are rooted in runtime, namely, a Technical Capacity for Flexibility and Accommodating of Change, a Technical Capacity for Introducing Capabilities to Generified Hardware via Software-Definition, and a Technical Capacity for Dynamic Tailoring of Infrastructure, may emerge to promote continued core architectural evolution in structurally elaborated digital infrastructures.

8.2 Corollary Theoretical Contributions

Key theoretical insights were rooted in the notion of software *in the act of executing*. The thesis demonstrated a need to extend digital materiality, reframed as *digital ontology*, in the context of software, to include *runtime*. It identified for the first time in IS research on digital infrastructures, *ephemeral architecture*, a runtime notion, as facilitating architectural ambivalence that underlies continued core architectural evolution, and co-existence of multiple heterogeneous possibly incompatible architectures in digital infrastructures. It extended the articulation of digital infrastructures for the first time in IS research to include a class of digital infrastructures that is ontologically a runtime notion: *ephemeral infrastructure*.

It extended theorising on the generative capacity of architecture by demonstrating that there are at least two types of generative capacities of architecture: a generative capacity for the creation and evolution of core architectures as accommodated by an ambivalent meta-architecture – as in SDN infrastructures, and the already theorised generative capacity of a singular architecture (Zittrain, 2008) that lends itself to continuous innovative exploitation that does not change underlying architecture. It identified for the first time in IS research, *computational ontology* that is purely virtual and constituted at runtime as the outcome of computation, and exists solely at runtime. It explained computational ontology at runtime as underlying ephemeral architecture and ephemeral infrastructure. Further, it demonstrated that runtime, and computational ontology at runtime, are not a superfluous academic pursuit, as exemplified by how they pervade the complexity of the social necessary internal relations in the networking industry.

As original corollary theoretical contributions, I argue that runtime, and computational ontology at runtime provide valuable theoretical tools through which digital infrastructures can be generally analysed.

IS theorising on architectural evolution in digital infrastructures, deems it not possible to achieve underlying architectural evolution of existing digital infrastructures without necessitating some form of infrastructure replacement, because it is premised on modularity theory and not as strongly on the leveraging of the digital ontology of digital infrastructures. One of the implications of this research's contributions is added knowledge that upward flexibility, via the digital ontology of digital infrastructures, can be exploited, as seen in the case study, to evolve underlying architecture in existing digital infrastructures. This is notable, and it is rooted in runtime. It is a finding that has hitherto not been reported in the IS literature on digital infrastructures.

Another important implication of these contributions is the assertion that the make-up of digital infrastructures is important. It was necessary to comprehend thoroughly the "digital" of digital infrastructures, in order to uncover how architectural evolution in sociotechnically ossified digital infrastructures occurs – and this led to the aforementioned corollaries. This thesis has shown that alongside the *"when"* of infrastructure, i.e. when does something become infrastructural (Star & Ruhleder, 1996), grasping the details of *"what"* is the infrastructure, regardless of how technical the details or the exercise of uncovering them may be, is still a valid and important pursuit, as demonstrated via the revelatory case of the advent of SDN infrastructures.

Finally, as a contribution to the use of theory in IS research, an evaluation of the philosophical consistency of critical realism in this research's context of architectural evolution in sociotechnically ossified digital infrastructures, found that though useful for causal explanation, the general suitability of critical realism as a philosophical base for IS research (particularly in the study of the transformation of structures that are in some way technological) remains open to further investigation.

9 Appendix A

Infrastructure Virtualisation Sample Research Access Request Letter

<DATE>

Dr Carsten Sørensen
Reader
Department of Management
Information Systems and Innovation Group
Houghton Street
London WC2A 2AE
New Academic Building 3.11
<CONTACT DETAILS>

<INTERVIEWEE NAME>,
<INTERVIEWEE ORGANISATION>

Research Access for Reuel Ocho:

I am Dr. Carsten Sørensen, Reader (Associate Professor) in the Information Systems and Innovation Group, in the Department of Management, at the London School of Economics.

My student, Reuel Ocho, is currently conducting his PhD research at the LSE to get a holistic understanding of how virtual infrastructures are being fundamentally transformed by the advent of the software-defined model, and what are the likely implications for organisations that adopt software-defined virtual infrastructures.

Already Reuel has had a strong response from experts at virtualisation technology vendors whom he has been interviewing to understand how they perceive the business impact of cloud-based virtual infrastructures.

I would be grateful if you can give Reuel an opportunity to arrange to include your expert views in his research. It would involve interviewing five or six persons for no more than one hour each, at a time arranged at your convenience, in order to get a comprehensive understanding of views within your organisation. The names of persons interviewed, and your organisation's name will be made anonymous in any research results that we may publish, and we will also share any published paper of our research findings with you. Should you be interested in participating, please contact Reuel on email (<EMAIL CONTACT DETAILS>).

If you have further questions, please do not hesitate to contact me.

Yours Sincerely,

Dr. Carsten Sørensen

Software-Defined Networking Sample Research Access Request Letter

<DATE>

Dr Carsten Sørensen
Reader (Associate Professor)
Department of Management
Information Systems and Innovation Group
Houghton Street
London WC2A 2AE
New Academic Building 3.11
<CONTACT DETAILS>

<INTERVIEWEE NAME>,
<INTERVIEWEE ORGANISATION>

Research Access for Reuel Ocho:

I am Dr. Carsten Sørensen, Reader (Associate Professor) in the Information Systems and Innovation Group, in the Department of Management, at the London School of Economics.

My student, Reuel Ocho, is currently conducting his PhD research at the LSE to get a holistic understanding of how virtual infrastructures are being fundamentally transformed by the advent of the software-defined model, and what are the likely implications for organisations that adopt software-defined virtual infrastructures.

Already Reuel has had a strong response from experts in networking and virtualisation technology.

I would be grateful if you can give Reuel an opportunity to arrange to include some of your expert perspectives in his research in order to get a comprehensive understanding of the formative views surrounding software-defined virtual infrastructures. Your name and your organisation's name will be made anonymous in any research results that we may publish, and we will also share any published paper of our research findings with you. Should you be interested in participating, please contact Reuel on email (<EMAIL CONTACT DETAILS>).

If you have further questions, please do not hesitate to contact me.

Yours Sincerely,

Dr. Carsten Sørensen

Software-Defined Networking Sample Introductory Email/LinkedIn Correspondence

Dear <INTERVIEWEE>,

I'm a PhD student at the London School of Economics and Political Science. I'm undertaking research that aims to get a holistic understanding of how networking infrastructures are being transformed by the advent of the software-defined model, and the implications for organisations involved.

So far I've had a strong response from experts in the networking industry. Given your leadership and expertise in networking, I wanted to know, whether I could arrange an interview with you (for no more than 1 hour) to discuss your perspectives on SDN.

Please let me know your interest, and I will provide a letter which formally introduces the research.

Kind Regards,

Reuel Ocho

10 Appendix B

Software-Defined Networking Sample Topic Guide Questions

How would you describe the problem statement for which SDN is a solution?
How would you explain what is meant by the term "software-defined"?
Why now? What has given SDN the momentum that we see at this particular time? /   For example, why are so many large telecommunications companies so interested in what it has to offer? /   What role do you see SDN fulfilling and for whom?
What does "software-defined" bring about that can't already be done using orchestrated virtualised infrastructure?
So what exactly is changing, conceptually, operationally and technically, when networking infrastructure transitions from just being virtualised to being software-defined? /   In other words, what characteristics, capabilities, technical architecture, behaviour etc. distinguishes a software-defined networking infrastructure from a typical virtualised networking infrastructure?
Added Topic: Why is there an emphasis on Openness in SDN and what kind of openness are we talking about here? /   How does it matter what is opened versus what remains closed and for whom does it matter? /   How are they responding to openness?
How does programmability transform networking infrastructure? /   You can answer in terms of how both technical and non-technical people use or think of infrastructure.
What are businesses saying? What are they going after?   Is it a SDN infrastructure in and of itself? /   What is it that they're trying to get at?

In the commercial deployments of SDN that you've seen, what key challenges have you seen organisations face during their transition from a traditional network?
Based on your experience what are some key non-technical concerns about SDN?
Some industry concerns are that SDN by itself is not a complete solution, because inter-organisational controller to controller coordination is impractical. / What are your thoughts on this?
Given the goals of SDN, and further of NFV, what would you say is becoming the role of hardware in a software-defined infrastructural context?
What to you makes up the infrastructure, in a software-defined context?
At present networking infrastructure is organised following a layered modular architecture across hardware and software. / How does the introduction of NFV or SDN to today's networking infrastructure affect the existing layered modular architecture across hardware and software? / Does it affect it?
What risks emerge when networking infrastructure is defined fundamentally in terms of software and less in terms of hardware?
What are the implications for organisational control of networking infrastructure, when the SDN model enables application-driven rapid short-lived inter-networking that can cross organisational and other boundaries?
Based on your experience, what essential things do you see missing from the understanding of organisations/adopters/technology contributors' of the wider implications of SDN and NFV or more generally of software-defined infrastructure? / *Or from the other side:* What understanding do you think those who are defining SDN do not fully have of the concerns of organisations that they are hoping will adopt SDN?

11 Appendix C

Infrastructure Virtualisation Sample Topic Guide Questions

Based on your experience, what are some business concerns addressed through infrastructure virtualisation?
How would you describe virtualisation in a business sense? / Can you provide some examples of the kind of business factors that indicate that infrastructure virtualisation is a preferred option? / What have you seen as being the benefits of virtualisation to those kinds of businesses? / Based on your experience, under what conditions should these businesses go ahead and virtualise their infrastructure? / Based on your experience, how should these businesses virtualise?
When deciding whether to virtualise [part of] a business' infrastructure what are some of the things that you consider? / What factors in your decision making processes?
Can you provide an example of how you begin to virtualise some infrastructure and what are the factors involved in determining the virtualisation strategy? / What are implications for different lines of the business that share the infrastructure that is to be virtualised?
What are businesses saying? What are they going after? Is it a virtualised infrastructure in and of itself? What is it that they're trying to get at?
In your experience, under what circumstances might the risks of infrastructure virtualisation leave a business without intended benefits? / How can a business identify these risks beforehand?
Under what circumstances would you advise a business, against virtualising its infrastructure? /

How could a business today survive, without any virtualised infrastructure?
Where is the emphasis in virtualisation (compute/network/storage/etc.) today and what have you seen as the [business] reasons for this?
When you encounter organisations, what essential understanding do you see missing from their virtualisation strategy?
Based on your experience of strategically implementing virtual infrastructures, how would you explain the relationship between virtualisation and cloud computing? / When does a virtualised infrastructure become a "cloud" for a business? / Can you describe how a cloud is operationally different from a virtualised infrastructure? Describe a "cloud" to me. What is the cloud doing? / How does cloud computing fulfill purposes that are distinct from virtualisation? / Is cost really as big a factor as it has been made out to be, because extended use of the cloud (such as a public cloud) could create expenses equivalent to buying a server or new storage. / How have you seen this kind of agility introduce problems to a business? / How should a business attempt to address these kinds of problems?
Where would you place software-defined-* (Networking, Storage) in relation to virtualisation and cloud computing? / Describe software-defined networking to me. What is the software-defined network doing? / What role do you see software-defined-* fulfilling for businesses? / How is it operationally different for a business?
Would you say that software-defined-* is different depending on whether or not it is implemented/accessed in a cloud-computing context? / In what way is it different?
What are some of the technical limitations that confine what virtualisation can do for businesses right now? / Which ones do you consider to be are long versus short term? / Can any be overcome now?

12 Appendix D

Sample of Technical Data used for First Unit of Analysis

All URLs verified as accessible as of December 9th, 2016.

Publication Year	Purpose	Document Type	Organisation	Topic	Online URL
2005	Architectural History of SDN	Research Presentation	Stanford University, Carnegie Mellon University, University of California Berkeley	Architectural History of SDN Southbound API/Switches	http://archive.openflow.org/documents/OpenFlow.ppt
2007	Architectural History of SDN	Blog	OpenFlow Consortium	Architectural History of SDN Southbound API/Switches	http://archive.openflow.org/wp/
2008	Architectural History of SDN	Specification	OpenFlow Consortium	Architectural History of SDN Southbound API/Switches	http://archive.openflow.org/documents/openflow-spec-v0.8.9.pdf
2008	Architectural History of SDN	Article	Proceedings of the 4th ACM/IEEE Symposium on Architectures for Networking and Communications Systems	Architectural History of SDN Southbound API/Switches	http://doi.acm.org/10.1145/1477942.1477944
2009	Architectural History of SDN	Specification	OpenFlow Consortium	Architectural History of SDN Southbound API/Switches	http://archive.openflow.org/documents/openflow-spec-v0.9.0.pdf

2009	Architectural History of SDN	Specification	OpenFlow Consortium	Architectural History of SDN Southbound API/Switches	http://archive.openflow.org/documents/openflow-spec-v1.0.0.pdf and https://www.opennetworking.org/images/stories/downloads/sdn-resources/onf-specifications/openflow/openflow-spec-v1.0.0.pdf
2012	Architectural History of SDN	Specification	Open Networking Foundation	Architectural History of SDN Southbound API/Switches	https://www.opennetworking.org/images/stories/downloads/sdn-resources/onf-specifications/openflow/openflow-spec-v1.3.1.pdf
2012	Architectural History of SDN	Specification	OpenFlow Consortium	Architectural History of SDN Southbound API/Switches	https://www.opennetworking.org/images/stories/downloads/sdn-resources/onf-specifications/openflow/openflow-spec-v1.0.1.pdf
2015	Architectural History of SDN	Specification	Open Networking Foundation	Architectural History of SDN Southbound API/Switches	https://www.opennetworking.org/images/stories/downloads/sdn-resources/onf-specifications/openflow/openflow-switch-v1.5.1.pdf
2009-2010	Architectural History of SDN	Archival Documents	OpenFlow Consortium	Architectural History of SDN Southbound API/Switches	http://archive.openflow.org/wk/index.php/OpenFlow_Meeting

2005	Architectural History of SDN	Archived Research Project Presentation	Stanford University	Early SDN Architecture	http://yuba.stanford.edu/ethane/files/ms_ethane.ppt
2005	Architectural History of SDN	Archived Research Project Website	Stanford University	Early SDN Architecture	http://yuba.stanford.edu/ethane/files/sane.ppt
2006	Architectural History of SDN	Archived Research Project Website	Stanford University	Early SDN Architecture	http://yuba.stanford.edu/ethane/
2006	Architectural History of SDN	Archived Research Project Presentation	Stanford University	Early SDN Architecture	http://yuba.stanford.edu/ethane/files/Ethane_Security_forum.ppt
2006	Architectural History of SDN	Article	Stanford University, Carnegie Mellon University, University of California Berkeley	Early SDN Architecture	http://yuba.stanford.edu/ethane/sane.pdf
2007	Architectural History of SDN	Article	ACM SIGCOMM Computer Communication Review	Early SDN Architecture	http://doi.acm.org/10.1145/1282427.1282382
2007	Architectural History of SDN	PhD Thesis	Stanford University	Early SDN Architecture	http://yuba.stanford.edu/~casado/mcthesis.pdf
2008	Architectural History of SDN	Article	ACM SIGCOMM Computer Communication Review	Early SDN Architecture	http://doi.acm.org/10.1145/1384609.1384625
2009	Architectural History of SDN	Article	IEEE/ACM Transactions on Networking	Early SDN Architecture	https://doi.org/10.1109/TNET.2009.2026415

2013	Architectural History of SDN	Archival Documents	Open Networking Foundation	History of OpenFlow and SDN	https://www.opennetworking.org/images/stories/downloads/about/onf-what-why.pdf
2009	Architectural History of SDN	Article	OpenFlow Consortium	Network Virtualisation	http://archive.openflow.org/downloads/technicalreports/openflow-tr-2009-1-flowvisor.pdf
2013	Architecture of SDN Infrastructure	White Paper	Cisco	Alternative SDN Infrastructure	http://www.cisco.com/c/dam/en_us/solutions/industries/docs/gov/cis13090_sdn_sled_white_paper.pdf
2014	Architecture of SDN Infrastructure	White Paper	Cisco	Alternative SDN Infrastructure	http://www.cisco.com/c/en/us/solutions/collateral/data-center-virtualization/application-centric-infrastructure/white-paper-c11-733456.pdf
2016	Architecture of SDN Infrastructure	White Paper	Cisco	Alternative SDN Infrastructure	http://www.cisco.com/c/en/us/solutions/collateral/data-center-virtualization/application-centric-infrastructure/white-paper-c11-736899.pdf
2013	Architecture of SDN Infrastructure	Industry Report	NTT Communications/IDC	Details of Network Operator SDN Infrastructure	http://www.ntt.com/content/dam/nttcom/affiliate/cmn/pdf/resouces/analysis/idc_marketscope_en.pdf

2015	Architecture of SDN Infrastructure	Industry Report	NTT Communications/Gartner	Details of Network Operator SDN Infrastructure	URL Removed from http://www.eu.ntt.com/en/resources/analyst-reports.html Available at: https://www.gartner.com/doc/3078021/ntt-communications-sdn-nfv-deployment
2016	Architecture of SDN Infrastructure	Industry Report	NTT Communications/Current Analysis	Details of Network Operator SDN Infrastructure	http://www.ntt.com/content/dam/nttcom/affiliate/cmn/pdf/resouces/analysis/current_analysis_company_assessment_2016.pdf
2013	Architecture of SDN Infrastructure	Website	Juniper	Production SDN Controller	http://www.opencontrail.org/
2013	Architecture of SDN Infrastructure	Website	Juniper	Production SDN Controller	http://www.juniper.net/uk/en/products-services/sdn/contrail/
2013	Architecture of SDN Infrastructure	Website	OpenDaylight Foundation/Linux Foundation	Production SDN Controller	https://www.opendaylight.org/
2013	Architecture of SDN Infrastructure	Press Release	OpenDaylight Foundation/Linux Foundation	Production SDN Controller	https://www.opendaylight.org/news/foundation-news/2013/09/opendaylight-

					project-releases-new-architecture-details-its-software
2014	Architecture of SDN Infrastructure	Archived Presentation	OpenDaylight Foundation/Linux Foundation	Production SDN Controller	https://wiki.opendaylight.org/images/archive/6/63/20150228235147%21Helium-diagram.pptx
2015	Architecture of SDN Infrastructure	Product Data Sheet	Brocade Communications Systems	Production SDN Controller	Document Replaced at URL
2015	Architecture of SDN Infrastructure	Archived Release Details	OpenDaylight Foundation/Linux Foundation	Production SDN Controller	https://www.opendaylight.org/lithium
2016	Architecture of SDN Infrastructure	Product Data Sheet	Brocade Communications Systems	Production SDN Controller	http://www.brocade.com/en/backend-content/pdf-page.html?/content/dam/common/documents/content-types/datasheet/brocade-sdn-controller-ds.pdf
2016	Architecture of SDN Infrastructure	Website	OpenDaylight Foundation/Linux Foundation	Production SDN Controller	https://www.opendaylight.org/odlbe
	Architecture of SDN Infrastructure	Website	Brocade Communications Systems	Production SDN Controller	http://www.brocade.com/en/products-services/software-networking/sdn-controllers-applications/sdn-controller.html

	Architecture of SDN Infrastructure	Website	Brocade Communications Systems	Production SDN Controller	http://www.brocade.com/en/support/document-library/dl-segment-products-os-detail-page.brocadevyattacontroller.product.html
	Architecture of SDN Infrastructure	Website	Juniper	Production SDN Controller	http://www.opencontrail.org/opencontrail-architecture-documentation/
	Architecture of SDN Infrastructure	Website	Juniper	Production SDN Controller	http://www.opencontrail.org/network-virtualization-architecture-deep-dive/
	Architecture of SDN Infrastructure	Website	OpenDaylight Foundation/Linux Foundation	Production SDN Controller	https://www.opendaylight.org/opendaylight-features-list
	Architecture of SDN Infrastructure	Website	OpenDaylight Foundation/Linux Foundation	Production SDN Controller	https://www.opendaylight.org/software/release-archives
2015	Architecture of SDN Infrastructure	Product Data Sheet	Radisys	SDN Forwarding Plane/SDN Switching	http://www.radisys.com/assets/flowengine-tde-platforms-intelligent-traffic-distribution-systems
	Architecture of SDN Infrastructure	Website	Radisys	SDN Forwarding Plane/SDN Switching	http://www.radisys.com/flowengine

2013	Architecture of SDN Infrastructure	White Paper	Citrix	SDN Infrastructure	https://www.citrix.com/content/dam/citrix/en_us/documents/products-solutions/drive-intelligence-into-next-generation-networks.pdf
2014	Architecture of SDN Infrastructure	White Paper	Big Switch	SDN Infrastructure	http://go.bigswitch.com/rs/974-WXR-561/images/BigSwitch_BigCloudFabric_WP_FINAL.pdf
2014	Architecture of SDN Infrastructure	White Paper	Cisco	SDN Infrastructure	http://www.cisco.com/c/dam/en/us/solutions/collateral/service-provider/open-network-environment-service-providers/white-paper-c11-732672.pdf
2014	Architecture of SDN Infrastructure	White Paper	Cisco	SDN Infrastructure	http://www.cisco.com/c/en/us/solutions/collateral/service-provider/open-network-environment-service-providers/white-paper-c11-732587.pdf
2014	Architecture of SDN Infrastructure	RFC/Recommendation/Informational	IETF	SDN Infrastructure	https://tools.ietf.org/html/rfc7149

2014	Architecture of SDN Infrastructure	RFC/Recommendation/Informational	ITU-T	SDN Infrastructure	https://www.itu.int/rec/T-REC-Y.3300-201406-I/en
2015	Architecture of SDN Infrastructure	White Paper	Arista	SDN Infrastructure	URL Removed Latest document version available here: https://www.arista.com/assets/data/pdf/Whitepapers/SDCN_Whitepaper.pdf
2015	Architecture of SDN Infrastructure	RFC/Recommendation/Informational	IETF	SDN Infrastructure	https://tools.ietf.org/html/rfc7426
2015	Architecture of SDN Infrastructure	Solution Brief	Juniper	SDN Infrastructure	https://www.juniper.net/assets/us/en/local/pdf/solutionbriefs/3510516-en.pdf
2016	Architecture of SDN Infrastructure	White Paper	Big Switch	SDN Infrastructure	http://go.bigswitch.com/rs/974-WXR-561/images/BCF-White-Paper-Secure%20and%20Resilient%20SDN-2.pdf
2016	Architecture of SDN Infrastructure	Product Data Sheet	Big Switch	SDN Infrastructure	http://go.bigswitch.com/rs/974-WXR-561/images/Big%20Cloud%20Fabric%20Datasheet%20WEB.pdf

	Architecture of SDN Infrastructure	Website	Big Switch	SDN Infrastructure	http://www.bigswitch.com/products/big-cloud-fabrictm/big-cloud-fabric-0
	Architecture of SDN Infrastructure	Website	Ericsson	SDN Infrastructure	https://www.ericsson.com/networks/topics/sdn
	Architecture of SDN Infrastructure	Website	IEEE	SDN Infrastructure	http://sdn.ieee.org/
2012	Architecture of SDN Infrastructure	White Paper	IBM	SDN Infrastructure/Production SDN Controller	URL Removed
2012	Architecture of SDN Infrastructure	Solution Brief	Nicira	SDN Infrastructure/Production SDN Controller	URL Removed
2012	Architecture of SDN Infrastructure	White Paper	Nicira	SDN Infrastructure/Production SDN Controller	URL Removed
2012	Architecture of SDN Infrastructure	White Paper	Nicira	SDN Infrastructure/Production SDN Controller	URL Removed
2013	Architecture of SDN Infrastructure	White Paper	IBM	SDN Infrastructure/Production SDN Controller	https://www.research.ibm.com/haifa/dept/stt/papers/QCW03028USEN.PDF

2013	Architecture of SDN Infrastructure	E-Book	Nuage Networks/Alcatel-Lucent	SDN Infrastructure/Production SDN Controller	URL Removed
2013	Architecture of SDN Infrastructure	Website	VMware	SDN Infrastructure/Production SDN Controller	https://www.vmware.com/uk/products/nsx.html
2013	Architecture of SDN Infrastructure	White Paper	VMware	SDN Infrastructure/Production SDN Controller	http://www.vmware.com/content/dam/digitalmarketing/vmware/en/pdf/whitepaper/products/nsx/vmware-nsx-network-virtualization-platform-white-paper.pdf
2013	Architecture of SDN Infrastructure	White Paper	VMware	SDN Infrastructure/Production SDN Controller	http://www.vmware.com/content/dam/digitalmarketing/vmware/en/pdf/products/nsx/vmware-nsx-on-cisco-n7kucs-design-guide.pdf
2014	Architecture of SDN Infrastructure	Technical Documentation	IBM	SDN Infrastructure/Production SDN Controller	http://www.redbooks.ibm.com/redbooks/pdfs/sg248203.pdf
2014	Architecture of SDN Infrastructure	Presentation	IBM	SDN Infrastructure/Production SDN Controller	URL Removed

2014	Architecture of SDN Infrastructure	Product Data Sheet	Juniper	SDN Infrastructure/Production SDN Controller	https://www.juniper.net/assets/us/en/local/pdf/datasheets/1000521-en.pdf
2014	Architecture of SDN Infrastructure	White Paper	VMware	SDN Infrastructure/Production SDN Controller	https://www.vmware.com/content/dam/digitalmarketing/vmware/en/pdf/products/nsx/vmware-nsx-palo-alto-networks.pdf
2015	Architecture of SDN Infrastructure	Technical Documentation	IBM	SDN Infrastructure/Production SDN Controller	http://www.redbooks.ibm.com/redbooks/pdfs/sg248238.pdf
2015	Architecture of SDN Infrastructure	White Paper	IBM	SDN Infrastructure/Production SDN Controller	https://public.dhe.ibm.com/common/ssi/ecm/ic/en/icw03011usen/ICW03011USEN.PDF
2015	Architecture of SDN Infrastructure	White Paper	Juniper	SDN Infrastructure/Production SDN Controller	https://www.juniper.net/assets/us/en/local/pdf/whitepapers/2000615-en.pdf
2015	Architecture of SDN Infrastructure	Technical Documentation	Juniper	SDN Infrastructure/Production SDN Controller	http://www.juniper.net/us/en/local/pdf/whitepapers/2000535-en.pdf
2015	Architecture of SDN Infrastructure	White Paper	Juniper/Vmware	SDN Infrastructure/Production SDN Controller	http://www.juniper.net/assets/cn/zh/local/pdf/whitepapers/2000589-en.pdf

2015	Architecture of SDN Infrastructure	White Paper	VMware	SDN Infrastructure/Production SDN Controller	http://www.vmware.com/content/dam/digitalmarketing/vmware/en/pdf/whitepaper/products/nsx/vmware-nsx-brownfield-design-and-deployment-guide-white-paper.pdf
2015	Architecture of SDN Infrastructure	Technical Documentation	VMware	SDN Infrastructure/Production SDN Controller	https://pubs.vmware.com/NSX-62/topic/com.vmware.ICbase/PDF/nsx_62_install.pdf
2015	Architecture of SDN Infrastructure	Technical Documentation	VMware	SDN Infrastructure/Production SDN Controller	https://communities.vmware.com/servlet/JiveServlet/downloadBody/27683-102-8-41631/NSX%20Reference%20Design%20Version%203.0.pdf
2016	Architecture of SDN Infrastructure	Website	Nuage Networks/Nokia	SDN Infrastructure/Production SDN Controller	http://www.nuagenetworks.net/wp-content/uploads/2014/11/MKT2014097652EN_NN_VSP_Virtualized_Services_Platform_R3_Datasheet.pdf
2016	Architecture of SDN Infrastructure	Product Data Sheet	VMware	SDN Infrastructure/Production SDN Controller	https://www.vmware.com/content/dam/digitalmarketing/vmware/en/pdf/products/nsx/vmware-nsx-datasheet.pdf

~2012	Architecture of SDN Infrastructure	Product Data Sheet	Nicira	SDN Infrastructure/Production SDN Controller	URL Removed
~2012	Architecture of SDN Infrastructure	Solution Brief	Nicira	SDN Infrastructure/Production SDN Controller	URL Removed
	Architecture of SDN Infrastructure	Website	Nuage Networks/Nokia	SDN Infrastructure/Production SDN Controller	http://www.nuagenetworks.net/products/virtualized-services-platform/
2013	Architecture of SDN Infrastructure	White Paper	Arista	SDN Infrastructure/Switching	https://www.arista.com/assets/data/pdf/Whitepapers/Arista_Cloud_Networks.pdf
2012	Architecture of SDN Infrastructure	Product Data Sheet	Tail-f	SDN Management Southbound Interfacing	http://www.tail-f.com/wordpress/wp-content/uploads/2013/02/Tail-f-ConfD-Datasheet.pdf
2013	Architecture of SDN Infrastructure	Product Brief	Tail-f	SDN Management Southbound Interfacing	http://www.tail-f.com/wordpress/wp-content/uploads/2014/04/Tail-f_ConfDUseCase-Established_rev-D-2014-04-09.pdf
2014	Architecture of SDN Infrastructure	White Paper	Cisco	SDN+NFV Infrastructure	http://www.cisco.com/c/en/us/solutions/collateral/service-provider/service-provider-strategy/brochure-c02-731348.pdf

2015	Architecture of SDN Infrastructure	Presentation	IETF	SDN+NFV Infrastructure	https://www.ietf.org/edu/tutorials/sdn-nfv-openflow-forces.pdf
2015	Architecture of SDN Infrastructure	White Paper	NTT Communications	SDN+NFV Infrastructure	http://www.nttcominsight.com/reinventing-enterprise-networks/
2016	Architecture of SDN Infrastructure	Technical Documentation	Verizon	SDN+NFV Infrastructure	http://innovation.verizon.com/content/dam/vic/PDF/Verizon_SDN-NFV_Reference_Architecture.pdf
2013	Background Information	White Paper	CloudEthernet	SDN+NFV Infrastructure/Cloud Networking Challenges	URL Removed (CloudEthernet.org domain removed, MEF did not replace URL).
2014	Boundaries Between SDN and NFV	Solution Brief	Open Networking Foundation	SDN+NFV Infrastructure	https://www.opennetworking.org/images/stories/downloads/sdn-resources/solution-briefs/sb-sdn-nvf-solution.pdf
2010	ForCES	RFC/Recommendation/Informational	IETF	Alternative SDN Infrastructure	https://tools.ietf.org/html/rfc5812
2010	ForCES	RFC/Recommendation/Informational	IETF	Alternative SDN Infrastructure	https://www.rfc-editor.org/info/rfc5810
2015	ForCES	Article	IEEE	Alternative SDN Infrastructure	http://ieeexplore.ieee.org/document/7118637/

2007	Hypervisor Virtualisation	White Paper	VMware	Virtualisation	https://www.vmware.com/content/dam/digitalmarketing/vmware/en/pdf/techpaper/ESXi_architecture.pdf
2011	Hypervisor Virtualisation	Technical Documentation	VMware	Virtualisation	URL Removed
2012	Hypervisor Virtualisation	White Paper	Bromium	Virtualisation	URL Removed
2013	Hypervisor Virtualisation	White Paper	Citrix	Virtualisation	https://www.citrix.com/content/dam/citrix/en_us/documents/products-solutions/powering-the-worlds-largest-clouds-with-an-open-approach.pdf
2013	Hypervisor Virtualisation	Product Data Sheet	Citrix	Virtualisation	https://www.citrix.com/content/dam/citrix/en_us/documents/products-solutions/citrix-xenserver-industry-leading-open-source-platform-for-cost-effective-cloud-server-and-desktop-virtualization.pdf
	Hypervisor Virtualisation	Website	Citrix	Virtualisation	http://xenserver.org/

2012	Network Functions Virtualisation	White Paper	ETSI NFV ISG	SDN+NFV Infrastructure	https://portal.etsi.org/NFV/NFV_White_Paper.pdf
2013	Network Functions Virtualisation	White Paper	ETSI NFV ISG	SDN+NFV Infrastructure	http://portal.etsi.org/NFV/NFV_White_Paper2.pdf
2013	Network Functions Virtualisation	Technical Specification	ETSI NFV ISG	SDN+NFV Infrastructure	http://www.etsi.org/deliver/etsi_gs/nfv/001_099/001/01.01.01_60/gs_nfv001v010101p.pdf
2014	Network Functions Virtualisation	White Paper	ETSI NFV ISG	SDN+NFV Infrastructure	http://portal.etsi.org/Portals/0/TBpages/NFV/Docs/NFV_White_Paper3.pdf
2014	Network Functions Virtualisation	Technical Specification	ETSI NFV ISG	SDN+NFV Infrastructure	http://www.etsi.org/deliver/etsi_gs/NFV-SWA/001_099/001/01.01.01_60/gs_nfv-swa001v010101p.pdf
2014	Network Functions Virtualisation	Technical Specification	ETSI NFV ISG	SDN+NFV Infrastructure	http://www.etsi.org/deliver/etsi_gs/NFV/001_099/003/01.02.01_60/gs_nfv003v010201p.pdf
2014	Network Functions Virtualisation	Technical Specification	ETSI NFV ISG	SDN+NFV Infrastructure	http://www.etsi.org/deliver/etsi_gs/NFV-INF/001_099/010/01.01.01_60/gs_nfv-inf010v010101p.pdf

2014	Network Functions Virtualisation	Technical Specification	ETSI NFV ISG	SDN+NFV Infrastructure	http://www.etsi.org/deliver/etsi_gs/NFV-INF/001_099/007/01.01.01_60/gs_NFV-INF007v010101p.pdf
2014	Network Functions Virtualisation	Technical Specification	ETSI NFV ISG	SDN+NFV Infrastructure	http://www.etsi.org/deliver/etsi_gs/NFV-INF/001_099/003/01.01.01_60/gs_nfv-inf003v010101p.pdf
2014	Network Functions Virtualisation	Technical Specification	ETSI NFV ISG	SDN+NFV Infrastructure	http://www.etsi.org/deliver/etsi_gs/nfv/001_099/002/01.02.01_60/gs_nfv002v010201p.pdf
2015	Network Functions Virtualisation	Technical Specification	ETSI NFV ISG	SDN+NFV Infrastructure	http://www.etsi.org/deliver/etsi_gs/NFV-INF/001_099/001/01.01.01_60/gs_nfv-inf001v010101p.pdf
2015	Network Functions Virtualisation	Technical Specification	ETSI NFV ISG	SDN+NFV Infrastructure	http://www.etsi.org/deliver/etsi_gs/NFV-INF/001_099/005/01.01.01_60/gs_NFV-INF005v010101p.pdf
2015	Network Functions Virtualisation	Technical Specification	ETSI NFV ISG	SDN+NFV Infrastructure	http://www.etsi.org/deliver/etsi_gs/NFV-INF/001_099/004/01.01.01_60/gs_nfv-inf004v010101p.pdf

2014	Network Virtualisation	White Paper	Cisco	Network Virtualisation	URL Removed An 2016 version that is almost identical can be found here: http://www.cisco.com/c/dam/en/us/solutions/collateral/service-provider/network-functions-virtualization-nfv/white-paper-c11-731958.pdf
2014	Network Virtualisation	Website	Rackspace	Network Virtualisation	https://developer.rackspace.com/docs/cloud-networks/v2/release-notes/
2012	Network Virtualisation	White Paper	VMware	Network Virtualisation/Virtual Switches	http://www.vmware.com/content/dam/digitalmarketing/vmware/en/pdf/techpaper/whats-new-vmware-vsphere-51-network-technical-white-paper.pdf
	Open Source Hardware/Whiteboxing	Website	Open Compute Project	SDN Infrastructure/Switching	http://www.opencompute.org/

2015	OpenFlow Compliant Switches	Product Data Sheet	Brocade Communications Systems	Production SDN Controller	Document Replaced at URL
2015	OpenFlow Compliant Switches	Product Data Sheet	Arista	SDN Infrastructure/Switching	https://www.arista.com/assets/data/pdf/AristaQRG.pdf
2013	OpenFlow-Based SDN reference architecture	Technical Report	Open Networking Foundation	Architecture	https://www.opennetworking.org/images/stories/downloads/sdn-resources/technical-reports/SDN-architecture-overview-1.0.pdf
2014	OpenFlow-Based SDN reference architecture	Technical Reference	Open Networking Foundation	Architecture	https://www.opennetworking.org/images/stories/downloads/sdn-resources/technical-reports/TR_SDN_ARCH_1.0_06062014.pdf
2014	OpenFlow-Based SDN reference architecture	Solution Brief	Open Networking Foundation	Architecture	https://www.opennetworking.org/images/stories/downloads/sdn-resources/solution-briefs/sb-of-enabled-transport-sdn.pdf
2016	OpenFlow-Based SDN reference architecture	Technical Recommendation	Open Networking Foundation	Architecture	https://www.opennetworking.org/images/stories/downloads/sdn-resources/technical-reports/SDN_Architecture_for_Transport_Networks_TR522.pdf
2016	OpenFlow-Based SDN	Technical Reference	Open Networking Foundation	Architecture	https://www.opennetworking.org/images/stories/downloads/sdn-resources/technical-reports/TR-

	reference architecture				521_SDN_Architecture_issue_1.1.pdf
2015	Product Information	Technical Documentation	Juniper	OpenFlow Support in Product	https://www.juniper.net/documentation/en_US/junos15.1/information-products/pathway-pages/junos-sdn/junos-sdn-openflow.pdf
	Product Information	Product Data Sheet	NTT Communications	SDN Infrastructure	http://www.ntt.com/content/dam/nttcom/hq/en/cmn/pdf/NTT-EC_DataSheet.pdf
2016	Product Information	Solution Brief	Verizon	SDN+NFV Infrastructure	http://www.verizonenterprise.com/resources/virtual-network-services-solutions-brief_en_xg.pdf
2016	Product Informtion	Solution Brief	AT&T	SDN+NFV Infrastructure	URL REMOVED. RE-branded Product Here: https://www.business.att.com/content/productbrochures/network-function-virtualization-product-brief.pdf
2015	Reasons for SDN	White Paper	Juniper	Network Operators' Motivation	https://www.juniper.net/assets/cn/zh/local/pdf/pov/3200050-en.pdf

2015	Reasons for SDN	White Paper	Verizon	Network Operators' Motivation	http://www.verizonenterprise.com/resources/reports/rp_digital-transformation-powers-your-business_en_xg.pdf
2016	Reasons for SDN	White Paper	IDC/Verizon	Network Operators' Motivation	http://www.verizonenterprise.com/resources/reports/wp_digital-transformation-obstacles-and-how-to-overcome-them_xg_en.pdf
2013	SDN Switch Management	Solution Brief	Arista	SDN Infrastructure/Switching	URL Removed Latest document version available here: https://www.arista.com/assets/data/pdf/Whitepapers/Arista_eAPI_FINAL.pdf
2015	SDN Switch Management/ Operating System	White Paper	Arista	SDN Infrastructure/Switching	URL Removed Latest document version available here: https://www.arista.com/assets/data/pdf/EOSWhitepaper.pdf
	SDN Switch Management/	Website	Open Compute Project	SDN Infrastructure/Switching	http://www.onie.org/

	Operating System				
2014	Technical architecture and architectural history of OpenFlow compliant switches.	Article	IEEE/Open Networking Foundation	Architectural History of SDN Southbound API/Switches	Original Document: https://www.opennetworking.org/images/stories/downloads/sdn-resources/IEEE-papers/evolution-of-sdn-and-of.pdf Peer Reviewed Article: http://dx.doi.org/10.1109/MC.2014.326
2015	Vendor Product Information	White Paper	Juniper	Network Operators' Motivation	https://www.juniper.net/assets/de/de/local/pdf/whitepapers/2000611-en.pdf

13 Works Cited

Aanestad, M. & Jensen, T. B., 2011. Building Nation-wide Information Iinfrastructures in Healthcare through Modular Implementation Strategies. *The Journal of Strategic Information Systems ,* 20(2), pp. 161-176.

Amazon Web Services, n.d.-a. *About AWS.* [Online] Available at: http://aws.amazon.com/about-aws/ [Accessed 2 September 2016].

Amazon Web Services, n.d.-b. *Amazon API Gateway.* [Online] Available at: https://aws.amazon.com/api-gateway/ [Accessed 2 September 2016].

Anderson, T., Peterson, L., Shenker, S. & Turner, J., 2005. Overcoming the Internet Impasse through Virtualization. *Computer,* April, 38(4), pp. 34-41.

Anon., n.d. *PlanetLab.* [Online] Available at: https://www.planet-lab.org/

Anon., n.d. *PlanetLab History.* [Online] Available at: https://www.planet-lab.org/history

Archer, M., 1998. Introduction: Realism in the Social Sciences. In: M. Archer, et al. eds. *Critical Realism: Essential Readings.* Routledge: Routledge, pp. 189-205.

Archer, M. S., 1982. Morphogenesis versus Structuration: On Combining Structure and Action. *British Journal of Sociology,* 33(4), pp. 455-483.

Archer, M. S., 1995. *Realist Social Theory: The Morphogenetic Approach.* s.l.:Cambridge University Press.

Archer, M. S., 2000. For Structure: its reality, properties and powers: A reply to Anthony King. *Sociological Review,* 48(3), pp. 464-472.

Arista Networks Inc., n.d.-a. *Arista - Software Driven Cloud Networking.* [Online] Available at: http://www.arista.com/en/solutions/software-driven-cloud-networking [Accessed 2 September 2016].

Arista, Networks Inc., n.d.-b. *Arista - Corporate Profile.* [Online] Available at: https://www.arista.com/en/company/company-overview [Accessed 2 September 2016].

Arnold, R., 1989. Software Restructuring. *Proceedings of the IEEE,* Apr, 77(4), pp. 607-617.

AT&T, 2013. *AT&T Vision Alignment Challenge Technology Survey: AT&T Domain 2.0 Vision White Paper,* s.l.: s.n.

AT&T, 2014. *AT&T Introduces the "User-Defined Network Cloud": A Vision for the Network of the Future.* [Online] Available at: http://www.att.com/gen/press-room?pid=25274&cdvn=news&newsarticleid=37439 [Accessed 2 September 2016].

AT&T, 2014. *AT&T Unveils "Network On Demand" Trial, IBM Cloud Agreement and New High-Speed Internet Services for Fiber-Ready Buildings for Business Customers.* [Online] Available at: http://about.att.com/story/att_unveils_network_on_demand_trial.html [Accessed 2 September 2016].

AT&T, 2015a. *AT&T and Brocade Join Forces to Extend Network on Demand Platform.* [Online] Available at: http://about.att.com/story/att_brocade_join_forces_to_extend_network_on_demand_platform.html [Accessed 2 September 2016].

AT&T, 2015b. *AT&T and Cisco Advance AT&T's Network on Demand Platform.* [Online] Available at:

http://about.att.com/story/cisco_advances_network_on_demand_platform.html [Accessed 2 September 2016].

AT&T, 2015c. *AT&T Expands Network on Demand Platform: Launches Managed Internet Service on Demand.* [Online] Available at: http://about.att.com/story/expands_network_demand_platform.html [Accessed 2 September 2016].

AT&T, 2015d. *AT&T Expands Network On Demand with Juniper Networks.* [Online] Available at: http://about.att.com/story/network_on_demand_juniper.html [Accessed 2 September 2016].

AT&T, 2015e. *AT&T Switched Ethernet Service Now Available in Six New Locations.* [Online] Available at: http://about.att.com/newsroom/att_switched_ethernet_service_now_available_in_6_new_locations.html [Accessed 2 September 2016].

AT&T, 2015f. *Continuing Our Software Transformation with Managed Internet on Demand.* [Online] Available at: http://about.att.com/innovationblog/09302015internet [Accessed 2 September 2016].

AT&T, 2016a. *AT&T Expands Flexible On-Demand Networking Capabilities.* [Online] Available at: http://about.att.com/story/att_expands_flexible_on_demand_networking_capabilities.html [Accessed 3 October 2016].

AT&T, 2016b. *AT&T Simplifies Networking, Helps Save Costs for Businesses in 76 Countries and Territories.* [Online] Available at: http://about.att.com/story/att_simplifies_networking_helps_save_costs_for_busine

sses.html
[Accessed 2 September 2016].

AT&T, n.d.-b. *AT&T Virtual Network Functions (FlexWare).* [Online] Available at: https://www.business.att.com/enterprise/Family/network-services/virtual-network-functions/
[Accessed 3 October 2016].

Avgerou, C., 2013. Social Mechanisms for Causal Explanation in Social Theory Based IS Research. *Journal of the Association for Information Systems,* August, 14(8), pp. 399-419.

Avital, M. & Te'eni, D., 2009. From Generative Fit To Generative Capacity: Exploring an Emerging Dimension Of Information Systems Design and Task Performance. *Information Systems Journal,* 19(4), pp. 345-367.

Baldwin, C. Y., 2008. Where do transactions come from? Modularity, transactions, and the boundaries of firms. *Industrial & Corporate Change,* 17(1), pp. 155-195.

Barbour, R. S., 2014. Quality of Data Analysis. In: U. Flick, ed. *The SAGE Handbook of Qualitative Data Analysis.* London: SAGE, pp. 496-509.

Bass, L., Clements, P. & Kazman, R., 2013. *Software Architecture in Practice.* 3rd ed. Upper Saddle River, NJ: Addison-Wesley.

Benkler, Y., 1998. Communications Infrastructure Regulation and the Distribution of Control Over Content. *Telecommunications Policy ,* 22(3), pp. 183-196.

Benkler, Y., 2000. From Consumers to Users: Shifting the Deeper Structures Of Regulation Toward Sustainable Commons And User Access. *Federal Communications Law Journal,* 05, 52(3), pp. 561-579.

Benton, T., 1998. Realism and Social Science: Some comments on Roy Bhaskar's 'The Possibility of Naturalism'. In: M. Archer, et al. eds. *Critical Realism: Essential Readings.* Routledge: Routledge, pp. 297-312.

Berry, D. M., 2011. *The Philosophy of Software: Code and Mediation in the Digital Age.* 1 ed. England: Palgrave Macmillan.

Bhaskar, R., 1998a. Philosophy and Scientific Realism. In: M. Archer, et al. eds. *Critical Realism: Essential Readings.* Routledge: Routledge, pp. 16-47.

Bhaskar, R., 1998b. The Logic of Scientific Discovery. In: M. Archer, et al. eds. *Critical Realism: Essential Readings.* Routledge: Routledge, pp. 48-103.

Big Switch Networks, n.d.. *Big Switch Networks: The Leader in Open Software Defined Networking.* [Online]
Available at: http://www.bigswitch.com/
[Accessed 2 September 2016].

Blanchette, J.-F., 2011. A Material History of Bits. *Journal of the American Society for Information Science & Technology,* 62(6), pp. 1042-1057.

Boucadair, M. & Jacquenet, C., 2014. *Software-Defined Networking: A Perspective from within a Service Provider Environment,* s.l.: s.n.

British Council, n.d. *English Grammar, Learn English, British Council, Perfect Aspect.* [Online]
Available at: https://learnenglish.britishcouncil.org/en/english-grammar/verbs/perfective-aspect
[Accessed 2 September 2016].

Brocade Communications Systems Inc., n.d.. *SDN Controller and Applications - Brocade.* [Online]
Available at: http://www.brocade.com/en/products-services/software-networking/sdn-controllers-applications.html
[Accessed 2 September 2016].

Bulmer, M., 2008. The Ethics of Social Research. In: N. Gilbert, ed. *Researching Social Life.* 3rd ed. London: SAGE, pp. 145-161.

Bygstad, B., 2010. Generative Mechanisms for Innovation in Information Infrastructures. *Information and Organization ,* 20(3-4), pp. 156-168.

Bygstad, B., Munkvold, B. E. & Volkoff, O., 2016. Identifying Generative Mechanisms. *Journal of Information Technology,* Mar, 31(1), pp. 83-96.

Carlsson, S. A., 2004. Using Critical Realism in IS Research. In: M. E. Whitman & A. B. Woszczynski, eds. *The Handbook of Information Systems Research.* London: Idea Group Publishing, pp. 323-338.

Carlsson, S. A., 2012. The Potential of Critical Realism in IS Research. In: Y. K. Dwivedi, M. R. Wade & S. L. Schneberger, eds. *Information Systems Theory: Explaining and Predicting Our Digital Society.* s.l.:Springer-Verlag New York, pp. 281-304.

Carpenter, B. & Brim, S., 2002. *Middleboxes: Taxonomy and Issues,* s.l.: RFC Editor.

Casado, M., 2007. *Architectural Support for Security Management in Enterprise Networks,* s.l.: s.n.

Casado, M. et al., 2007. Ethane: Taking Control of the Enterprise. *ACM SIGCOMM Computer Communication Review,* August, 37(4), pp. 1-12.

Casado, M. et al., 2009. Rethinking Enterprise Network Control. *Networking, IEEE/ACM Transactions on,* Aug, 17(4), pp. 1270-1283.

Casado, M., Garfinkel, T., Akella, A. & Freedman, M. J., 2006. *SANE: A Protection Architecture for Enterprise Networks.* Vancouver, BC, Canada, s.n., pp. 137-151.

Chikofsky, E. & Cross, J. I., 1990. Reverse Engineering and Design Recovery: A Taxonomy. *Software, IEEE,* Jan, 7(1), pp. 13-17.

Chowdhury, N. M. K. & Boutaba, R., 2010. A Survey of Network Virtualization. *Computer Networks ,* 54(5), pp. 862-876.

Chun, B. et al., 2003. PlanetLab: An Overlay Testbed for Broad-coverage Services. *ACM SIGCOMM Computer Communication Review,* July, 33(3), pp. 3-12.

Ciborra, C., ed., 2000. *From Control to Drift: The Dynamics of Corporate Information Infastructures.* Oxford: Oxford University Press.

Cisco Systems, 2013a. *Cisco Completes Acquisition of Insieme Networks.* [Online] Available at: https://newsroom.cisco.com/press-release-content?articleId=1298696 [Accessed 2 September 2016].

Cisco Systems, 2013b. *Cisco Pioneers Real-Time Application Delivery in Global Data Centers and Clouds to Enable Greater Business Agility.* [Online] Available at: https://newsroom.cisco.com/press-release-content?articleId=1281713 [Accessed 2 September 2016].

Cisco Systems, 2013c. *Software-Defined Networking: Why We Like It and How We Are Building On It,* s.l.: Cisco Systems.

Cisco Systems, 2014a. *Cisco Announces Intent to Acquire Tail-f Systems.* [Online] Available at: https://newsroom.cisco.com/press-release-content?articleId=1438152 [Accessed 2 September 2016].

Cisco Systems, 2014b. *Cisco Completes Acquisition of Tail-f Systems.* [Online] Available at: https://newsroom.cisco.com/press-release-content?type=webcontent&articleId=1451764 [Accessed 2 September 2016].

Cisco Systems, 2014c. *Introducing OpFlex – A new standards-based protocol for Application Centric Infrastructure.* [Online] Available at: http://blogs.cisco.com/datacenter/introducing-opflex-a-new-standards-based-protocol-for-application-centric-infrastructure [Accessed 2 September 2016].

Cisco Systems, 2014d. *Is Cisco Application Centric Infrastructure an SDN Technology?.* [Online] Available at: http://www.cisco.com/c/en/us/solutions/collateral/data-center-virtualization/application-centric-infrastructure/white-paper-c11-733456.html [Accessed 2 September 2016].

Cisco Systems, 2016. *Cisco Reports Fourth Quarter and Fiscal Year 2016 Earnings.* [Online]
Available at: http://investor.cisco.com/investor-relations/news-and-events/news/news-details/2016/Cisco-Reports-Fourth-Quarter-and-Fiscal-Year-2016-Earnings/
[Accessed 2 September 2016].

Cisco Systems, n.d.-a. *Certifications - Training and Certifications - Cisco.* [Online]
Available at: http://www.cisco.com/c/en/us/training-events/training-certifications/certifications.html
[Accessed 2 September 2016].

Cisco Systems, n.d.-b. *Cisco Firepower 9000 Series.* [Online]
Available at: http://www.cisco.com/c/en/us/products/security/firepower-9000-series/index.html
[Accessed 2 September 2016].

Coffey, A., 2014. Analysing Documents. In: U. Flick, ed. *The SAGE Handbook of Qualitative Data Analysis.* London: SAGE, pp. 297-312.

Cohen, L., Manion, L. & Morrison, K., 2011. *Research Methods in Education.* 7th ed. Oxford: Routledge.

Collier, A., 1994. *Critical Realism: An Introduction to Roy Bhaskar's Philosophy.* London: Verso.

Corley, K. G. & Gioia, D. A., 2011. Building Theory About Theory Building: What Constitutes A Theoretical Contribution?. *Academy of Management Review,* 36(1), pp. 12-32.

Crawley, E., Cameron, B. & Selva, D., 2016. *System Architecture: Strategy and Product Development for Complex Systems.* Sussex, England: Pearson Education Limited.

Crotty, M., 1998. *The Foundations of Social Research: Meaning and Perspective in the Research Process.* London: SAGE.

Current Analysis Inc., 2016. *Company Assessment of NTT Communications – Global Enterprise.* [Online] Available at: http://www.ntt.com/content/dam/nttcom/affiliate/cmn/pdf/resouces/analysis/current_analysis_company_assessment_2016.pdf
[Accessed 2 September 2016].

Deering, S. E. & Hinden, R. M., 1998. *Internet Protocol, Version 6 (IPv6) Specification,* s.l.: RFC Editor.

DiMaggio, P. J., 1995. Comments on "What Theory is Not". *Administrative Science Quarterly,* Volume 3, pp. 391-397.

Doria, A. et al., 2010. *Forwarding and Control Element Separation (ForCES) Protocol Specification.* [Online] Available at: http://www.rfc-editor.org/rfc/rfc5810.txt
[Accessed 2 September 2016].

Easton, G., 2010. Critical Realism in Case Study Research. *Industrial Marketing Management ,* 39(1), pp. 118-128.

Edge-Core Networks, n.d.. *Edge-Core Networks.* [Online] Available at: http://www.edge-core.com/
[Accessed 2016 September 2].

Edwards, P. N., Jackson, S. J., Bowker, G. C. & Knobel, C. P., 2007. *Understanding Infrastructure: Dynamics, Tensions, and Design: Report of a Workshop on History & Theory of Infrastructure, Lessons for New Scientific Cyberinfrastructures,* s.l.: NSF.

Egyedi, T. & Spirco, J., 2011. Standards in Transitions: Catalyzing Infrastructure Change. *Futures,* 43(9), pp. 947-960.

Eisenhardt, K. M., 1989. Building Theories from Case Study Research. *Academy of Management Review,* 14(4), pp. 532-550.

Elder-Vass, D., 2007. For Emergence: Refining Archer's Account of Social Structure.. *Journal for the Theory of Social Behaviour,* 37(1), pp. 25-44.

Engholm, P., 2007. TMSA (transformational model of social activity). In: M. Hartwig, ed. *Dictionary of Critical Realism.* London: Routledge, pp. 467-470.

Eriksson, O. & Ågerfalk, P. J., 2010. Rethinking the Meaning of Identifiers in Information Infrastructures. *Journal of the Association for Information Systems,* August, 11(8), pp. 433-454.

Esterberg, K. G., 2002. *Qualitative Methods in Social Research.* Boston, USA: McGraw-Hill.

European Commission, 2015. *Digital Single Market, Our commitment to Net Neutrality.* [Online]
Available at: https://ec.europa.eu/digital-single-market/en/eu-actions-net-neutrality#Article
[Accessed 2 September 2016].

European Parliament, Council of the European Union, 2015. Regulation (EU) 2015/2120 of the European Parliament and of the Council of 25 November 2015 laying down measures concerning open internet access and amending Directive 2002/22/EC on universal service and users' rights relating to electronic communications networks and services and Regulation (EU) No 531/2012 on roaming on public mobile communications networks within the Union (Text with EEA relevance). *Official Journal of the European Union,* November, OJ-L(310), pp. 1-18.

Faraj, S. & Azad, B., 2012. The Materiality of Technology: An Affordance Perspective. In: B. A. N. J. K. Paul M. Leonardi, ed. *Materiality and Organizing: Social Interaction in a Technological World.* Oxford: Oxford University Press, pp. 237-258.

Faulkner, P. & Runde, J., 2009. On the Identity of Technological Objects and User Innovations in Function. *Academy of Management Review,* 34(3), pp. 442-462.

Faulkner, P. & Runde, J., 2013. Technological Objects, Social Positions, and The Transformational Model Of Social Activity. *MIS Quarterly,* 37(3), pp. 803-818.

Feamster, N., Rexford, J. & Zegura, E., 2014. The Road to SDN: An Intellectual History of Programmable Networks. *ACM SIGCOMM Computer Communication Review,* April, 44(2), pp. 87-98.

Federal Communications Commission, 2015a. *Open Internet.* [Online] Available at: https://www.fcc.gov/general/open-internet [Accessed 2 September 2016].

Federal Communications Commission, 2015. *Open Internet Report and Order on Remand, Declaratory Ruling, and Order, FCC-15-24.* [Online] Available at: https://www.fcc.gov/document/fcc-releases-open-internet-order [Accessed 2 September 2016].

Fielding, N. & Thomas, H., 2008. Qualitative Interviewing. In: G. N. Gilbert, ed. *Researching Social Life.* 3rd ed. London: SAGE, pp. 245-265.

Fielding, R. T. et al., 1999. *Hypertext Transfer Protocol -- HTTP/1.1,* s.l.: RFC Editor.

FIRE STUDY Team, 2016. *The European Open Ecosystem for Future Internet Experimentation & Innovation,* s.l.: s.n.

FIRE, n.d.. *FIRE.* [Online] Available at: https://www.ict-fire.eu/ [Accessed 2 September 2016].

Firestone, W. A., 1993. Alternative Arguments for Generalizing from Data as Applied to Qualitative Research. *Educational Researcher,* 22(4), pp. 16-23.

Fisher, C. B. & Anushko, A. E., 2008. Research Ethics in Social Science. In: P. Alasuutari, L. Bickman & J. Brannen, eds. *The SAGE Handbook of Social Research Methods.* London: SAGE, pp. 95-109.

Fleetwood, S., 2009. The Ontology of Things, Properties and Powers. *Journal of Critical Realism,* 8(3), pp. 343-366.

Fleetwood, S., 2011. Powers and Tendencies Revisited. *Journal of Critical Realism,* 10(1), pp. 80-99.

Flick, U., 2014a. *An Introduction to Qualitative Research.* 5th ed. London: SAGE.

Flick, U., 2014b. Mapping the Field. In: U. Flick, ed. *The SAGE Handbook of Qualitative Data Analysis.* London: SAGE, pp. 3-18.

Gartner Inc., 2015a. *Gartner Says 6.4 Billion Connected "Things" Will Be in Use in 2016, Up 30 Percent From 2015.* [Online] Available at: http://www.gartner.com/newsroom/id/3165317 [Accessed 2 September 2016].

Gartner Inc., 2015b. *NTT Communications' SDN and NFV Deployment Redefines Network Value,* s.l.: Gartner Inc..

Gartner, 2016. *Magic Quadrant for Data Center Networking.* [Online] Available at: https://www.gartner.com/doc/reprints?id=1-36UFTCZ&ct=160517 [Accessed 2 September 2016].

Gaskell, G., 2000. Individual and Group Interviewing. In: M. W. Bauer & G. Gaskell, eds. *Qualitative Researching with Text Image and Sound: A Practical Handbook.* London: SAGE, pp. 38-56.

GENI, n.d.. *GENI.* [Online] Available at: http://www.geni.net/ [Accessed 2 September 2016].

Gibbs, G. R., 2007. *Analyzing Qualitative Data.* London: SAGE.

Gobo, G., 2008. Re-conceptualizing Generalization: Old Issues in a New Frame. In: P. Alasuutari, L. Bickman & J. Brannen, eds. *The SAGE Handbook of Social Research Methods.* 1 ed. s.l.:London : SAGE, pp. 193-213.

Greenberg, Albert, 2015. *Microsoft Showcases Software Defined Networking Innovation at SIGCOMM.* [Online] Available at: https://azure.microsoft.com/en-gb/blog/microsoft-showcases-software-defined-networking-innovation-at-sigcomm-v2/ [Accessed 2 September 2016].

Greenberg, A. et al., 2005. A Clean Slate 4D Approach to Network Control and Management. *ACM SIGCOMM Computer Communication Review,* October, 35(5), pp. 41-54.

Green, N., 2008. Formulating and Refining a Research Question. In: G. N. Gilbert, ed. *Researching Social Life.* 3rd ed. London: SAGE, pp. 43-62.

Gregor, S., 2006. The Nature Of Theory In Information Systems.. *MIS Quarterly,* 30(3), pp. 611-642.

Grisot, M., Hanseth, O. & Thorseng, A. A., 2014. Innovation Of, In, On Infrastructures: Articulating the Role of Architecture in Information Infrastructure Evolution. *Journal of the Association for Information Systems,* 15(4), pp. 197-219.

Gude, N. et al., 2008. NOX: Towards an Operating System for Networks. *ACM SIGCOMM Computer Communication Review,* July, 38(3), pp. 105-110.

Haleplidis, E. et al., 2015. *Software-Defined Networking (SDN): Layers and Architecture Terminology,* s.l.: Internet Research Task Force (IRTF).

Halpern, J. & Salim, J. H., 2010. *Forwarding and Control Element Separation (ForCES) Forwarding Element Model,* s.l.: RFC Editor.

Hanseth, O., 2000. The Economics of Standards. In: C. Ciborra, ed. *From Control to Drift: The Dynamics of Corporate Information Infastructures.* Oxford: Oxford University Press, pp. 56-70.

Hanseth, O., 2001. Gateways - Just as Important as Standards: How the Internet Won the "Religious War" over Standards in Scandinavia. *Knowledge, Technology and Policy,* 14(3), pp. 71-89.

Hanseth, O. & Braa, K., 2000. Who's in Control: Designers, Managers - or Technology? Infrastructures at Norsk Hydro. In: C. Ciborra, ed. *From Control to Drift: The Dynamics of Corporate Information Infastructures.* Oxford: Oxford University Press, pp. 125-147.

Hanseth, O. & Lundberg, N., 2001. Designing Work Oriented Infrastructures. *Computer Supported Cooperative Work (CSCW),* 10(3-4), pp. 347-372.

Hanseth, O. & Lyytinen, K., 2010. Design Theory for Dynamic Complexity in Information Infrastructures: The Case of Building Internet. *Journal of Information Technology,* 25(1), pp. 1-19.

Hanseth, O. & Monteiro, E., 1997. Inscribing Behaviour in Information Infrastructure Standards. *Accounting, Management and Information Technologies ,* 7(4), pp. 183-211.

Hanseth, O., Monteiro, E. & Hatling, M., 1996. Developing Information Infrastructure: The Tension between Standardization and Flexibility. *Science, Technology, & Human Values,* 21(4), pp. pp. 407-426.

Harman, M., 2010. *Why Source Code Analysis and Manipulation Will Always Be Important.* Washington, DC, USA, IEEE Computer Society, pp. 7-19.

Hartwig, M., ed., 2007. *Dictionary of Critical Realism.* London: Routledge.

Hauger, D., 2010. *Windows Azure General Availability.* [Online] Available at: https://blogs.technet.microsoft.com/microsoft_blog/2010/02/01/windows-azure-general-availability/
[Accessed 2 September 2016].

Henderson, R. M. & Clark, K. B., 1990. Architectural Innovation: The Reconfiguration of Existing Product Technologies and the Failure of Established Firms. *Administrative Science Quarterly,* 35(1), pp. 9-30.

Henfridsson, O. & Bygstad, B., 2013. The Generative Mechanisms Of Digital Infrastructure Evolution. *MIS Quarterly,* 37(3), pp. 907 - A5.

Henfridsson, O., Mathiassen, L. & Svahn, F., 2014. Managing Technological Change in the Digital Age: The Role of Architectural Frames. *Journal of Information Technology,* Mar, 29(1), pp. 27-43.

Housley, R., Crocker, D. & Burger, E., 2011. *Reducing the Standards Track to Two Maturity Levels,* s.l.: RFC Editor.

Iannacci, F., 2010. When is an information infrastructure? Investigating the Emergence of Public Sector Information Infrastructures. *European Journal of Information Systems,* February, 19(1), pp. 35-48.

IEEE SDN, n.d.. *IEEE Software Defined Networks.* [Online] Available at: http://sdn.ieee.org/
[Accessed 2 September 2016].

International Data Corporation, 2014. *IDC MarketScape: Asia/Pacific Next-Generation Telcos Datacenter and Hosted Cloud Services 2013-2014 Vendor Analysis (NTT Communications Excerpt).* [Online] Available at: http://www.ntt.com/content/dam/nttcom/mig2/aboutus_e/news/data/pdf/20131218.pdf
[Accessed 2 September 2016].

International Data Corporation, 2015. *Mobile Internet Users to Top 2 Billion Worldwide in 2016, According to IDC.* [Online] Available at: https://www.idc.com/getdoc.jsp?containerId=prUS40855515
[Accessed 2 September 2016].

Internet Engineering Task Force, 2016. *Official Internet Protocol Standards.* [Online] Available at: http://www.rfc-editor.org/standards/
[Accessed 2 September 2016].

ITU-T, 2014. *Framework of software-defined networking,* s.l.: s.n.

Jain, S. et al., 2013. B4: Experience with a Globally-Deployed Software Defined WAN. *ACM SIGCOMM Computer Communication Review,* August, 43(4), pp. 3-14.

Jarraya, Y., Madi, T. & Debbabi, M., 2014. A Survey and a Layered Taxonomy of Software-Defined Networking. *IEEE Communications Surveys Tutorials,* Fourthquarter, 16(4), pp. 1955-1980.

Juniper Networks Inc., 2012. *Juniper Networks to Acquire Contrail Systems.* [Online] Available at: http://forums.juniper.net/t5/The-New-Network/Juniper-Networks-to-

Acquire-Contrail-Systems/ba-p/170940
[Accessed 2 September 2016].

Juniper Networks Inc., 2015. *Juniper Networks Contrail Selected to Support the AT&T Integrated Cloud.* [Online]
Available at: http://newsroom.juniper.net/press-release/juniper-networks-contrail-selected-support-att-integrated-cloud
[Accessed 2 September 2016].

Juniper Networks Inc., n.d.. *Juniper Networks Certification Program - JNCP - Juniper Networks.* [Online]
Available at: http://www.juniper.net/us/en/training/certification/
[Accessed 2 September 2016].

Kallinikos, J., 2004. Farewell to Constructivism: Technology and Context-Embedded Action. In: C. Avgerou, C. Ciborra & F. Land, eds. *The Social Study of Information and Communication Technology: Innovation, Actors and Contexts.* Oxford: Oxford University Press.

Kallinikos, J., 2011a. Control and Complexity in a Connected World. In: *Governing Through Technology: Information Artefacts and Social Practice.* Basingstoke: Palgrave Macmillan, pp. 72-90.

Kallinikos, J., 2011b. *Governing Through Technology: Information Artefacts and Social Practice.* Basingstoke: Palgrave Macmillan.

Kallinikos, J., 2012. *Form, Function, and Matter: Crossing the Border of Materiality.* Oxford: Oxford University Press.

Kallinikos, J., Aaltonen, A. & Marton, A., 2010. A Theory of Digital Objects. *First Monday,* 15(6).

Kallinikos, J., Aaltonen, A. & Marton, A., 2013. The Ambivalent Ontology of Digital Artifacts. *MIS Quarterly,* 37(2), pp. 357-370.

Kallinikos, J., Leonardi, P. M. & Nardi, B. A., 2012. The Challenge of Materiality: Origins, Scope, and Prospects. In: *Materiality and Organizing.* Oxford: Oxford University Press, pp. 3-22.

Kang, J. M., Bannazadeh, H. & Leon-Garcia, A., 2013. *SAVI Testbed: Control and Management of Converged Virtual ICT Resources.* s.l., s.n., pp. 664-667.

Katz, M. L. & Shapiro, C., 1985. Network Externalities, Competition, and Compatibility. *American Economic Review,* 75(3), p. 424.

Katz, M. L. & Shapiro, C., 1986. Technology Adoption in the Presence of Network Externalities. *Journal of Political Economy,* 94(4), pp. 822-841.

Katz, M. L. & Shapiro, C., 1994. Systems Competition and Network Effects. *Journal of Economic Perspectives,* 8(2), pp. 93-115.

Kautz, K. & Jensen, T. B., 2013. Sociomateriality at the Royal Court of IS: A Jester's Monologue. *Information and Organization,* 23(1), pp. 15-27.

Kelle, U., 2014. Theorization from Data. In: U. Flick, ed. *The SAGE Handbook of Qualitative Data Analysis.* s.l.:London: SAGE, pp. 554-568.

Kennedy, M. M., 1979. Generalizing From Single Case Studies. *Evaluation Review,* 3(4), pp. 661-678.

Kirschenbaum, M. G., 2008. *Mechanisms: New Media and the Forensic Imagination.* s.l.:MIT Press.

Kreutz, D. et al., 2015. Software-Defined Networking: A Comprehensive Survey. *Proceedings of the IEEE,* Jan, 103(1), pp. 14-76.

Kurose, J. F. & Ross, K. W., 2013. *Computer Networking: A Top-Down Approach: International Edition.* 6 ed. s.l.:Pearson Higher Education.

Kvale, S., 2007. *Doing Interviews.* London: SAGE.

Lammle, T., 2011. *CCNA Cisco Certified Network Associate Study Guide.* 7 ed. s.l.:Wiley Publishing.

Langlois, R. N. & Robertson, P. L., 1992. Networks and Innovation in a Modular System: Lessons from the Microcomputer and Stereo Component Industries. *Research Policy,* 21(4), pp. 297-313.

Leonardi, P., 2010. Digital Materiality? How Artifacts Without Matter, Matter. *First Monday,* 15(6).

Leonardi, P. M., 2011. When Flexible Routines Meet Flexible Technologies: Affordance, Constraint, And The Imbrication Of Human And Material Agencies.. *MIS Quarterly,* 35(1), pp. 147-168.

Leonardi, P. M., 2012. Materiality, Sociomateriality, and Socio-Technical Systems: What Do These Terms Mean? How Are They Different? Do We Need Them?. In: B. A. N. J. K. Paul M. Leonardi, ed. *Materiality and Organizing: Social Interaction in a Technological World.* Oxford: Oxford University Press, pp. 67-87.

Leonardi, P. M., 2013. Theoretical Foundations for the Study of Sociomateriality. *Information and Organization,* 23(2), pp. 59-76.

Leonardi, P. M. & Barley, S. R., 2008. Materiality and Change: Challenges to Building better Theory about Technology and Organizing. *Information and Organization ,* 18(3), pp. 159-176.

Lessig, L., 2001. *The Future of Ideas: The Fate of the Commond in a Connected World.* s.l.:Random House.

Lyytinen, K. & Rose, G. M., 2003. The Disruptive Nature of Information Technology Innovations: The Case of Internet Computing in Systems Development Organizations. *MIS Quarterly,* 27(4), pp. pp. 557-596.

Mabry, L., 2008. Case Study in Social Research. In: P. Alasuutari, L. Bickman & J. Brannen, eds. *The SAGE Handbook of Social Research Methods.* London: Sage, pp. 214-227.

Macdonald, K., 2008. Using Documents. In: G. N. Gilbert, ed. *Researching Social Life.* 3rd ed. London: SAGE, pp. 285-303.

Maciaszek, L. A., Liong, L. B. & Bills, S., 2005. *Practical Software Engineering: A Case Study Approach.* Harlow, Essex: Pearson/Addison Wesley.

Maier, M. W. & Rechtin, E., 2002. *The Art of Systems Architecting.* 2nd ed. Boca Raton ; London: CRC Press.

Manovich, L., 2002. *The Language of New Media.* Cambridge, Mass.: MIT Press.

Markus, M. L. & Robey, D., 1988. Information Technology and Organizational Change: Causal Structure in Theory and Research. *Management Science,* 34(5), pp. 583-598.

Marzano, M., 2012. Informed Consent. In: J. F. Gubrium, J. A. Holstein, A. B. Marvasti & K. D. McKinney, eds. *The SAGE Handbook of Interview Research: The Complexity of the Craft.* 2nd ed. Thousand Oaks, California: SAGE, pp. 443-456.

Maxwell, J. A. & Chmiel, M., 2014. Generalization in and from Qualitative Analysis. In: U. Flick, ed. *The SAGE Handbook of Qualitative Data Analysis.* London: SAGE, pp. 540-553.

May, T. & Perry, B., 2011. Case Study Research. In: *Social Research: Issues, Methods and Process.* 4th ed. Maidenhead: Open University Press, pp. 219-242.

McCulloch, G., 2011. Historical and Documentary Research in Education. In: L. Cohen, L. Manion & K. Morrison, eds. *Research Methods in Education.* 7th ed. Oxford: Routledge, pp. 248-255.

McKeown, N. et al., 2008. OpenFlow: Enabling Innovation in Campus Networks. *ACM SIGCOMM Computer Communication Review,* March, 38(2), pp. 69-74.

Medina, V. & García, J. M., 2014. A Survey of Migration Mechanisms of Virtual Machines. *ACM Computing Surveys,* January, 46(3), pp. 1-33.

Mens, T. & Tourwe, T., 2004. A Survey of Software Refactoring. *Software Engineering, IEEE Transactions on,* Feb, 30(2), pp. 126-139.

Mertens, D. M., 2014. Ethical Use of Qualitative Data Findings. In: U. Flick, ed. *The SAGE Handbook of Qualitative Data Analysis.* London: SAGE, pp. 510-523.

Metro Ethernet Forum, 2014a. *Carrier Ethernet and SDN Part 1: An Industry Perspective,* s.l.: s.n.

Metro Ethernet Forum, 2014b. *Carrier Ethernet and SDN Part 2: Practical Considerations,* s.l.: s.n.

Metro Ethernet Forum, 2015. *The Third Network: Lifecycle Service Orchestration Vision,* s.l.: s.n.

Microsoft, n.d.. *API Management: Establish Gateways - Microsoft Azure.* [Online] Available at: https://azure.microsoft.com/en-gb/services/api-management/ [Accessed 2 September 2016].

Mingers, J., 2004a. Real-izing Information Systems: Critical Realism as an Underpinning Philosophy for Information Systems. *Information and Organization,* 14(2), pp. 87-103.

Mingers, J., 2004b. Re-establishing the Real: Critical Realism and Information Systems. In: J. Mingers & L. Willcocks, eds. *Social Theory and Philosophy for Information Systems.* Chichester: John Wiley & Sons, pp. 372-406.

Mojatatu, n.d.. *Products - Mojatatu.* [Online] Available at: http://www.mojatatu.info/products [Accessed 2 September 2016].

Monteiro, E., 1998. Scaling Information Infrastructure: The Case of Next-Generation IP in the Internet. *Information Society,* 14(3), pp. 229-245.

Monteiro, E., Pollock, N., Hanseth, O. & Williams, R., 2013. From Artefacts to Infrastructures. *Computer Supported Cooperative Work (CSCW),* 22(4-6), pp. 575-607.

Morgan, J., 2007a. Emergence. In: M. Hartwig, ed. *Dictionary of Critical Realism.* London: Routledge, pp. 166-167.

Mutch, A., 2002. Actors and Networks or Agents and Structures: Towards a Realist View of Information Systems. *Organization,* August, 9(3), pp. 477-496.

Mutch, A., 2013. Sociomateriality: Taking the wrong turning?. *Information and Organization,* 23(1), pp. 28-40.

Naous, J. et al., 2008. *Implementing an OpenFlow Switch on the NetFPGA Platform.* New York, NY, USA, ACM, pp. 1-9.

National Science Foundation, TRUST, n.d. *Award 0424422 - Team for Research in Ubiquitous Secure Technology (TRUST).* [Online] Available at: http://www.nsf.gov/awardsearch/showAward?AWD_ID=0424422

Nicira Networks Inc., 2012a. *It's Time to Virtualize the Network: Network Virtualization for Cloud Data Centers,* s.l.: s.n.

Nicira Networks Inc., 2012b. *Networking in the Era of Virtualization,* s.l.: s.n.

Nicira Networks Inc., 2012c. *Nicira Network Virtualization Platform (NVP),* s.l.: s.n.

Nicira Networks Inc., 2012d. *The Seven Properties of Network Virtualization,* s.l.: s.n.

Nielsen, P. & Aanestad, M., 2006. Control Devolution as Information Infrastructure Design Strategy: A Case Study of a Content Service Platform for Mobile Phones in Norway. *Journal of Information Technology,* September, 21(3), pp. 185-194.

Nokia Corporation, 2016. *Nokia Gains Control of Alcatel-Lucent Through Successful Public Exchange Offer; Nokia To Hold Nearly 80% Of Outstanding Alcatel-Lucent Shares.* [Online] Available at: http://company.nokia.com/en/news/press-releases/2016/01/04/nokia-gains-control-of-alcatel-lucent-through-successful-public-exchange-offer-nokia-to-hold-nearly-80-of-outstanding-alcatel-lucent-shares
[Accessed 2 September 2016].

Norris, C., 2007. Transcendental Realism. In: M. Hartwig, ed. *Dictionary of Critical Realism.* London: Routledge, pp. 474-481.

NSERC, 2011. *NSERC Investments Information and Communications Technologies,* s.l.: s.n.

NSERC, n.d. *NSERC Strategic Research Network for Smart Applications on Virtual Infrastructures.* Canada: s.n.

NTT Communications, 2015. *NTT Communications Introduces Arcstar Universal One with SDN for Advanced Cloud-based VPN Service.* [Online] Available at: http://www.ntt.com/en/about-us/press-releases/news/article/2015/20150514.html
[Accessed 2 September 2016].

NTT Communications, n.d.. *Enterprise Cloud - NTT Communications Global ICT Services Provider.* [Online]
Available at: http://www.ntt.com/en/services/cloud/enterprise-cloud.html
[Accessed 2 September 2016].

Nuage Networks, n.d.. *Software Defined Networking: Nuage Networks.* [Online] Available at: www.nuagenetworks.net/
[Accessed 2 September 2016].

Nunes, B. A. A. et al., 2014. A Survey of Software-Defined Networking: Past, Present, and Future of Programmable Networks. *IEEE Communications Surveys Tutorials,* Third, 16(3), pp. 1617-1634.

Open Compute Project, n.d.. *About The Open Compute Project.* [Online] Available at: http://www.opencompute.org/about/

Open Networking Foundation, 2009. *OpenFlow Switch Specification: Version 1.0.0 (Wire Protocol 0x01).* [Online]
Available at: https://www.opennetworking.org/images/stories/downloads/sdn-resources/onf-specifications/openflow/openflow-spec-v1.0.0.pdf
[Accessed 2 September 2016].

Open Networking Foundation, 2012a. *OpenFlow Switch Errata: Version1.0.1.* [Online] Available at: https://www.opennetworking.org/images/stories/downloads/sdn-

resources/onf-specifications/openflow/openflow-spec-v1.0.1.pdf
[Accessed 2 September 2016].

Open Networking Foundation, 2012b. *OpenFlow Switch Specification: Version 1.3.1 (Wire Protocol 0x04).* [Online]
Available at: https://www.opennetworking.org/images/stories/downloads/sdn-resources/onf-specifications/openflow/openflow-spec-v1.3.1.pdf
[Accessed 2 September 2016].

Open Networking Foundation, 2015. *OpenFlow Switch Specification: Version 1.5.1 (Protocol version 0x06).* [Online]
Available at: https://www.opennetworking.org/images/stories/downloads/sdn-resources/onf-specifications/openflow/openflow-switch-v1.5.1.pdf
[Accessed 2 September 2016].

Open Networking Foundation, n.d.-a. *Code Resources, Open Networking Foundation.* [Online]
Available at: https://www.opennetworking.org/sdn-resources/code-resources
[Accessed 2 September 2016].

Open Networking Foundation, n.d.-b. *Open Networking Foundation.* [Online]
Available at: https://www.opennetworking.org/

Open Networking Foundation, n.d.-c. *OpenFlow, Open Networking Foundation.* [Online]
Available at: https://www.opennetworking.org/sdn-resources/openflow/

Open Networking Foundation, n.d.-d. *Products Listing, Open Networking Foundation.* [Online]
Available at: https://www.opennetworking.org/products-listing/
[Accessed 2 September 2016].

Open Networking Foundation, n.d.-e. *Technical Library, Open Networking Foundation.* [Online]
Available at: https://www.opennetworking.org/sdn-resources/technical-library

Open Source SDN, n.d. *Open Source SDN.* [Online] Available at: http://opensourcesdn.org/

Open vSwitch, n.d.. *Open vSwitch.* [Online] Available at: http://openvswitch.org/ [Accessed 2 September 2016].

OpenDaylight Foundation, n.d.. *The OpenDaylight Platform.* [Online] Available at: https://www.opendaylight.org/ [Accessed 2 September 2016].

OpenFlow Consortium, n.d.. *OpenFlow - Enabling Innovation in Your Network.* [Online] Available at: http://archive.openflow.org/ [Accessed 2 September 2016].

Orlikowski, W. J., 2007. Sociomaterial Practices: Exploring Technology at Work. *Organization Studies,* 28(9), pp. 1435-1448.

Orlikowski, W. J. & Scott, S. V., 2008. Sociomateriality: Challenging the Separation of Technology, Work and Organization.. *Academy of Management Annals,* 2(1), pp. 433-474.

Outhwaite, W., 1998. Realism and Social Science. In: M. Archer, et al. eds. *Critical Realism: Essential Readings.* Routledge: Routledge, pp. 282-296.

Parnas, D. L., 1972. On the Criteria to be Used in Decomposing Systems into Modules. *Communications of the ACM,* December, Volume 15, pp. 1053-1058.

Pearce, M., Zeadally, S. & Hunt, R., 2013. Virtualization: Issues, Security Threats, and Solutions. *ACM Computer Surveys,* March, 45(2), pp. 1-39.

Pentland, B. T. & Singh, H., 2012. Materiality: What are the Consequences?. In: B. A. N. J. K. Paul M. Leonardi, ed. *Materiality and Organizing: Social Interaction in a Technological World.* Oxford: Oxford University Press, pp. 288-295.

Peterson, L. et al., 2006. GENI Design Principles. *Computer,* September, 39(9), pp. 102-105.

Pinkstone, B. & Hartwig, M., 2007. Tendency. In: M. Hartwig, ed. *Dictionary of Critical Realism.* London: Routledge, pp. 458-460.

Polit, D. F. & Beck, C. T., 2010. Generalization in Quantitative and Qualitative Research: Myths and Strategies. *International Journal of Nursing Studies,* 47(11), pp. 1451-1458.

Porpora, D. V., 1989. Four Concepts of Social Structure. *Journal for the Theory of Social Behaviour,* 19(2), pp. 195-211.

Postel, J., 1981. *Internet Protocol,* s.l.: RFC Editor.

Postel, J., 1981. *Transmission Control Protocol,* s.l.: RFC Editor.

Postel, J. & Reynolds, J., 1985. *File Transfer Protocol,* s.l.: RFC Editor.

Pratten, S. B., 2007. Explanation. In: M. Hartwig, ed. *Dictionary of Critical Realism.* London: Routledge, pp. 193-196.

Psillos, S., 2007. Causal Law. In: M. Hartwig, ed. *Dictionary of Critical Realism.* London: Routledge, pp. 60-61.

Quanta Cloud Technolgy, n.d.. *White Box Bare Metal Switches Featuring SDN, ONIE, ONL Compliant For Cloud Datacenters.* [Online] Available at: http://www.qct.io/Product/Networking/Bare-Metal-Switch-c77c75c159

Radisys Corporation, 2016. *Radisys Names First Commercial Customer Using FlowEngine Intelligent Traffic Distribution System.* [Online] Available at: http://www.radisys.com/press-releases/radisys-names-verizon-flowengine-customer/
[Accessed 2 September 2016].

Rapley, T., 2014. Sampling Strategies in Qualitative Research. In: U. Flick, ed. *The SAGE Handbook of Qualitative Data Analysis.* London: SAGE, pp. 49-63.

Reichertz, J., 2014. Induction, Deduction, Abduction. In: U. Flick, ed. *The SAGE Handbook of Qualitative Data Analysis.* s.l.:London: SAGE, pp. 123-135.

Reuters, 2016. *Cisco to Cut 5,500 Jobs in Shift from Switches to Software.* [Online] Available at: http://www.reuters.com/article/us-cisco-systems-results-idUSKCN10S26P
[Accessed 2 September 2016].

Rexford, J. et al., 2004. *Network-Wide Decision Making: Toward A Wafer-Thin Control Plane.* s.l., s.n.

Ribes, D. & Finholt, T. A., 2009. The Long Now of Technology Infrastructure: Articulating Tensions in Development. *Journal of the Association for Information Systems,* 10(5), pp. 375-398.

Richards, L., 2005. *Handling Qualitative Data: A Practical Guide.* Thousand Oaks, CA: SAGE.

Robson, C., 2011. *Real World Research: A Resource for Users of Social Research Methods in Applied Settings.* 3rd ed. Chichester: Wiley.

Rodon, J. & Silva, L., 2015. Exploring the Formation of a Healthcare Information Infrastructure: Hierarchy or Meshwork?. *Journal of the Association for Information Systems,* 16(5), pp. 394-417.

Rose, J., Lindgren, R. & Henfridsson, O., 2004. Socio-Technical Structure: An Experiment in Integrative Theory Building. In: B. Kaplan, et al. eds. *Information Systems Research: Relevant Theory and Informed Practice.* Boston, MA: Springer, pp. 411-432.

Rosenblum, M. & Garfinkel, T., 2005. Virtual Machine Monitors: Current Technology and Future Trends. *IEEE Computer,* May, 38(5), pp. 39-47.

Runeson, P. & Höst, M., 2008. Guidelines for Conducting and Reporting Case Study Research in Software Engineering. *Empirical Software Engineering,* 14(2), pp. 131-164.

Russinovich, M., 2015. *Report from Open Networking Summit: Achieving Hyper-Scale with Software Defined Networking.* [Online] Available at: https://azure.microsoft.com/en-us/blog/report-from-open-networking-

summit-achieving-hyper-scale-with-software-defined-networking/
[Accessed 2 September 2016].

Sahay, S., Aanestad, M. & Monteiro, E., 2009. Configurable Politics and Asymmetric Integration: Health e-Infrastructures in India. *Journal of the Association for Information Systems,* 10(5), pp. 399-414.

Saldaña, J., 2013. *The Coding Manual for Qualitative Researchers.* 2nd ed. Los Angeles: Sage.

Saltzer, J. H., Reed, D. P. & Clark, D. D., 1984. End-to-end Arguments in System Design. *ACM Transactions on Computer Systems,* November, 2(4), pp. 277-288.

Sanchez, R. & Mahoney, J. T., 1996. Modularity, Flexibility, And Knowledge Management In Product And Organization Design. *Strategic Management Journal,* Volume 17, pp. 63-76.

Sanner, T. A., Manda, T. D. & Nielsen, P., 2014. Grafting: Balancing Control and Cultivation in Information Infrastructure Innovation. *Journal of the Association for Information Systems,* April, 15(4), pp. 220-243.

SAVI, 2015. *SAVI AGM '15 Poster Booklet,* s.l.: s.n.

SAVI, n.d. *Smart Applications on Virtual Infrastructure.* [Online] Available at: http://www.savinetwork.ca/

Sayer, R. A., 2000. *Realism and Social Science.* London: Sage Publications Ltd.

Schilling, M. A., 2000. Toward a General Modular Systems Theory and Its Application to Interfirm Product Modularity. *The Academy of Management Review,* 25(2), pp. pp. 312-334.

Scott, S. V. & Orlikowski, W. J., 2012. Reconfiguring relations of accountability: Materialization of social media in the travel sector. *Accounting, Organizations and Society ,* 37(1), pp. 26-40.

Scott, S. V. & Orlikowski, W. J., 2014. Entanglements in Practice: Performing Anonymity Through Social Media. *MIS Quarterly,* 38(3), pp. 873-893.

Seaman, C. B., 1999. Qualitative Methods in Empirical Studies in Software Engineering. *IEEE Transactions on Software Engineering,* July, 25(4), pp. 557-572.

Shapiro, C. & Varian, H. R., 1999. *Information Rules: A Strategic Guide to the Network Economy.* Boston, Massachuesetts: Harvard Business Review Press.

Silverman, D., 2010. *Doing Qualitative Research.* 3rd ed. London: SAGE.

Smith, J. & Nair, R., 2005. The Architecture of Virtual Machines. *Computer,* 38(5), pp. 32-38.

Smith, M. L., 2006. Overcoming Theory-Practice Inconsistencies: Critical Realism and Information Systems Research. *Information and Organization ,* 16(3), pp. 191-211.

Stanford University, Ethane, n.d. *Ethane: A Protection Architecture for Enterprise Networks.* [Online]
Available at: http://yuba.stanford.edu/ethane/index.html

Stanford University, SANE, n.d. *SANE/inSANE: Designing Secure Networks from the Ground-Up.* [Online]
Available at: http://yuba.stanford.edu/sane/

Star, S. L. & Ruhleder, K., 1996. Steps Toward an Ecology of Infrastructure: Design and Access for Large Information Spaces.. *Information Systems Research,* 7(1), pp. 111-134.

Suarez, F. F., 2005. Network Effects Revisited: The Role of Strong Ties in Technology Selection. *Academy of Management Journal,* 48(4), pp. 710-720.

Subramaniam, K., 2016. *Microsoft showcases "Software for Open Networking in the Cloud (SONiC)".* [Online]
Available at: https://azure.microsoft.com/en-gb/blog/microsoft-showcases-software-for-open-networking-in-the-cloud-sonic/
[Accessed 2 September 2016].

Sutton, R. I. & Staw, B. M., 1995. What Theory is Not. *Administrative Science Quarterly,* 40(3), pp. 371-384.

Tanenbaum, A. S., 2003. *Computer Networks.* 4th ed. Upper Saddle River, N.J.: Prentice Hall.

Taylor, R. N., Medvidović, N. & Dashofy, E. M., 2010. *Software Architecture: Foundations, Theory, and Practice.* Hoboken, N.J.: Wiley.

Tilson, D., Lyytinen, K. & Sorensen, C., 2010a. *Desperately Seeking the Infrastructure in IS Research: Conceptualization of "Digital Convergence" As Co-Evolution of Social and Technical Infrastructures.* s.l., s.n., pp. 1-10.

Tilson, D., Lyytinen, K. & Sørensen, C., 2010b. Digital Infrastructures: The Missing IS Research Agenda. *Information Systems Research,* 21(4), pp. 748-759.

Tokuda, L. & Batory, D., 2001. Evolving Object-Oriented Designs with Refactorings. *Automated Software Engineering,* 8(1), pp. 89-120.

van Schewick, B., 2010. *Internet Architecture and Innovation.* London, England: The MIT Press.

Verizon, 2016. *Verizon Network Infrastructure Planning: SDN-NFV Reference Architecture Version 1.0,* s.l.: s.n.

Verizon, 2016. *Verizon to Deliver Virtual Network Services to Support Digital Transformation of Enterprises.* [Online]
Available at: http://www.verizon.com/about/news/verizon-deliver-virtual-network-services-support-digital-transformation-enterprises
[Accessed 2 September 2016].

Verizon, n.d.. *Virtual Network Services: NFV - Verizon Enterprise Solutions.* [Online]
Available at: http://www.verizonenterprise.com/products/networking/sdn-nfv/virtual-network-services/index.xml
[Accessed 2 September 2016].

VMware Inc., 2012. *VMware to Acquire Nicira.* [Online] Available at: http://ir.vmware.com/overview/press-releases/press-release-details/2012/VMware-to-Acquire-Nicira/
[Accessed 9 December 2016].

VMware Inc., 2013a. *The VMware NSX Network Virtualization Platform,* s.l.: s.n.

VMware Inc., 2013b. *VMware NSX Network Virtualization Design Guide: Deploying VMware NSX with Cisco UCS and Nexus 7000,* s.l.: s.n.

VMware Inc., 2016. *VMware NSX The Network Virtualization Platform,* s.l.: s.n.

VMware Inc., n.d.-a. *Network Virtualization with VMware NSX Virtualized Network.* [Online]
Available at: http://www.vmware.com/uk/products/nsx/
[Accessed 2 September 2016].

VMware vSphere, n.d.-b. *Virtual Machine Live Migration vSphere vMotion.* [Online] Available at: http://www.vmware.com/uk/products/vsphere/features/vmotion/
[Accessed 2 September 2016].

VMware, 2016. *VMware NSX: From Turning Point to Tipping Point.* [Online] Available at: https://blogs.vmware.com/networkvirtualization/2016/05/vmware-nsx-from-turning-point-to-tipping-point.html
[Accessed 2 September 2016].

Volkoff, O. & Strong, D. M., 2013. Critical Realism And Affordances: Theorizing It-Associated Organizational Change Processes. *MIS Quarterly,* 37(3), pp. 819-834.

Volkoff, O., Strong, D. M. & Elmes, M. B., 2007. Technological Embeddedness and Organizational Change. *Organization Science,* 18(5), pp. 832-848.

Wang, A. et al., 2013. Network Virtualization: Technologies, Perspectives, and Frontiers. *Journal of Lightwave Technology,* Feb, 31(4), pp. 523-537.

Wegner, P., 1997. Why Interaction is More Powerful Than Algorithms. *Commun. ACM,* May, 40(5), pp. 80-91.

Weick, K. E., 1995. What Theory Is Not, Theorizing Is. *Administrative Science Quarterly,* 40(3), pp. 385-390.

Whetten, D. A., 1989. What Constitutes a Theoretical Contribution?. *The Academy of Management Review,* 14(4), pp. pp. 490-495.

Wolff, S., 2004. Analysis of Documents and Records. In: U. Flick, E. v. Kardorff & I. Steinke, eds. *A Companion to Qualitative Research.* London: SAGE, pp. 284-289.

Woodard, C. J., 2008. *Architectural Control Points.* Atlanta, Georgia, USA, s.n.

Wu, P. et al., 2013. Transition from IPv4 to IPv6: A State-of-the-Art Survey. *IEEE Communications Surveys Tutorials,* Third, 15(3), pp. 1407-1424.

Wyatt, S., 2008. Technological Determinism is Dead; Long Live Technological Determinism. In: O. A. M. L. Edward J. Hackett & J. Wajcman, eds. *The Handbook of Science and Technology Studies.* 3rd ed. Cambridge, Mass.: Cambridge, Mass. : MIT Press, pp. 165-180.

Wynn, J. D. & Williams, C. K., 2012. Principles for Conducting Critical Realist Case Study Research in Information Systems. *MIS Quarterly,* 36(3), pp. 787-810.

Yin, R. K., 2014. *Case Study: Design and Methods.* 5th ed. London: Sage Publications Ltd..

Yoo, Y., Henfridsson, O. & Lyytinen, K., 2010. The New Organizing Logic of Digital Innovation: An Agenda for Information Systems Research. Issue 4, pp. 724-735.

Zhang, Q., Cheng, L. & Boutaba, R., 2010. Cloud Computing: State-of-the-art and Research Challenges. *Journal of Internet Services and Applications,* 1(1), pp. 7-18.

Zhu, K., Kraemer, K. L., Gurbaxani, V. & Xin Xu, S., 2006. Migration to Open-Standard Interorganizational Systems: Network Effects, Switching Costs, and Path Dependency. *MIS Quarterly,* Volume 30, pp. 515-539.

Zittrain, J., 2008. The Generative Pattern. In: *The Future of the Internet and How to Stop it.* New Haven, US: Yale University Press, pp. 67-100.