DATA JOURNEY MODELLING: IDENTIFYING COST AND RISK IN LARGE, COMPLEX, SOCIO-TECHNICAL SYSTEMS

A THESIS SUBMITTED TO THE UNIVERSITY OF MANCHESTER
FOR THE DEGREE OF DOCTOR OF PHILOSOPHY
IN THE FACULTY OF SCIENCE AND ENGINEERING

2018

By
Iliada Eleftheriou
School of Computer Science

Contents

Word Count: 47487

List of Tables

List of Figures

To my partner and family whose love and support were vital throughout my journey.

Abstract

DATA JOURNEY MODELLING:
IDENTIFYING COST AND RISK
IN LARGE, COMPLEX,
SOCIO-TECHNICAL SYSTEMS
Iliada Eleftheriou
A thesis submitted to the University of Manchester
for the degree of Doctor of Philosophy, 2018

Managers in complex organisations often have to make quick decisions on whether new information sharing developments are worth undertaking or not. Such decisions are hard to make, especially at an enterprise level. Both costs and risks are regularly underestimated. Existing approaches to managing risk and estimating cost are principally focused on creating detailed predictions based on substantial models of the planned development. They aim to support project managers throughout the development process itself, rather than giving a low-cost indicator for use in early-stage decision making. Our objective is to help managers and stakeholders of large, complex organisations, such as the National Health Service (NHS) in the UK, make better informed decisions on points of cost and risk of new software systems that will reuse or extend their existing information infrastructure, before any implementation is undertaken.

We analysed 18 case studies describing recent software developments introduced by providers of health care services, looking for common points of high cost and risk. From the case studies analysis, we found that the movement of data within and between organisations was a key indicator of high cost and risk. Data movement can be hindered by numerous technical barriers, but also by other challenges arising from social aspects of an organisation. Hence, we devised a catalogue of socio-technical data movement anti-patterns that under certain conditions can introduce high cost and risk to the organisation.

In this thesis, we propose a new method aiming to identify places of high cost and risk when existing data needs to move to a new development. The method is low-cost

and combines both technical and social aspects, but relies only on information that is likely to be already known to key stakeholders, or will be cheap to acquire. The method is based on the data journey model, a new lightweight technique that captures movements of data within or between organisations. The data journey model describes an abstraction of large, complex eco-systems focusing on the high-level journeys data take through networks of people and systems.

To assess the effectiveness of our method and the accuracy of our predictions, we applied the method in real world settings in the NHS domain. We worked with clinicians to model the movements of data in five NHS studies from different Foundation Trusts across the UK. The results of the evaluation showed that our method was able to cheaply and quickly identify most of the points of high cost/risk that the hospital staff had identified, along with several other possible directions that the staff did not identify for themselves, but agreed could be promising.

Finally, the results of the evaluation showed that the data journey modelling method can be completed in less than a couple of hours (including training). Also, the simplicity of our modelling technique can empower domain experts with no particular modelling expertise to quickly identify opportunities for cost savings in new developments, as well as existing ones.

Declaration

No portion of the work referred to in this thesis has been submitted in support of an application for another degree or qualification of this or any other university or other institute of learning.

Copyright

Acknowledgements

"It's the people we meet along the way that help us appreciate the journey."

This project came to an end with the kind help and support of many individuals. I would like to extend my sincere thanks to all of the people who travelled this journey with me.

Foremost, I would like to express the deepest appreciation to my supervisors Suzanne Embury, and Andy Brass, the giants whose shoulders upon I stand. I particularly express my deepest gratitude to Suzanne for her unwavering support, collegiality, mentorship, and attention to detail throughout this project, but also, for her excitement in teaching and all the valuable life lessons she gave me. I also thank Andy for all his great ideas for this project, but also for prompting me to always see the big picture, to step back and reflect and finally teaching me that chaos is an opportunity waiting to be embraced.

Additionally, I would like to thank all of my collaborators from the Health eReseach Centre (HeRC), the School of Health Sciences and the Farr Institute of the University of Manchester for providing us access to the NHS case studies, and most importantly all the clinical staff who wrote them and those participated in the evaluation studies of this project. I particularly thank Rebecca Moden, Peter Dobinson, Algy Taylor, and the group of Clinical Consultants for their invaluable input and feedback that enable the application and evaluation of our contributions to a variety of NHS Foundation Trusts. I would also like to thank my office mates throughout the years, and particularly Mariam, Zety, Rene, Duhai and Ayesha, for providing the laughs, talks and cakes that kept me going. This project is supported by funding from the UK Engineering and Physical Sciences Research Council (EPSRC).

Last, but not least, I would like to thank my family and friends whose support and encouragement throughout my journey have been a vital part in completing this project.

Firstly, I would like to thank my partner Angelos for his love, unlimited patience and persistence, for believing in me and cheering any successes as trivial as they might be, but most importantly for being by my side through all the ups and downs of this journey. Next, I would like to thank my sister Sophie, who is more than just family to me, for all the emotional support, and constant sunshine she brings in my life, and for proving that a person is not complete without such a bright soulmate. I would also like to thank my brothers Captain Mike and mighty Kostui and friends Chairepetios, Kyriacos, Kouroupides, Vasilis, Valanto, and Ilia for all their encouragement, uplifting, and warm smiles throughout the years.

Finally, I am deeply grateful to my parents and life role models, Maria and Eleftherios, for the constant unconditional love they give me, for forming me into a strong and independent individual, and for always inspiring me and motivating me to become and accomplish more. Hope you will always remember how grateful I am, and how much I love you!

Chapter 1

Introduction

"Nothing is more simple than greatness; indeed, to be simple is to be great."

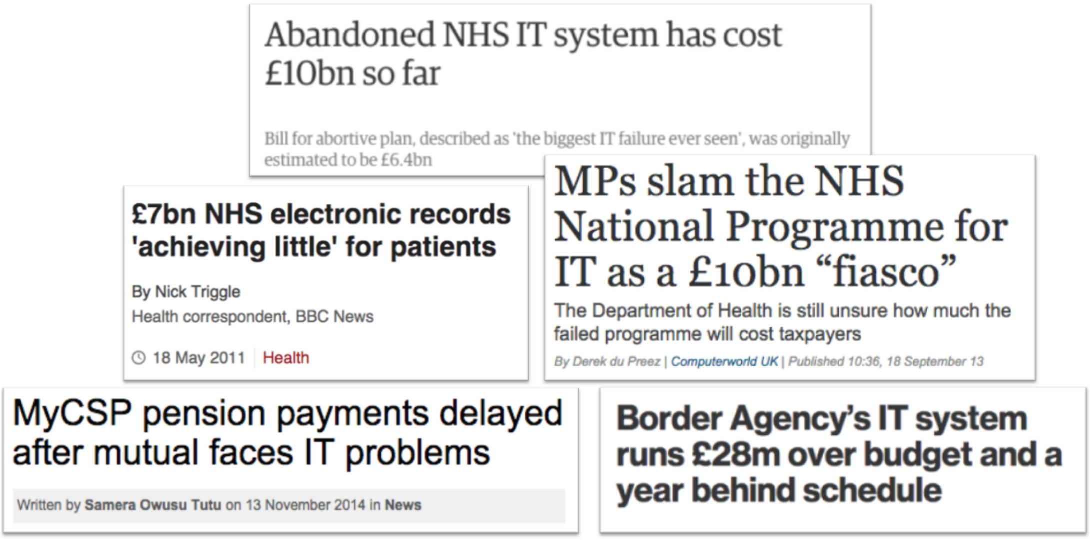
Ralph Waldo Emerson

1.1 Research Context and Motivation

Technological advances and business change drive organisations to develop new, more advanced information systems (IS) to either replace or add new functionality to the existing systems in place. However, realising value from these new ISs requires hard decisions to be taken. Is the new system development or re-design worth undertaking? Is the value to be gained more than the costs of developing and maintaining the new development? How can we make a good decision, and avoid wasting time and money in developing an IS that will not realise the planned benefits?

Managers and stakeholders of large, complex organisations have to take such decisions quickly and cheaply in the early-stages of the system's development. But, the costs of developing new IS are often underestimated. Costs far outweigh the value the new software developments propose to create and end up wasting immense amounts of money and resources.

An indicative example of cost underestimation in a large organisation, is the National Programme for Information Technology (NPfIT), an initiative by the Department of Health in the United Kingdom (UK). The NPfIT aimed to use modern information technologies to improve the delivery of health services and the quality of patient care [73, 126, 151]. Numerous information sharing and integration solutions were introduced under NPfIT, but after 12 years and a total forecasted cost of 9.8 billion GBP



Abandoned NHS IT system has cost £10bn so far

Bill for abortive plan, described as 'the biggest IT failure ever seen', was originally estimated to be £6.4bn

£7bn NHS electronic records 'achieving little' for patients

By Nick Triggle
Health correspondent, BBC News

18 May 2011 | Health

MPs slam the NHS National Programme for IT as a £10bn "fiasco"

The Department of Health is still unsure how much the failed programme will cost taxpayers

By Derek du Preez | Computerworld UK | Published 10:36, 18 September 13

MyCSP pension payments delayed after mutual faces IT problems

Written by Samera Owusu Tutu on 13 November 2014 in News

Border Agency's IT system runs £28m over budget and a year behind schedule

Figure 1.1: Newspaper headlines of failed IT developments.

(Great Britain Pound), the planned central, integrated system has yet to be established [15, 58, 89].

The review by the Cabinet Offices Major Projects Authority (MPA) on the NHS National Programme for Information Technology (IT) reported that although there have been substantial achievements which are now firmly established (i.e. Spine and N3 Network), the programme has not and cannot deliver to its original intent [136].

Looking through newspaper headlines or the Risks Digest List[1] we can see numerous other examples of large, complex Information Technology (IT) projects failing to deliver their planned benefits. Figure 1.1 shows some newspaper headlines on the failure of the Borders system, the pension system, and the NPfIT in the UK indicating the amounts of money wasted. Similarly, the Chaos Reports conducted by the Standish Group over the years show that although the percentage of IT systems that failed to realise the planned benefits is decreasing, it is still significantly high [70, 99]. In 2004, the Standish Group reported a shocking 29% project success rate, while 53% of the projects had overruns of costs, time or missing functionality, and 18% of the projects failed outright. Subsequent reports show similar figures [70]. The high percentage of challenged[2] and failed projects highlight the need for better understanding of potential

[1] www.risks.org

[2] According to the Chaos reports, a project is considered as *failed* if it was cancelled at some point during the development cycle, whereas a project is assessed as *challenged* in case it is completed and operational but over budget, over the initial time forecast, and if it offers fewer features and functions than originally specified. On the contrary, a project is *successful* if it is completed on time, within budget, and with all features and functions as initially specified.

points of cost and risk of a new IT development early in the decision making process.

Underestimating the costs may result in managers and stakeholders of the organisations to approve proposed systems that then exceed their budgets and fail to complete on time. The systems can end up with underdeveloped functions and poor quality, which can lead to the project to be cancelled, wasting resources and ultimately the loss of jobs [128]. This kind of IT failure, can ruin an organisation's reputation, as well as having a negative impact on the everyday lives of people who depend on the organisations' services. But most importantly, in the healthcare domain time, money, and other resources wasted on developing unsuccessful software, can ultimately cause the loss of patients' lives.

Current approaches to estimating costs and managing risks of software developments are principally focused on creating detailed predictions based on substantial models of the planned developments [26, 98, 132, 171, 172]. They are aimed at supporting project managers throughout the development process itself, rather than giving a low-cost indicator for use in early-stage decision making.

Managers and stakeholders need better tools for identifying potential points of high cost and risk in order to make a well informed go / no go decision on new IT developments, especially in large, complex systems, such as national health and social care providers. A project should ideally proceed only if it will deliver the benefits expected of it, for roughly the planned implementation costs. It should not proceed if the costs will be much higher than planned, and will exceed the value to be gained from it.

The Medical Research Council (MRC) guidelines for complex interventions provide guidance on the process for developing and evaluating health technology interventions, such as decision support systems [57]. The aim of the framework is to ensure that interventions are empirically and theoretically founded, and that considerations are given both to the effectiveness of the intervention and how it works. Although the MRC guidelines are helpful in ensuring medical research, they do not provide us with the tools to identify points of high cost and risk in a planned new development. They are, however, an important set of guidelines to follow for later on in the go / no go decision making process.

1.2 Research Problem

In this thesis we describe our efforts looking for an approach to better identify high cost and risk early in the process to help managers and stakeholders of large, complex organisations make good decisions on whether is worth developing a new IT system, or not. The approach must be able to be used by low-level managers and stakeholders when some IT development is planned to happen usually to save costs or to allow some new functionality that was not supportable before. Clearly, any decision-making aid must be capable of modelling the effects of a range of factors (both technical and social), in a lightweight way that is suitable for use before much (or ideally any) development work has taken place.

The approach must be low-cost that can be completed in the course of a small number of days to be suitable for early-stage decision making. Also, the approach must give reliable predictions of potential places of high cost and risk in the new development.

1.3 Proposed Solution

In this thesis we propose a new low-cost method for identifying points of high cost and risk in future IT developments [67]. A fundamental part of our method is the data journey model, a novel lightweight modelling technique that captures the movement of data through complex networks of people and systems [65].

The method and the data journey model are created by analysing a collection of 18 case studies written by staff working for the National Health Service (NHS) in the UK, looking for factors that contributed to the failure or success of a recent IT development in which the case study authors had been involved in. Of the 18 case studies, only three were described by their authors as having been successful. The remaining 15 were categorised by the study authors as having (completely or partly) failed to deliver the expected benefits. The results of our analysis show that IT projects failed due to a mixture of human and technical factors, with the human factors being by far the most dominant.

A common feature of the case studies where the new software was deemed to have been unsuccessful was the movement of data. Whenever data were moved to new contexts, and used for purposes other than those for which it was originally designed, the system owners and end-users faced a host of additional challenges, be they organisational, technical, human, governance oriented or political in nature. These challenges lead to unforeseen costs and sometimes dramatic reductions in the benefits expected from the new software. We therefore hypothesise that identifying the need for movement of data in a new development can provide early warning signal for failure. We devised a method that firstly maps the journey a set of data make through complex networks of people and systems, using the data journey model. The second step in our method overlays on the data journey model socio-technical (social, people, organisation and technical-oriented) information that can help us identify the places in the journey where data move a boundary and can introduce high cost and risk to the organisation.

To test our hypothesis, we applied our method to real world studies in the healthcare domain. We worked with clinicians from five NHS Foundation Trusts (FT) across the UK to model the movements of data of recently introduced IT systems and identified points of high cost and risk. The results of the evaluation showed that our method was able to cheaply and quickly identify most of the points of high cost/risk that the hospital staff had identified, along with several other possible directions that the staff did not identify for themselves, but agreed could be promising [66].

1.4 Research Questions

In this thesis, we are investigating the primary research question of whether we can quickly and cheaply identify potential points of cost and risk in new large, complex IT developments to assist the early stage decision making process. We begin addressing this question by breaking it down into seven smaller tangible research questions (RQ):

- **RQ 1: What factors contribute to the failure of a new IT development that can impose high cost and risk to the organisation?**
  We have seen several cases where cost and risk of developing large complex IT systems have been underestimated, or sometimes not even properly identified beforehand. To predict whether a new development is likely (or not) to fail realising the expected benefits we first have to identify the factors contributing to the failure of such systems. Are those factors coming from the technical nature of large IT systems, or from the environment they are placed in? What are the characteristics of an IT system that, if present, can increase the likelihood of significant cost/risk?

- **RQ 2: Are there any early warning signs of high cost and risk when developing a new IT system?**
  Are there any indicators that can indicate the existence of high cost and risk in a new IT system? How can these indicators be used to provide early warning signs of potential system failure?

- **RQ 3: How can we abstract away from the complexity of large organisational IT systems, to quickly and cheaply identify the aforementioned warning signs?**
  In early stage decision making process, we must be able to quickly and cheaply identify warning signs. We can not spend a long time and resources on analysing and planning a system that we might not pursue. To achieve this, we need a conceptual model that abstracts away from the inherent complexity of large IT systems. How can we quickly model a system and the environment in which it will be consolidated in? How can we abstract away from the complex details of business processes, and environmental interactions of the IT eco-system and mainly focus on identifying warning signs of failure?

- **RQ 4: How can we quickly identify the warning signs of high cost/risk in the aforementioned abstract model of the new system?**
  We need a lightweight approach to overlay on the conceptual model of the planned new system the indicators of system failure in order to reveal the early warning signs. Is the approach quick and cheap to be suitable for the early stage decision making process?

- **RQ 5: Can the warning signs accurately and cheaply identify points of high cost and risk in a planned new system?**
  How useful are the warning signs in predicting cost/risk places in the development? Can they really help managers and stakeholders make better informed decisions on the cost and risk of developing new functionality? Is the method of identifying the warning signs truly lightweight, so that it can be used in early stage decision making?

- **RQ 6: Can the warning signs identify points of high cost and risk of new systems, across domains?**
  Can we apply the warning signs in a variety of settings, or will each setting require domain-specific warning signs to be found? We must have a set of signs that are stable across domains for the approach to be lightweight.

- **RQ 7: Do the warning signs identify all the significant points of high cost and risk of a new system?**
  Are all the important places of cost and risk identified, or do we need additional

signs when applied in other domains? There is the possibility that some particular organisation, or context might have highly specific requirements that should be taken into account in the identification of cost and risk. Is there a low cost way to determine whether such a setting has specific requirements?

1.5 Research Contributions

Attempting to answer the above research questions we produced the following research contributions:

1. **A set of IT failure factors in healthcare.**
   Analysing the case studies from the NHS domain, we extracted a set of 32 factors that contributed to the failure of the IT systems recently introduced in a variety of settings in the NHS. We found a mixture of technical, human and organisational factors that according to NHS staff were responsible for the failure of the systems.

2. **A catalogue of data movement anti-patterns.**
   We devised a catalogue of data movement anti-patterns providing early warning indicators of problematic movements, that under certain criteria, may introduce high cost and risk in planned new developments.

3. **A novel modelling technique that captures data movements in large complex IT systems, called data journey modelling.**
   Large IT systems require movement of data between system components, but also between people and organisational structures. In order to capture these movements we propose a new modelling technique that provides an abstraction of large enterprises focusing on the journeys data make between collaborating systems of often different organisations.

   Data journey models capture movements of data between complex networks of people, systems, and organisations. They are lightweight; they abstract away from the complexity of business processes. They focus on capturing socio-technical elements that can contribute as early warning signs of high cost and risk. The simplicity of the model enables domain experts to quickly, and cheaply design data journeys, without requiring extensive technical modelling knowledge.

4. **A lightweight method that identifies socio-technical points of cost and risk, called boundaries, in data journey models.**
   Data movement boundaries are cheap socio-technical warning indicators of problematic movements that can potentially impose high cost and risk to the organisation and potentially cause the failure of the planned new IT system.

   We found five types of boundaries, that are cheap to acquire and apply. The boundaries cover factors affecting the movement of data on an organisational level (i.e. data movement between systems of different organisational structures, and data movement between organisational structures with different governance frameworks), human level (i.e. data movement between systems used by different staff roles), and technical level (i.e. data movement between systems of different technical requirements, and movement of immense volume of data).

   We devised a method that applies the above socio-technical boundaries onto a data journey model to identify the places in the journey of potential high cost and risk. The method is lightweight and low-cost. It can be completed in a couple of hours (depending on the scale and abstraction level of an IT project) and relies only on cheap-to-acquire and cheap-to-apply information.

5. **A low-cost method to identify boundaries in domain-specific settings.**
   We propose an up-front and low-cost approach to identify socio-technical boundaries in settings where domain-specific requirements that might drive the need for additional boundaries.

6. **The application of the model and method in real-world case studies in the healthcare domain.**
   We evaluated the accuracy, stability and completeness of our data journey modelling method in identifying places of high cost and risk in five case studies in the NHS domain.

1.6 Publications

All the novel contributions described in the previous section, have been published in conference and journal articles in both the computer science and health informatics domains. Below we give details of the publications, and list the contributions which each describes:

- Eleftheriou Iliada, Suzanne M. Embury, and Andrew Brass. *"Data Journey Modelling: Predicting Risk for IT Developments."* In IFIP Working Conference on The Practice of Enterprise Modeling, pp. 72-86. Springer International Publishing, 2016 [65].
  Contributions described: 1-3.

- Eleftheriou Iliada, Suzanne M. Embury, Rebecca Moden, Peter Dobinson, and Andrew Brass. *"Data Journeys: Identifying Social and Technical Barriers to Data Movement in Large, Complex Organisations."* Journal of Biomedical Informatics, 2017 [68].
  Contributions described: 4-5, and the accuracy part of 7.

- Eleftheriou Iliada, Suzanne M. Embury, and Andrew Brass. *" Light Touch Identification of Cost/Risk in Complex Socio-Technical Systems"* In IFIP Working Conference on The Practice of Enterprise Modeling, pp. 65-80. Springer International Publishing, 2017 [66].
  Contributions described: 6-7.

1.7 Thesis Overview

The remainder of this thesis is organised as follows:

Chapter 2 provides background on existing cost estimation and risk identification techniques and describes the need for a lightweight and cheap approach in identifying points of cost/risk.

Chapter 3 reviews the literature on data movement and data interoperability and demonstrates the research gap that is being addressed in this thesis.

Chapter 4 explains and justifies the methodological approach we followed and why it is best suited in achieving the research questions.

Chapter 5 investigates factors contributing to the failure of recent IT developments in the healthcare domain (RQ 1). It also proposes a set of data movement antipatterns as early warning signs of high cost/risk in future developments (RQ 2).

Chapter 6 proposes a new lightweight modelling technique that captures the movement of data through and across organisations, aiming to identify problematic data movements that can lead to high risk and cost (RQ 3).

Chapter 7 describes a low-cost method that identifies places in the journey of data that can impose high cost and risk to the organisation (RQ 4). The method uses cheap socio-technical boundaries to identify points of high cost/risk.

Chapter 8 evaluates the main contributions of the thesis in a variety of real-world case studies in the NHS. The first case study evaluates the accuracy and lightweight property of our modelling method in identifying potential high cost and risk in a new development in the radiology department of an NHS hospital (RQ 5). Then, we assess the stability and completeness of the method in four case studies in the domain of Clinical Genomics (RQ 6 and RQ 7).

Chapter 9 provides the research contributions presented in this thesis, discusses findings, their significance in the information systems community and strengths and weaknesses of the methodological approach. Finally, it provides future research directions.

Chapter 10 summarises the findings of this thesis.

Chapter 2

Background

2.1 Introduction

In this chapter, we provide a background on the key concepts we will explore in this thesis. We begin by providing a rationale for the research and explore the properties of large, complex IT systems that make implementation of go / no go decisions hard when adding new functionality in already crowded infrastructures (section 2.2). Then we investigate existing techniques and tools for identifying points of high cost and risk in large, complex organisations. We explore current software cost estimation tools and techniques and risk identification approaches demonstrating the need for a lightweight approach (section 2.3).

2.2 Research Rationale

Our economy and society is becoming increasingly dependent on large, complex IT systems that are often created by integrating and orchestrating independently managed software systems [165]. There is a constant need for new developments to be embedded into existing networks of systems to either optimise old ones, or to expand current functionality. In large IT systems, there is a complex mix of factors to be considered when deciding whether to proceed with a new development, or not. In this section we explore the factors that make decision making so hard in large, complex organisations, providing a rationale for the research in this thesis. While exploring the factors we define key concepts that will be used throughout the thesis.

A key characteristic of large IT systems is that systems are assembled from other existing and new systems, which can be independently controlled and managed by

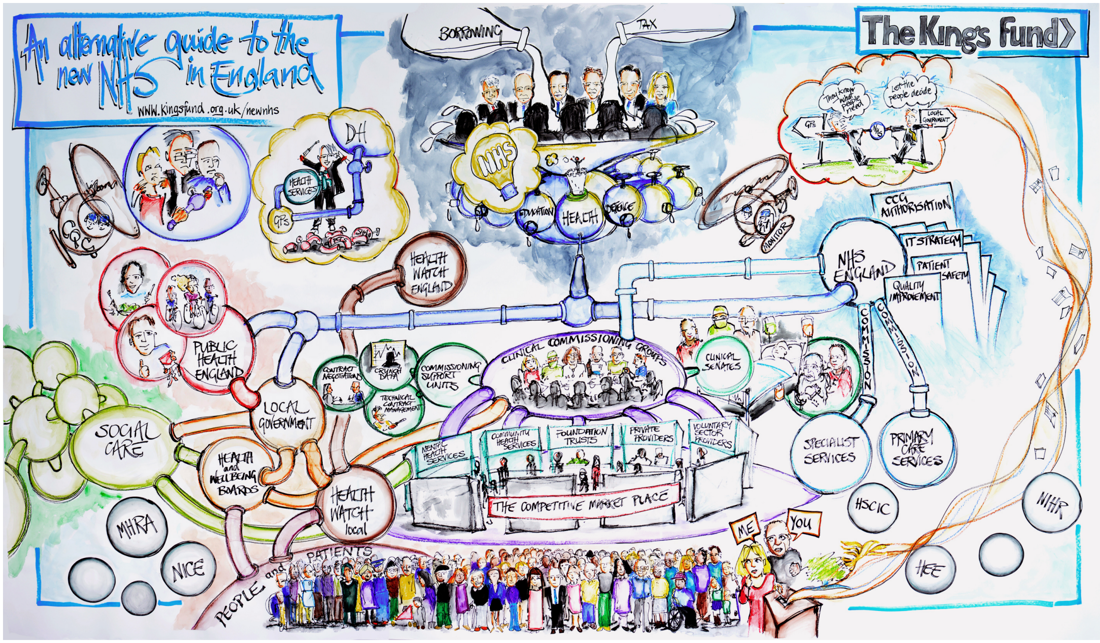


Figure 2.1: NHS structure, as given by The Kings Fund.

different teams of a large organisation.

Definition 2.2.1. Large IT system: An Information Technology system of an organisation that consists of other smaller systems that have to interact together to accomplish the functionality needed to operate the organisation.

Definition 2.2.2. IT development: An Information Technology software system or software functionality that is planned to be integrated in the existing infrastructure of an organisation. In the rest of the thesis we refer to this term as a new development.

For example, the NHS is assembled from a myriad of Foundation Trusts, General Practitioners (GPs), health services, specialist services, regulation agencies, Clinical Commissioning Groups (CCGs), etc. Figure 2.1 shows a drawing created by The Kings Fund[1] attempting to show the different organisations that assemble the NHS and how they fit together. Each bubble in the figure is an independent organisation with several other sub-organisations that have to work together to provide better health care. Each sub-organisation has its own independent but interdependent IT systems in place reflecting the guidelines, processes, and politics of the organisation.

[1] The drawing was created in 2013 and does not fully represent the current NHS structure. An updated version reflecting recent changes is due to be published soon at https://www.kingsfund.org.uk/

This example broadly outlines the three big challenges of large, complex IT systems: complexity, information sharing, and socio-technical environment. We describe each of them in the following sections.

2.2.1 Complexity

According to Sommerville et al. the complexity of a large scale IT system stems from the number and type of relationships between the system components, but also between the system and the environment in which it runs [165]. In large scale organisations, it is rarely the case that top-down systems design is followed. Instead, a bottom-up approach is more common. Sub-organisations and departments each produce their own software developments that cover the requirements and needs of the department, or organisational team. If new regulations or requirements come in that need sharing or integration of information among these departments, then new functionality needs to be designed that integrates or shares the information stored in the various departmental systems. This is referred to in the systems engineering research, as a **system of systems** (SoS) [1, 94].

Maier mentions that the distinction between a system of systems and a complex monolithic system is that the elements of a SoS are operationally and managerially independent [125]. The need of these independent components of a SoS to collaborate significantly increases their complexity. In this thesis we use the term *network of systems*, as to include interdependent systems that often constitute large organisations.

Definition 2.2.3. Network of systems: A network of systems, or system of systems, is a set of operationally independent and/or interdependent IT systems that have to collaborate together to achieve some value.

Inherent complexity stems from the dynamic relationships between the components in a network of systems. The relationships between the components of such a network change since they are not independent of the ways that the constituent systems are used. It is very hard to analyse and predict inherent complexity during system development as it depends on the systems dynamic operating environment [103, 165].

However, even if the relationships between system components are simpler and static, the number of components can be so large that tracking and understanding the relationships between them can be hard. Rushby characterises this type of complexity as epistemic; stems from our lack of knowledge about the system rather than inherent system characteristics [157]. Epistemic complexity increases with the size of the system coalition and the number of relationships between them becoming harder to understand, and making it even harder to identify potential cost and risk.

Introducing new functionality in an already crowded large coalition of systems increases both inherent and epistemic complexity. Understanding the system and the relationships between the existing and the new components becomes harder. Identifying potential cost and risk that the new development can impose on the network of systems and the organisation can be labour intensive and a very time consuming process.

2.2.2 Information Sharing

Large IT systems are complex; they consist of lots of independent sub-systems, each organised and run by different departments. This coalition of independent systems have to work together to deliver the expected organisational benefits. Information sharing across a network of systems is considered to be one of the key aspects to establishing organisational efficiency and performance.

However, information sharing can be a complex task. Identifying factors that influence information sharing is critical. In the literature, research in information sharing focuses on the interpersonal (i.e. face-to-face discussions, exchange of emails), intra-organisational (sharing of information between departments of an organisation), and inter-organisational (information sharing between organisations) levels.

Definition 2.2.4. Information Sharing: Information sharing is the exchange of information between sender and receiver points. Exchange points can be either people or electronic systems and the transfer of information is bidirectional.

Information sharing can be implemented by a myriad of open and proprietary protocols, coding, message and file formats. In the case of electronic information sharing, several initiatives were formed to standardise sharing protocols, such as Extensible Markup Language (XML), Simple Object Access Protocol (SOAP), and Web Services Description Language (WSDL).

Different industries have and use other protocols for sharing information. For example, in the healthcare domain a set of international standards, called Health Level 7 (HL7) has been formed by the Health Level Seven International, an international standards organisation. HL7[2] overlooks the transfer of clinical and administrative data

[2] The HL7 standards focus on the application layer, which is “layer 7” in the Open Systems Inteconnection (OSI) model.

between software applications used by various healthcare providers.

HL7 protocol specifies a number of flexible standards, guidelines, and methodologies by which various healthcare systems across the world can communicate with each other. Such guidelines or data standards are a set of rules that allow information to be shared and processed in a uniform and consistent manner. These data standards aim to allow healthcare organisations to easily share clinical information.

Another example of information sharing protocols in the business domain is the Electronic Data Interchange (EDI). EDI is the concept of businesses electronically communicating information that was traditionally communicated on paper, such as purchase orders and invoices.

Achieving information sharing within a large complex IT system that constitutes a network of systems brings several challenges. Different organisations have various types of hardware and software systems as part of their infrastructures and it is a challenge to integrate heterogeneous information systems of different platforms, data standards, data schemas, and data with various data quality issues.

2.2.3 Socio-Technical Environment

The complexity of a large coalition of systems also depends on the socio-technical environment in which is running. Russel defines a system's environment as:

> "The environment of a system is a set of elements and their relevant properties, which elements are not part of the system but a change in any of which can produce a change in the state of the system. Thus a system's environment consists of all variables which can affect its state." [1]

Large, complex IT systems are affected by the environment in which they are formulated. People and technology need to work together to produce socio-technical systems facilitating the complex interactions in a workplace. Socio-technical systems (STS) in organisational development is an approach to complex organisational work design that recognises the interaction between people and technology in workplaces. The term also refers to the interaction between society's complex infrastructures and human behaviour [154, 173].

A complex mix of factors come from socio-technical environments when deciding whether to proceed with a new development, or not. Technical difficulties arise when sharing or integrating information, often stemming from the diverse data sources involved. Other challenges stem from the social aspects of the organisation: its people, policies, processes, governance, etc.

Examples can be found in the health care domain in which people are reluctant to change their current processes to use the new system in place, or user requirements are not met because of conflicting organisational policies and governance issues, and many more [8, 20, 87, 91, 180, 188]. In particular, Lann examines 'resistance to change' as a challenge in large-scale agency IT projects [175]. Lann reports that a dominant common problem with such large projects is the manifestations of resistance to the tools developed for the projects' implementation. This resistance may occur inadvertently when new grammars of modern IT projects clash with more traditional grammars often found in the culture of public organisations [175]. In another example, Greenhalgh *et al.* show the importance of human factors affecting the integration of electronic patient record (EPR) systems [80]. They state that the lack of consideration of the human factors is detrimental when bridging the model-reality gap in EPR systems.

All these challenges have immediate effect on the cost and risk of developing a new IT system. Any decision making tool aiming to capture cost/risk of developing and maintaining an IT system must be able to identify both technical and social challenges stemming from the socio-technical environment in which the system will be embedded.

2.2.4 Summary

Having explored the three main challenges of developing large complex IT systems, we have seen the reasons why the decision making process of whether to proceed with a planned new development is hard. The big number of software project failures, where software is delivered late and over-budget, is a consequence of these challenges [48].

To help us address the big challenge of developing new large IT systems, we need to embrace inherent complexity and coalitions of IT systems. We also must consider people and organisations that constitute the systems' socio-technical environment. We need to represent, analyse, and model the eco-system environments for such IT systems to help us understand the complex relationships between them.

2.3 Cost and Risk Estimation

In this section we provide a background on existing approaches on cost estimation and risk identification emphasising on their limitations and the reasons why we can not or can use them in our research project. We begin be defining a protocol by which we are formulating our investigation based on a set of criteria that are needed to answer our hypothesis.

In particular, we are looking for techniques that allows managers of large, complex organisations to make good, low-cost go / no go decisions at an early stage in the project. We need a lightweight, low-cost approach suitable for early stage decisions that can estimate potential cost and risk of the planned development.

We need an approach with the following characteristics:

- **Complexity:** An approach that allows us to predict points of cost and risk of planned new IT developments that are often part of bigger, more complex ecosystems and need to share information from collaborating organisations.
- **Socio-technical environment:** The approach must give equal prominence to both social and technical factors affecting the cost and risk of developing a new IT development.
- **Lightweight:** The approach must be sufficiently lightweight and low-cost to be used as a decision-making aid in the early stages of a development cycle, ideally before any implementation is initiated.

2.4 Software Cost Estimation Models

Having established our criteria, in this section we describe existing software cost estimation models and techniques that cover some if not all of the above characteristics. Software cost estimation models are typically categorised into two broad clusters: algorithmic and non-algorithmic. Algorithmic models are based on mathematical formulas of a wide variety of complexity. Some of them are based on simple arithmetic formulas using statistic algorithms, such as means and standard deviations [64]. Others are based on regression models [178] and differential equations [149]. We begin by describing the metrics that these models use, we then present the algorithmic approaches and then outline the non-algorithmic techniques.

According to Lederer *et al.* the main factors that are typically estimated at the beginning of an IS development project are: cost, size, schedule, people resources, quality, effort, resources, maintenance costs, and complexity [113]. A plethora of cost estimation techniques has been proposed in the last few decades [26, 37, 100, 105]. Cost is usually estimated in terms of the time and effort needed to develop and maintain a new system, and the complexity of the new system.

Definition 2.4.1. IT Cost: In information technology, the term ‘cost’ is used as the cost associated with software development, acquisition, and maintenance. In this thesis we use cost in terms of the *time*, *effort* and *resources* needed to acquire, develop and maintain an IT development.

Software cost estimation models calculate complexity cost in terms of its size. The software size is one of the most important factors that affects the software cost [114]. The most popular software size metrics used in practice are described below:

Lines of Code commonly known as LOC, lines of code represents the number of lines of the delivered source code of the software [3, 155] Although LOC depends on the programming language chosen, it is the most widely used software size metric [155]. Most models use this to measure software cost. However, exact values of LOC can only be obtained after the project has been completed. Hence, any software cost models using this metric are not suitable for our criteria as estimating the code size of a program before it is actually built is almost as hard as estimating the cost of the program.

Code length and volume metrics have been proposed by Halstead [85]. Code length is used to measure the source code program length based on the number of operator and operand occurrences. Volume represents the amount of required storage space for the software and is based on the number of distinct operators and operands in a software program [85].

Function points express the amount of business functionality a software system provides to the users [3]. The method first identifies the functional user requirements of the software and categorises them into one of five types: outputs, inquiries, inputs, internal files, and external interfaces. Each requirement is then assessed based on its complexity and is assigned a number of function points [147].

Feature points extend the function point metric and estimates cost based on the main features of the software. Features can be identified based on the algorithms a software has [97]. However, as several features in a software can have different degrees of complexities, feature point is not a reliable technique to use [97].

Object points measure the size of a software system based on the number and complexity of the following objects: screens, reports and third-generation language (3GL) components [52, 88, 114, 134]. The method counts the instances of such objects in the software and assigns a value ranging from one (simple screen) to 10 (3GL component). This method is lightweight and suitable for the early stages of designing a new software [114].

2.4.1 Algorithmic Methods

Algorithmic software cost estimation approaches are classified by Boehm *et al.* in 6 categories: model based approaches, regression based models, learning oriented models, expert based approaches, and finally composite bayesian methods [26]. In this section we present the most popular ones, and then we describe how they solve (part of) the problem.

Constructive Cost Model (COCOMO) is one of the most popular algorithmic cost estimation models developed by Barry Boehm and published in 1981 [28]. COCOMO predicts the effort and duration of developing a new software project, based on inputs relating to the size of the resulting systems and several cost driver parameters that according to Boehm affect productivity. The COCOMO Basic Model, a simplified version of the model, identifies effort 'E' as function of program size, using the following equation:

$$Effort = a * (size)^b \tag{2.1}$$

where *effort* is calculated in number of man-months (e.g. 152 working hours), *a* and *b* are the set of values on the complexity of software (for organic projects a=2.4 while b=1.05, for semi-detached a=3.0 and b=1.1.2, and for embedded a=3.6 and b=1.2). Software size is represented in terms of delivered source instructions. In an effort to improve the models accuracy, Boehm refined the equation to include the effects of 15 cost drivers (attributes of the end product, the computer used, the personnel staffing, and the project environment) and proposed the intermediate COCOMO model [28].

Although Constructive Cost Models have evidences of high accuracy and are widely popular, they require the system and software requirements to be predefined and stable [26]. In the cases that the COCOMO model is used at the early stages of software development, it may lead to estimation failures [105].

Software Life-cycle Model (SLIM) was proposed by Putnam *et al.* in the late 1970s [150]. SLIM is based on the Rayleigh distribution of the project personnel level versus time. It uses the Rayleigh curve to estimate project effort, schedule and defect rate. The curve is formulated based on Manpower Buildup Index (MBI) and

Technology Constant or Productivity factor (PF). To get the values for the MBI and PF, SLIM uses data from previously completed projects or in the case that data are not available then a set of questions can be answered that can find out both values. Other techniques were later proposed that are based on the SLIM. These are: SLIM-Estimate a project planning tool, SLIM-Control a project tracking tool, and SLIM-Metrics a software metrics repository and benchmarking tool [26, 105, 150].

Checkpoint is a knowledge-based software project estimating tool developed from Capers Jones studies [26, 150]. It has a proprietary database of about 8000 software projects and it focuses on four areas that need to be managed to improve software quality and productivity. It uses function points (or feature points) as its primary input of size. Checkpoint is designed to support the entire software development life-cycle [26].

The **PRICE** model was first released in 1977 as a proprietary model and was used in National Aeronautics and Space Administration (NASA) and other government software projects [26, 142]. The model forecasts software costs and schedules, but also facilitates estimating the size of the software to be developed. Software sizing uses Source Lines of Code (SLOC), function points or Predictive Object Points (POPs). POPs is an approach for sizing object oriented development projects introduced by Minkiewicz [134].

Other model based algorithmic techniques are ESTIMACS, COBRA, SEER-SEM, COSMIC (Common Software Measure- ment International Consortium), and SELECT estimator. Overall, model based techniques have been used for budgeting, tradeoff analysis, planning and control, and investment analysis [26]. They use software size as a primary way to estimate cost. Software size can be estimated in SLOC, function points, and object points. These approaches have evidently showed high accuracy in certain cases [26, 37, 100, 147].

However, as they are calibrated to past experience, their primary difficulty is with unprecedented situations. These approaches need detailed plans of the future IT development, that requires days, often months to capture. Although they are evidently accurate in most of the cases, they are not suitable for early stage decision making early in the IT development process. They are aimed for supporting the entire software development life-cycle.

2.4.2 Non-algorithmic Estimation Methods

The non-algorithmic estimation approaches use data collected from past projects and/or experts knowledge on cost estimation. They are usually used in combination with an algorithmic model.

Analogy costing This method estimates costs through reasoning by analogy using the actual costs of similar previous projects [163]. Estimation by analogy can be done either for the full project or for a subsystem. Although this method's estimates are based on actual project experience, new developments are rarely similar with each other. Additionally, the method does not explicitly recognise whether previous projects are representative of the constraints, environment and functions of the new system.

Expert judgement This method relies on one or more experts to provide estimates using their own methods and experience. In the event of inconsistencies, expert-consensus mechanisms, such as Delphi and PERT techniques can be used to reach consensus [93, 122]. The expert judgement approach must be used in combination with other methods.

Parkinson This method is based on the Parkinson's principle "work expands to fill the available volume" [144]. Costs are not estimated, but determined based on the available resources of the project.

Bottom-up This method separately estimates the cost of each component of the software development. The estimates are then aggregates to estimate the cost of the overall development. This approach requires an initial design plan of the development that shows how the system is decomposed into different components.

Top-down This approach estimates an overall cost of the system typically using other algorithmic or non-algorithmic method. The overall cost is then divided among the various components.

2.4.3 Overview of Cost Estimation Techniques

Despite a plethora of cost estimation models has been proposed in the literature, none of the models cover the criteria we are looking for: abstract, socio-technical, and lightweight. Model based approaches and regression models are fine-grained and require significant time and effort to be invested by domain experts to achieve a degree of accuracy. They rely on substantial models and focus on creating detailed predictions.

Although most of the algorithmic models are powerful techniques with some documented high levels of accuracy, they are more suitable for estimating costs in existing

IT developments since they need detailed information (like the number of code lines and function points) to produce an outcome; information that is not yet available to the managers of the organisation in the early stages of the decision making process. On the contrary, non-algorithmic approaches they do not necessarily require a detailed plan of the future development. However they rely on knowledge of previous projects that may not always be available.

At the initial stage of a project, there is high uncertainty about the project's attributes. The estimate produced at this stage is inevitably inaccurate, as the accuracy depends highly on the amount of reliable information available to the estimator. As we learn more about the project during analysis and later design stages, the uncertainties are reduced and more accurate estimates can be made. Most approaches produce exact models without embracing uncertainty.

Several researchers claim that these techniques need to be enhanced to produce a range of estimates and their probabilities. They believe that to produce better estimates, further research on improving our understanding of these project attributes and their causal relationships, model the impact of evolving environment, and develop effective ways of measuring software complexity [42].

Strike *et al.* claim that it is fundamental that the estimated cost of a particular software project is ascertained as early in the development cycle as possible as it enables project managers to make critical business decisions in a timely manner [167]. Knowing the cost estimate when the project is almost completed, or even halfway through the development, is of diminishing benefit because as the project progresses, more and more is invested, and the harder it becomes to abandon the runaway project [30, 124].

2.5 Software Risk Identification

Software risk management emerged several decades ago by Boehm [25, 44]. Websters dictionary defines 'risk' as the possibility of loss or injury. Risk can also be defined as the intentional interaction with uncertainty. Uncertainty is a potential, unpredictable, and uncontrollable outcome.

Risk exposure, also referred to as 'risk impact' or 'risk factor', is defined by the relationship:

$$RE = P(UO) * L(U0) \tag{2.2}$$

where RE is the risk exposure, P(U0) is the probability of an unsatisfactory outcome and L(U0) is the loss to the parties affected if the outcome is unsatisfactory in the form of money expenses [25]. Given that software projects involve several stakeholders (such as customers, developers, users, and maintainers), each with different, but equally important satisfaction criteria, then unsatisfactory outcome can be of different dimensions: budget overruns and schedule slips for customers and developers, wrong functionality, user-interface shortfalls, performance shortfalls, reliability shortfalls for users, and poor quality software for maintainers.

Definition 2.5.1. IT Risk: In information technology, risk is defined as the possibility of loss caused by an unsatisfactory outcome. Loss can be in the form of money expenses, lost time and/or resources.

Boehm defines the top 10 primary sources of risk on software projects (based on a survey with experienced project managers) to be the following [27]:

- Personnel shortfalls: Staff with top talent, job matching, team building, key personnel agreements,cross training
- Unrealistic schedules and budgets: Detailed multisource cost and schedule estimation, design to cost, incremental development, software reuse, and requirements scrubbing.
- Developing the wrong functions and properties: Organisation analysis, mission analysis, operations-concept formulation, user surveys and user participation, prototyping, early users manuals, off-nominal performance analysis, quality-factor analysis.
- Developing the wrong user interface: Prototyping, scenarios, task analysis, user participation.
- Gold-plating: Requirements scrubbing, prototyping, cost-benefit analysis, designing to cost.
- Continuing stream of requirements changes: High change threshold, information hiding, incremental development (deferring changes to later increments).
- Shortfalls in externally furnished components: Benchmarking, inspections, reference checking, compatibility analysis.
- Shortfalls in externally performed tasks: Reference checking, preaward audits, award-fee contracts, competitive design or prototyping, team-building.

- Real-time performance shortfalls: Simulation, benchmarking, modelling, prototyping, instrumentation, tuning.
- Straining computer-science capabilities: Technical analysis, cost-benefit analysis, prototyping, reference checking.

Risk identification is the first step in the risk management lifecycle of an IT project. In this first step, risk identification produces a list of risk items, like the one provided above, but specific to an IT project. The list of potential risks is then examined to assess likely compromise of a project's success that can impose high costs to the organisation, and risk the project's (partly or complete) failure.

Throughout the last decades, numerous techniques were found aiming to help risk identification. Here, we present those that cover (some) of our criteria we set earlier in the chapter.

Typical risk identification techniques for the early stage of an IT development are checklists, examination of decision drivers, comparison with experience (assumption analysis), and decomposition [27, 33, 92]. We outline these approaches below:

Taxonomy-based consists of a Taxonomy-Based Questionnaire and a process for applying the method [40]. The taxonomy organises software development risks into three levels: class, element, and attribute. The questionnaire has a set of questions under each taxonomic level which is specifically designed to elicit the range of risks and concerns potentially affecting the software product. Both the questionnaire and the application process have been developed using extensive expertise and have been extensively evaluated [40].

Continuous Risk Management (CRM) is a principal based way of handling the risks and opportunities during the software development life cycle. It controls the management of risks regardless of the tools and techniques to be used [162].

Team Risk Management (TRM) extends risk management with team oriented activities, involving both the team of customers and the development team [27, 90].

Risk Clinic is a type of workshop that combines the software CRM and Team Risk Management (TRM) together and aligns them with the communication, project management and risk management channel of the client/user [90, 148].

Survey, Questionnaires and Interviews they provide a way of direct communication with the customer and can identify risks in very short time [90, 162].

Some of the above approaches are used during the application of our contributions in the healthcare domain and are further described in chapter 8.

2.6 Conclusion

Although software cost estimation and risk identification are hard challenging processes and numerous approaches from a variety of research areas attempt to tackle it, still we could find none that is suitable for the very early stage of the decision making process. In the early phase of the process, managers and stakeholders of large organisations need lightweight techniques that can quickly provide accurate enough information to make a defensible go / no go decision. There is no value in spending significant time and resources in accumulating detailed information for a solution that will then be rejected. What we need, is a lightweight and low-cost approach to quickly identify points of high cost and risk, big 'no's' that would jeopardise the effective realisation of the expected benefits of the new development.

Chapter 3

Literature Review

“Research is the joy of discovering something no one knew before.”

Stephen Hawking

3.1 Setting the scene

In the previous chapter we explored existing techniques and tools in identifying high cost and risk in software developments and demonstrated the need for a lightweight approach that does not require extensive time and resources. Identifying cost and risk in new developments is hard, especially in large, complex organisations. Large complex organisations can be viewed as networks of producers and consumers. Producers create information, and consumers use the created knowledge to create some value to the organisation. Both producers and consumers can be people, or software systems.

However, new functionality is often needed to be integrated on to the existing networks of people, systems and organisations. Adding new functionality to an already crowded infrastructure, it can cause change in the existing network. Existing information created by a producer in the infrastructure is needed by new consumers, such as the new software functionality to be added to the network. This change in the network requires the movement of existing information to new places, new information flows to be created, and existing ones to be altered. Network changes come with challenges. For example, moving existing data to a new context other than the one it was originally designed for, introduces the risk of data loosing their original meaning and interpretation.

In this chapter, we review the literature to investigate challenges and issues of introducing new functionality in a large, complex network of people, systems, and organisations. We explore potential solutions and existing approaches on mitigating cost and risk when embedding new functionality to an already crowded infrastructure. In particular, in this literature review we address the following review questions:

- What factors affect data movement in large, complex IT systems that can introduce high cost and risk to an organisation?
- What are the challenges of achieving data interoperability in large complex organisations?

We begin by providing the methods we followed to review the literature emphasising on the search strategy and data synthesis approach we followed (section 3.2). Then we present the results of our review structured on the three broad themes we retrieved from the data analysis: data movement (section 3.3), data modelling (section 3.4), information portability (section 3.5), and data interoperability (section 3.6).

3.2 Methods

In order to address above literature review questions we devised a search strategy inspired by the Preferred Reporting Items for Systematic Reviews and Meta-Analyses (PRISMA) statement [118]. Our search strategy follows the phases of: (1) Identification of related publication papers, (2) Screening of papers, (3) Eligibility of included papers, (4) Synthesis of included results.

We began the **identification phase** of our strategy by identifying the information sources we will be retrieving our data from. We searched the following information sources for relevant publications: ACM Digital Library, Google Scholar, SCOPUS, and PubMed.

We then defined a set of terms to be used in order to identify related publications. Since we are looking for challenges and issues in large organisations that are caused by adding new functionality in the existing infrastructure, we searched the aforementioned databases with the key terms of: information sharing, data movement, information exchange, data interoperability, semantic interoperability, information portability, software portability, information movement, data transfer, data models, and modelling techniques.

Above terms were first used as stated to get a feel of the state-of-the-art in these areas. Then, we used above terms in combination with domain terms to retrieve current problems, potential solutions, and existing challenges. We used the following terms: issues, challenges, problems, cost and risk, high costs, solutions, models, health, healthcare, NHS, and standards.

Next, we proceed to the **screening phase** of our strategy. Searching the databases with above terms, resulted to a big number of publications. To limit the number of retrieved publications we defined a set of inclusion criteria that define the scope of our literature review. These are the following:

- **Study focus:** We are looking for approaches that are suitable to be used at an early stage in a project. We need a lightweight, low-cost approach suitable for early stage decisions that can estimate potential cost and risk of a planned development. We are looking for papers that capture challenges and/or solutions of data movement and interoperability with the following characteristics:
  - **Complexity:** An approach that examined both the data movement within and between systems and organisations to allow us to identify points of cost and risk of planned new IT developments that are often part of bigger, more complex eco-systems that need to share information from collaborating organisations.
  - **Socio-technical environment:** The approach must give equal prominence to both social and technical factors affecting the data movement and data interoperability when developing a new IT development.
  - **Lightweight:** The approach must be sufficiently lightweight and low-cost to be used as a decision-making aid in the early stages of a development cycle, ideally before any implementation is initiated.
- **Study design:** We included studies of any type of study design. We also included non-empirical studies (i.e. theoretical papers and literature reviews).
- **Publication types:** We included papers with the following characteristics:
  - Full publication papers.
  - English-written papers, regardless of the original language of research.

Publication papers were first screened by titles and abstracts. Selected papers were then screened for inclusion by reading the texts. Lastly, we reviewed the reference lists of included papers searching for further eligible papers.

Finally, we moved on to the last phase of our strategy, the **synthesis of results**. To do so, we followed an approach adopted by Thematic Analysis and particularly on the approach described by Braun *et al.* [35]. We followed the below steps to synthesise our results:

1. Familiarise with data: read included papers, noting down initial ideas.
2. Generate initial codes: coded interesting features of the papers across the entire dataset, and collated papers relevant to each code.
3. Search for themes: collated codes into potential themes, and gathered all papers relevant to each potential theme.
4. Review themes: checked if the themes work in relation to the coded extracts and the entire dataset, and generated a thematic map of the analysis.
5. Define themes: on-going refinement of each theme, and the overall analysis story.
6. Report of findings: selected compelling extract examples, analysed and related back to the pre-defined literature review questions.

In the following sections we present our findings based on the three broad categories we identified through our analysis: data movement, data modelling, information portability, and data interoperability. Figure 3.1 shows the thematic map produced capturing both level one and two of the identified themes.

3.3 Data Movement

Addressing the first literature review question we look for research studying the effects of data movement in large, complex organisations. As discussed earlier, there are several forms of data movement in large, complex organisations: data movement between IT systems, between people, and between organisations.

Data movement between IT systems can be characterised by three fundamental components: (1) retrieval of relevant data from the system's disk or storage area into

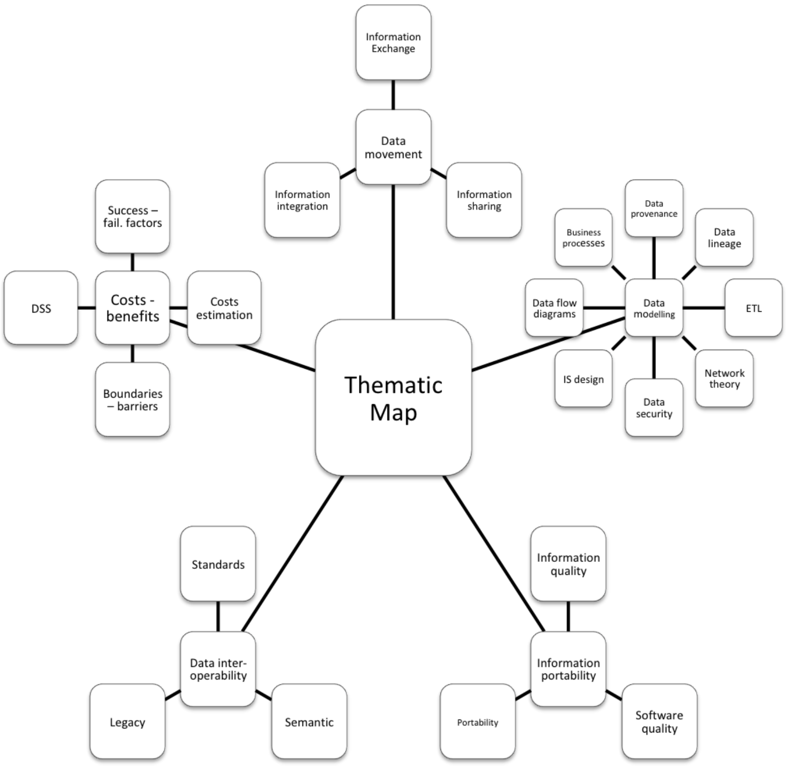


Figure 3.1: Literature review thematic map including level one and two themes.

the system's application memory, (2) data movement across the network, and (3) end-to-end data transfer from the application conducting the transfer.

Chang *et al.* mention that while large data movement, is a key operation in many applications and operating systems, contemporary systems perform this movement inefficiently, by transferring data from the memory to the processor, and vice-versa, across a narrow expensive channel [41]. To fix this, they propose a new Dynamic Random-Access Memory (DRAM) design, called Low-Cost Inter-Linked Subarrays (LISA), to enable fast and energy-efficient data movement across a large range of memory at low cost.

Bouremand *et al.* observed that data movement between the main memory system and computation units (e.g., CPUs, GPU, and special-purpose accelerators) is a major contributor to the total system energy and execution time, making the movement of data an expensive task [29]. This is because the energy cost of moving data is orders of magnitude higher than the energy cost of computation [104] Realising that data movement is a bottleneck they propose a new way of processing-in-memory (PIM) that can significantly reduce data movement, by performing part of the computation close to memory.

There is extensive research in the literature focusing on Near Data Processing (NDP), a new paradigm that brings computation to data, instead of bringing data to computation [168, 78, 119]. However, we can not always avoid the movement of data. There are cases that a data movement is required between organisational teams; the data need to travel from the memory to the application and through a network connection to the system of the other organisational team.

Often, an organisational team needs to create a new system. Embedding new functionality into an existing network of systems requires the movement of data from several sources into the new software system. For the new system to make sense of data coming from different and often heterogeneous sources data need to be integrated. Data integration is a well-reviewed topic in the literature. However, some challenges still remain.

Integrating the large structured data repositories in a large organisation, which are often scattered across many data silos, requires building end-to-end scalable workflows. Current semi-manual and rule-based systems simply do not scale and cannot accommodate the continuously growing data across the various business units. Several articles investigate the use of machine learning techniques to solve such kinds of challenges in data integration [21, 55, 63, 117]. However, Stonebraker *et al.* mention that while machine learning techniques is a way to go, multiple practical considerations arise, such as, the scarcity of training data, the need to explain the results to business owners, and the high cost of involving domain experts [166].

Despite a large literature on data movement challenges and protocols within a software system, we found little evidences of literature on movement of data across systems and specially between people and organisations. Existing literature that contains the term ‘data movement’ is primarily technical in nature. It focuses on the movement of data within a software system between the memory and application layer, and sometimes on data moved between software systems through a network connection. Data movement in high-performance computing (HPC) is well covered, but not as well in data movement between enterprises and organisations. Additionally, in large organisations and enterprises data are not only in electronic form. We could not find literature sources exploring challenges of data movement where data are in physical form. Finally, using the terms we defined earlier, we found no papers discussing or addressing data movement challenges between organisations.

3.4 Modelling Data Movement

A plethora of modelling techniques and notations have been proposed for use during information systems design, some of which include elements of the movement of data. In this section, we survey the principal modelling techniques, to see if any meet our inclusion requirements (given in section 3.2).

A number of software design techniques allow modelling of data movement from a technical point of view. Data flow diagrams (DFDs) are the most directly relevant of these [50]. Unfortunately, the focus in DFDs is on fine-grained flows, between low-level processing units, making it hard to capture higher-level aspects of the enterprise that can bring cost and risks, i.e. the social factors [14, 181]. Similarly, the Unified Modelling Language (UML) contains several diagrams detailing movement of data, notably sequence diagrams, collaboration diagrams and use case diagrams [156].

Although the abstraction level at which these diagrams (DFDs and UML) are used is in the control of the modeller, to an extent, they provide no help in singling out just those elements needed to identify cost and risk of a potential development. Also, social factors influencing information portability and introducing cost to the movement are not part of the focus of those approaches. These models are helpful in designing the low level detailed data flows within a future development, but can not help us decide which flows may introduce cost and risk to the development.

Other techniques are able to model high-level data movement between systems and organisations. Information Flow Diagram (IFD) models, for example, capture the flow of information between the internal departments and sub-systems [123, 137]. Although an IFD can be used to model the information flows within and between organisations, and we can get a valuable insight on the movements of data happening in large IT systems, the type and format of information that is used in the IFD and the mechanisms by which it is conveyed are not specified [12].

Data provenance systems log the detailed movement of individual data items through a network of systems [164]. While these logs can be a useful input to data journey modelling, they describe only the flows that are currently supported and that have actually taken place. They are not suited to modelling potentially new flows, and do not directly help us to see what social and organisation factors affect the flow.

Business process modelling (BPM) captures the behaviour of an organisation in terms of a set of events (something happens) and activities (work to be done) [2]. Although BPM can implicitly model flows of data between a network of systems, they typically contain much more detail than is needed for our purposes drifting away from the lightweight criterion.

Enterprise modelling (EM) is commonly regarded as the construction and use of conceptual models to describe, analyse, and redesign different aspects of an organisation [24]. Examples of enterprise modelling methods are: ArchiMate [86], Architecture of Integrated Information Systems (ARIS) [161], Business Engineering (BE) [182], Design and Engineering Methodology for Organisations (DEMO) [60, 61], For Enterprise Modelling Method (4EM; formerly Enterprise Knowledge Development) [153, 158], Multi-Perspective Enterprise Modelling (MEMO) [76, 74, 75], Semantic Object Model (SOM) [71], The Open Group Architecture Framework (TOGAF) [101], and Work System Theory (WST) [5]. DEMO concentrates on a few elementary organisational aspects and focuses on engineering an organisation (its structure)[8, p. 74]. According to Dietz *et al.* this aspect is not yet raised by other methods [61]. On the other hand, MEMO argues that an engineering point of view is not sufficient for shaping organisations, since organisations can also emerge through social construction and not just from engineering aspects [76]. However, although all the above models support organisational redesign efforts they do not implicitly capture the movements of data happening in an organisation.

Models that combine technical and social information, such as human, organisational, governance and ethical factors, can be found in the literature [140, 186, 187]. For example, the i* modelling framework aims to embed social understanding into system engineering models [186]. The framework models social actors (people, systems, processes, and software) and their properties, such as autonomy and intentionality. Although a powerful mechanism to understand actors of an organisation, the i* framework does not give us the information flows happening between the systems, nor any measures to identify costs.

In summary, we found no existing model that met all our requirements. The existing technical modelling methods give us a way to model the detail of a system to be constructed, but provide no guidance to the modeller as to which parts of the system should be captured for early-stage decision making and which can be safely ignored. The socio-technical models allow us to capture some of the elements that are important in identifying cost and risk in IT developments, but need to be extended with elements that can capture the technical movement of data.

3.5 Information Portability

Data created in one place, for use by one specific set of consumers, is rarely fit and ready for use by the range of secondary consumers that arise over its lifetime. This ability of a piece of data to survive being moved from one system to another is an aspect of information quality (IQ) that is typically referred to by the term "information portability". Information portability is the ability of data to survive movement to a context other than the one it was originally designed for. For example, if we consider a dataset that contains information about dates, we can see that the data entity "1/2/2017" is less portable than the data entity "1st February 2017". This is because different conventions on how to write dates in the US and Europe (for instance) make the meaning of the first value ambiguous. Does it refer to 1st February or 2nd January? If the data value is created in Europe, then it is likely to be viewed as quite portable if moved from one European system to another. It should be viewed as being much less portable when moved from Europe to a US-based system, where the local conventions will cause anyone unaware of the source of the data, or of the different conventions in writing dates, to interpret the data incorrectly. Work has to be put into reformatting the dates on transfer, if this risk of misinterpretation is to be removed.

Although a large literature on information quality exists, there is very little research on the portability of information and the factors limiting or enhancing the sharing of information with other systems. While Wang *et al.* identify portability as an information quality dimension, they do not provide a definition for it [179]. Others use the term portability in the context of system independence from the format of the data, but not in the context of information [95, 152]. At the time of writing, we were not able to find a complete definition nor practical measures to identify challenges that could impose cost and risk of information portability in potential developments in the research literature.

In this section, we review the literature of information quality looking for definition and measures of information portability that can help us identify cost and risk of developing new information sharing developments.

Despite a large literature on IQ, and the inclusion of the information portability concept in several well known taxonomies of information quality dimensions [120, 135, 152, 179], we could find very little research specifically on the topic of information portability. Where it is mentioned, it is as one of numerous other IQ properties, like accuracy and completeness, rather than as the focus of the paper. Where it is defined, the definition given is high level and does not go beyond what might be reasonably inferred by common sense from the term itself.

Wang *et al.* identify portability as an information quality dimension, but do not provide a definition for it [179]. Redman *et al.* and Jarke *et al.* use the term portability in the context of systems, and not information [152, 95]. The former describe it as system independence from the data format, while the latter as system independence of the software managing the data.

Knight *et al.*[106] and Parker *et al.* [143] compare IQ taxonomies found in the papers of Van Zeist *et al.* [174] and Leung *et al.* [115], in which the term portability is included as a dimension of data quality. However, the definition of the portability term used in the latter papers is derived from the British Standard (BS) ISO 9126 on software product quality [169]. The standard identifies portability as a characteristic of software quality rather than information quality and defines it as: "The capability of the software product to be transferred from one environment to another." ([169], p. 11)

A further definition and probably the most helpful, is given in BS ISO / IEC 25012:2008 standard on software engineering and data quality [170]. It categorises data portability as one of the fifteen quality characteristics that constitute the data quality model in software engineering. The definition states:

> "Data portability is the degree to which data have attributes that enable them to be installed, replaced or moved from one system to another preserving the existing quality in a specific context of use." ([170], p. 10)

Finally, although information portability is mentioned in the literature as a dimension of IQ taxonomies, we do not have a complete definition nor practical measures to identify challenges and costs of information portability in potential developments. However, to test our hypothesis that data movement can be an early warning signal of failure we need a definition to refer to information portability. For the rest of this thesis, we adapt the above definition provided by the ISO standard, and define information portability as follows:

Definition 3.5.1. Information Portability: Information portability is the ability of data to survive movement to a context other than the one it was originally designed for, sustaining its initial quality before the movement.

3.6 Data Interoperability

Numerous definitions have been given for data interoperability in the literature. The authoritative dictionary of IEEE standards terms provides the following four definitions of interoperability: (1) "The ability of two or more systems or elements to exchange information and to use the information that have been exchanged"; (2) "The capability for units of equipment to work efficiently together to provide useful functions"; (3) "The capability promoted but not guaranteed achieved through joint conformance with a given set of standards, that enables heterogeneous equipments, generally built by various vendors, to work together in a network environment"; (4) "The ability of two or more systems or components to exchange and use the exchanged information in a heterogeneous network" [36].

Major issues in collaboration and co-operation of enterprises, businesses and organisations are the problems in communication between people, between people and systems and between systems. Kosanke states that the first, people communication issues, is usually due to different cultures, languages and even professional jargons and can be addressed by either a common language (which is very unlikely) or through translations between the different languages and meanings [108]. The other two problem areas originate from the different software and hardware implementations of systems. These areas require both human and machine understanding of the exchanged information. As a result, the ability to interoperate is a rather complex task. According to Chen *et al.* who reported on the results of a European initiative on interoperability, there exist several levels of interoperability [47]. We summarise them below:

- **Technical Interoperability** is achieved among software systems when services or information could be exchanged directly and satisfactorily between them and their users [108, 139]. Technical interoperability is typically associated with hardware/software components, systems, and platforms that enable machine-to-machine communication. This type of interoperability often focuses on communication protocols and the infrastructure required for those protocols to function [176, 112].

- **Syntactic interoperability** is defined as the ability to exchange data. Syntactic interoperability is generally associated with data formats. The messages transferred by communication protocols should possess a well-defined syntax and encoding, even if only in the form of bit-tables [43].

- **Semantic interoperability** is defined as the ability to operate on that data according to agreed-upon semantics [116]. Semantic interoperability is normally related to the definition of content, and deals with the human rather than machine interpretation of this content. Thus, interoperability at this level denotes that a common understanding exists between people regarding the definition of the content (information) being exchanged [84, 176].

- **Organisational interoperability** describes the capability of organisations to effectively communicate and transfer meaningful data despite the use of a variety of information systems over significantly different types of infrastructure, possibly across various geographic regions and cultures. Organisational interoperability relies on the successful interoperability of the technical, syntactic, and semantic aspects [176].

In the healthcare domain, interoperability is defined by the Healthcare Information and Management Systems Society (HIMSS)[1] as “the ability of different information technology systems and software applications to communicate, exchange data, and use the information that have been exchanged” [19]. Lumpkin [121] provides three levels of health information technology interoperability in health:

Foundation interoperability enables data sharing between two systems, but does not require the target system to interpret the data.

Structural interoperability provides uniform movement of healthcare data between two systems so that clinical or operational purpose and meaning of the data is preserved. It ensures that data exchanges between information technology systems can be interpreted at the data field level.

Semantic interoperability enables two or more systems to exchange and use information. It takes advantage of the structuring and the codification of the data so that the target system can interpret the clinical meaning of the data. This level of interoperability supports the electronic exchange of patient summary information among caregivers [83].

Despite the increasing recognition of the importance and potential benefits of interoperability, significant challenges remain. The challenges in achieving semantic interoperability between two information systems transcend the technical issues. There is a

[1] http://www.himss.org/library/interoperability-standards/what-is-interoperability

variety of cultural, social, political and economical barriers to data exchange [121].

Efforts have been put in achieving interoperability in the NHS by defining standards that give structure to information created in healthcare. By doing so, information can be shared consistently within and across healthcare settings. Most importantly, standardising captured information enables exchanged information to be interpreted by both the producers and the variety of consumers of health data.

At the time of writing, there are three information standards in the NHS, the SNOMED CT, ICD-10, and OPCS-4. All three standards are a national requirement, they serve different purposes and are complimentary to each other. We investigate each in turn.

SNOMED CT is an acronym for Systematised Nomenclature of Medicine Clinical Terminology. It is the vocabulary for use by clinicians in an Electronic Patient Record (EPR) and is recognised internationally. It records consistent, reliable and comprehensive patient information as an integral part of an EPR. It is recorded at the point of patient care and focuses on what a clinician wants to record about their patient. SNOMED CT facilitates a number of processes such as decision support, care pathway management and drug alerts.

OPCS-4 and ICD-10 support secondary uses of data for statistical purposes, such as operational and strategic planning, epidemiology, public health analyses of population health and reimbursements. OPCS-4 and ICD-10 are recorded after a healthcare event. They are applied in accordance with business rules and focus on the information we need to capture in order to count for statistical and epidemiological analyses. The mandatory ICD-10/OPCS-4 data collected in Commissioning Data Sets (CDS) must comply with classification rules and principles, such as national clinical coding standards.

One of the challenges of achieving full semantic interoperability is that each foundation trust will differ in the way that they adopt and use the Electronic Patient Record (EPR) and SNOMED CT. It is likely that the role of the coder will evolve as EPRs become more common and as SNOMED CT and future classifications products (such as ICD-11) are implemented [111]. Also, in primary care electronic records exist, but use Read codes; the standard clinical terminology system used in general practice to support detailed clinical encoding of multiple patient phenomena. Having two terminologies in place, causes text degrades and lower data quality when data are exchanged. To address incompatibility issues, NHS provides classification maps that link codes from clinical information recorded by the clinician in the EPR using SNOMED

CT to ICD-10 and OPCS-4 codes. The classification maps are designed to assist coders in the accurate and timely assignment of classification codes in the coded record, using structured clinical information from the EPR. However, utilising maps from SNOMED CT for classification coding is not a fully automated process, and requires expertise in the application of the classifications.

Health Level 7, as mentioned earlier in section 2.2.2, is a not-for-profit standards development organisation that was established in 1987 to develop standards for the exchange of the health information and health systems interoperability in the hospitals FT information systems. HL7 provides electronic system to system messaging standards, as well as standards for electronic document structure and content standards [17].

The messaging standards are available as HL7 v2 and HL7 v3. HL7 v2 was initially developed in an ad hoc fashion to integrate hospital systems together, such as administrative and clinical systems [17]. Although HL7 v3 was published as an improved version of HL7 v2, it has some shortcomings. According to Mead *et al.*, the published HL7 v3 standards are not directly implementable and require substantial tooling for the generation of executable software systems [131]. In order to implement HL7 v3 one needs an in depth and detailed understanding of the Reference Information Model (RIM). RIM defines the structure of the semantic and lexical elements of HL7 v3. One of the biggest issues with HL7 standards, as stated by Bender *et al.*, is that HL7 v2 and v3 are not directly interoperable between them; information exchange between v2 and v3 requires the use of sophisticated translation software [17].

Sartipi *et al.* explores the challenges of applying the HL7 v3 standard message development platform to integrate legacy healthcare information systems [160]. They achieved semantic interoperability of a Clinical Decision Support System (CDSS) with the EMR of a specialist. Jayaratna *et al.* use semantic web technology to develop a tool, called TAMMP, to identify HL7 messages that perform healthcare transactions in an integration scenario [96]. Dehmoobad *et al.* propose an expansion to the core HL7 messages to provide cross-domain interoperability between healthcare and insurance organisations [159]. As an attempt to solve some of the aforementioned issues, HL7 has recently developed a new standard referred to as Fast Healthcare Interoperability Resources (FHIR). FHIR was developed in an incremental, iterative approach to reflect today's industry best practices for complex systems design [17].

Another challenge of data interoperability in health is the myriad of legacy systems that exist in the healthcare infrastructure. According to Bennet *et al.*, legacy systems can be described as large software systems that are vital to an organisation, and have been implemented years ago [18]. Legacy software was written using outdated tools and techniques, yet it continues to do useful work. Achieving interoperability between legacy systems comes with technical and nontechnical challenges, ranging from justifying the expense in dealing with outside contractors to using program understanding and visualisation techniques [22].

Finally, information portability is one of the most important characteristics of modern healthcare. The benefits of interoperability are numerous and can transform patient care. Effort has been put towards achieving interoperability and tremendous resources have been invested to date by industry and healthcare programs around the world. However, although information standards have been established and used across the NHS and the world, interoperability issues still remain making information interoperability a big challenge in the healthcare community.

3.7 Conclusion

In this chapter, we reviewed the literature on challenges and issues of achieving data movement and data interoperability in large complex information systems that can be viewed as complex networks of people, systems, and organisations. We described the search strategy and the methodological synthesis approach we followed and then presented our results.

Chapter 4

Research Methods

4.1 Introduction

In this chapter we discuss and justify the methodological approach we followed in this thesis in order to address the research questions provided in section 1.4. We also provide the reasons why our methodological approach is best suited in achieving the stated research aims. We further discuss why applied methods were chosen in contrast to alternative approaches. We organise this chapter based on the data collection methods (section 4.2), data analysis methods (section 4.3), development methodology (section 4.4), and evaluation methods (section 4.5) we followed.

4.2 Data Collection Methods

For our data collection phase, we use qualitative methods aiming to capture complex textual descriptions of how people experience IT developments and the challenges they face when using them in their everyday roles. There is a pool of qualitative methods to be used:

- **Observation:** Participant observation is appropriate for collecting data on naturally occurring behaviours in their usual contexts.

- **Interview:** In-depth interviews are optimal for collecting data on individuals personal histories, perspectives, and experiences, particularly when sensitive topics are being explored.

- **Focus groups:** Participatory groups are effective in eliciting data on the cultural norms of a group and in generating broad overviews of issues of concern to the cultural groups or subgroups represented.

- **Ethnography:** Goodson *et al.* describe ethnography as a social research method occurring in natural settings aiming to learn the culture and behaviour of the group under study [79]. Ethnography uses observation techniques and interviews to collect data.

- **Case Studies:** Case studies are up-close, in-depth, and detailed examinations of a subject of study and its related contextual conditions [77].

Throughout the course of this project we were fortunate enough to have access to case studies written by NHS staff taking a professional development course at the University of Manchester. The case studies were written as assessed coursework for a course on "Human and Organisational Factors in Health", offered as part of the University of Manchester's continuing and professional education provision, during the 2013 academic year.

The case studies describe a variety of IT developments in the NHS, covering cancer care, ambulance service management, in-patient management, heart failure care, diabetes care, bed management and more. The case study authors were asked to describe a recent IT development they had experience of in their working role. They were asked to categorise the development as having been either successful or unsuccessful, and to identify the human, organisational and technical factors that contributed towards that particular outcome.

As with many governmental institutions, aspects of the case studies are confidential. They describe existing systems and current processes of established Foundation Trusts. They also contain personal views of the authors and factors that caused the failure of often very expensive IT developments. Hence, while we can share the results of our examination and analysis (with any identification details removed) we can not report any confidential raw data.

4.3 Data Analysis Methods

Having collected the case studies, we followed an approach inspired by thematic analysis to analyse the information captured in the case studies. Thematic Analysis is a qualitative method that emphasises on pinpointing, examining, and recording themes within data [82]. Themes are patterns across a dataset that are important to the description of a phenomenon and are associated to a set of preset research questions. According to Dixon *et al.* and Mays *et al.* thematic analysis has been identified as one of the main approaches used to review and synthesise qualitative and quantitative evidence [62, 129]. The research questions (as retrieved from section 1.4) to be addressed in this thesis by analysing the case studies are:

- RQ 1: What factors contribute to the failure of a new IT development that can impose high cost and risk to the organisation?
- RQ 2: Are there any early warning signs of high cost and risk when developing a new IT system?

In order to address the first research question and identify the factors that contribute to the failure of recent IT developments we examined the collection of NHS case studies, first to categorise each one as a successful or a failing development, and second to understand the major root causes of failure in each case study.

To do so, we devised a strategy inspired by the inductive method [1] of thematic analysis in which themes occur after analysing the data, and particularly on the approach described by Braun *et al.* [35]. According to Guest *et al.* the advantage of using an inductive approach in finding themes is that the data are not overfitted into pre-defined models, but themes occur naturally from the data [82].

More specifically, we took the following strategy in extracting, analysing and using the data from the case studies:

1. **Data familiarisation:** familiarise with the data by reading all case studies.
2. **Data extraction:** extract from the case studies the relevant information into data extraction sheets. Prior to the actual extraction process, define what is to be extracted and in which form.
3. **Priori grouping:** group information from the data extraction sheets according to the review questions.

[1] Thematic analysis can be achieved by two approaches: the inductive and the deductive. Using an inductive approach, coding of the data into themes occurs without having a pre-defined model ensuring that identified themes are derived naturally from the data [82]. On the other hand, a deductive approach is theory-based. This form of analysis tends to be less descriptive because analysis is based on a predetermined model.

4. **Extraction and initial coding:** based on priori grouping, define further extraction details needed to address the research questions and extract the data from the case studies. Use codes to mark the data to then identify themes.

5. **Searching for themes:** once all extracted data are coded, sort out codes into broader, more conceptual categories to create themes.

6. **Reviewing themes:** review the themes in which extracted data lie into to verify relativity and determine whether the derived themes satisfactorily capture the extracted data.

Following the above steps, we extracted and organised the failure factors identified by the authors, and aggregated the results across the full set of case studies. Identifying the list of failure factors was a relatively trivial task in our case, as the authors of the case studies had been explicitly asked to identify and record the factors, social and technical, that contributed to the project outcome. We had only to extract them, to integrate them into a common set and to regularise the terms used.

In order to address the second research question and identify early warning signs of high cost and risk we followed our strategy once again to identify problematic types of data movement. This was more challenging, as the authors had not been asked to characterise the data movements involved. We had to infer the presence of movement patterns from the explanation in each case study. Again, we had to combine and regularise the patterns from different studies to create a consistent set. To do so, we followed a similar approach as before: data familiarisation, data extraction, initial coding, searching for themes, and finally reviewing themes.

Further details on how the aforementioned strategy has been applied to analyse the collected data are given in sections 5.2 and 5.3 where we describe how we used the strategy to retrieve IT failure factors and the data movement anti-patterns.

4.4 Model Development Methodology

Our research question to be answered by this method is:

- RQ 4: How can we quickly identify the warning signs of high cost/risk in the aforementioned abstract model of the new system?

In order to answer the above research question, we followed an agile-inspired approach. The agile approach is primarily used in software development in which user requirements and solutions evolve through the collaborative effort of the development team and the end users or customers. The agile approach to software development is getting more attention and is adapted to a wide range of product development, not just for software development. This is because the approach advocates adaptive planning, evolutionary development, early delivery, and ongoing improvement, while encouraging a rapid and flexible response to change; a set of practices that aim to ensure the continuous delivery of working product.

The four core values of Agile software development as stated by the Agile Manifesto [2] are: individuals and interactions over processes and tools; working software over comprehensive documentation; customer collaboration over contract negotiation; and. responding to change over following a plan [4].

Compared to traditional model development approaches, such as the waterfall approach, agile development mainly targets complex projects and product development with dynamic, non-deterministic and non-linear characteristics [54]. The waterfall approach depends on accurate estimates, stable plans, and predictions which are often hard to get right in the early stages of a project. Requirements change and design is held to be emergent. Big up-front specifications are often become outdated the minute they are drafted. These, among other reasons, have helped shape agile development's favor of adaptive, iterative and evolutionary development.

In this thesis, in order to develop our model we followed a method inspired by the iterative approach of the agile development methodology. We particularly developed our model in iterations of Plan - Act - Reflect cycles. The iterative approach we followed focuses on an initial, simplified implementation which then progressively gains more complexity and a broader feature set until the final model is complete.

During the planning phase of our iteration approach, we assess the current state of the model and set initial requirements (or improvement requirements) and measurable and attainable goals based on the outcomes of the reflect phase of the previous iteration (apart from the very first iteration). As part of this phase, we aimed to plan small, clear and concise steps so that they can be easily monitored, and tested. The acting phase of our approach involves implementing the requirements and goals set in the planning phase.

One of the main differences between the iterative agile development methodology and the waterfall method is the approach to quality and testing. In the waterfall method, testing of the model always happens after all the implementation has been

[2] http://agilemanifesto.org/principles.html

finished. However, in agile development testing is completed in the same iteration as implementation. In our approach we have embedded testing in the reflection phase of our iterations where we reflect on the methods used, model produced and incorporate feedback from the end users. Findings and decisions from the reflection phase act as input to the planning phase of the next iteration.

Finally, as one of our criteria in developing the model is to be lightweight without requiring substantial resources and time, it proved natural to follow an agile methodology which emphasises on the principles of minimum valuable products (MVP) and simplified functionality.

4.5 Model Evaluation Methods

In order to answer the last three research questions of this project we have to evaluate the produced model in terms of whether it can accurately and cheaply identify points of high cost and risk in a planned new IT system. Our research questions to be answered by the methods described in this section are:

- RQ 5: Can the warning signs accurately and cheaply identify points of high cost and risk in a planned new system?
- RQ 6: Can the warning signs identify points of high cost and risk of new systems, across domains?
- RQ 7: Do the warning signs identify all the significant points of high cost and risk of a new system?

Ideally, a research strategy to evaluate our model and answer our research questions would be to identify an organisation that is planning to introduce a new IT system in their infrastructure and follow the approach below:

1. Apply our model on the planned new IT system to identify places of potential high cost and risk and report back to them.
2. Establish two teams to develop the planned new IT system. Team A' develops the IT system as originally planned, whereas team B' takes into consideration our models findings and avoids implementing the identified high cost and risk places.

3. Run the two systems in parallel.
4. Conduct a cost/benefit analysis of the two systems developed.
5. Compare the results of the two developments to assess the validity of our model's findings.

Such a strategy is following a prospective evaluation approach. A prospective evaluation study watches for outcomes during the study period and relates this to other factors and parameters, such as suspected risk. The study usually involves taking a cohort of subjects and watching them over a long period of time [146].

However, the timeline and resources of this project are limited. We can not wait for a long time, often years, for the planned new system to be developed, integrated and run within the existing infrastructure in order to evaluate our model. Also, our resources are scarce and can not employ a team of developers to develop a mirror system taking into consideration our model's findings.

So, instead of following a prospective approach we used a retrospective approach to evaluate our model. A retrospective study looks backwards and examines exposures to suspected risk or protection factors in relation to an outcome that is established at the start of the study.

To do so, we worked with staff of organisations that have recently introduced new IT systems in their existing infrastructure. We carried out a retrospective analysis, following the re-engineering phase the organisation has executed, using the following approach:

1. **Before** the re-engineering phase:
    (a) Retrospectively model the data journeys existing before the re-engineering.
    (b) Apply our model to identify places of high cost and risk.
2. **After** the re-engineering phase:
    (a) Model the data journeys made after the re-engineering exercise.
    (b) Compare our findings with the changes made by the re-engineering team, to address points of high cost identified through their own separate analysis.

By doing so, we can compare our model's findings with what human experts identified as costly and risky parts that have been removed during the re-engineering exercise. However, this approach of evaluation comes with some shortcomings. When comparing our model's results, we assume that the human experts conducting the re-engineering phase have correctly identified a set of cost and risk improvements in the old system. However, we also assume that there are other places of potential cost savings that were not changed by staff for various reasons, such as limited time and resources.

In order to address the above limitations of the retrospective approach, we contacted domain experts that know the specifics of the organisations involved in our evaluation to assess the feasibility of our model as a second layer of approval. We only consider valid model findings the places that have been removed in the new system after the re-engineering exercise and have been approved by the domain experts.

However, there is still the possibility that organisations have not gone through major re-engineering exercises in the recent past but have identified opportunities for improvement. For these evaluation studies we followed an approach inspired by action research. Action research is research initiated to solve an immediate problem or a reflective process of progressive problem solving led by individuals working with others in teams or as part of a "community of practice" to improve the way they address issues and solve problems [81]. Meyer states that action research focus on developing solutions to practical problems, while its main strength is the ability to empower practitioners, by getting them to engage with research and the subsequent development or implementation activities [133].

Action research follows an approach of spiral, self-reflective cycles of plan-act-reflect [110]. The cycle begins by planning a change, then act and observe the process and consequences of the change, reflect on the processes and consequences and then re-plan accordingly. Then, the next cycle begins with acting, observing and reflecting, and so on [109].

To do so, we worked with teams that have identified problems with their existing systems and would like to change their processes to reduce cost and risk. We adapted the action research cycle approach to fit the circumstances of the opportunities we had to work with such teams, some of which arose at very short notice. We began by defining a procedure to follow, and ensure to only document domain expert responses, and not to lead them to answers we might have wished they had given. In each of the studies, we followed an approach similar to:

1. **Plan:** Get in touch with a team of domain experts that have identified opportunity for improvement in their existing system. We planned a set of activities each corresponding to a research question to be answered by this evaluation exercise.

2. **Act:** We asked them to model the existing system based on their interpretation and knowledge of how their system works.

3. **Reflect:** Domain experts reflected on what they think is causing high cost and risk.

4. **Plan:** Then plan the next iteration, and apply our model to identify places of high cost/risk, based on our model's findings.

5. **Act:** Compare domain experts' views on high cost/risk causes and model's findings.

6. **Reflect:** Reflect on the results of the comparison, the processes followed and consequences, and causes of differences.

We further describe the particulars of each evaluation study in chapter 8 where we evaluate the application of our model in several NHS settings.

4.6 Conclusion

In this chapter, we explained and justified the methodological approach we followed. We discussed why it is suited in answering the research questions we set in the introduction (section 1.4) of this thesis. Finally, we discussed why these methods were chosen in contrast to alternative approaches.

Chapter 5

IT Failure and Data Movement

5.1 Introduction

As we have seen in the previous chapters, a lightweight approach is needed to identify points of cost and risk in new software development early in the decision making process. In this chapter, we describe our attempts to design an early warning indicator of cost and risk for use in large complex organisations. As a starting point, we analysed a set of 18 case studies of new software developments in the NHS. From the case studies, we extracted the factors that according to the authors, have contributed towards the failure of recently introduced developments. We found not just technical, but also a wide range of social factors: human and organisational in nature.

A common feature of the retrieved factors where the new software was deemed to have been unsuccessful was the movement of data. While seemingly simple in theory, in reality, the movement of data from system to system is beset by all manner of challenges, many of which are concerned with people than with technological issues.

These challenges lead to unforeseen costs and sometimes dramatic reductions in the benefits expected from the new software. We therefore hypothesise that identifying the need for movement of data in a new development could provide the early warning signs of success or failure that we were looking for.

In this chapter, we present the following contributions:

- A catalogue of factors contributing IT failure.
- A set of data movement anti-patterns that under certain criteria may introduce high cost/risk in new developments.

We begin by examining the collection of case studies from the NHS to retrieve factors contributing to the failure of IT systems (Section 5.2). Then, we analyse the case studies to extract problematic data movements, and finally we present a catalogue of data movement anti-patterns (Section 5.3).

5.2 Early Warning Signs of IT Failure

We need a collection of early warning signs for IT managers, that can indicate when the data movement required by a new IT development will come with cost and risk that may outweigh the expected benefits, unless they are taken into account in the proposed development.

To identify these early warning signs, and to seek evidence for or against our hypothesis concerning the role of data movement in IT system success, we examined 18 case studies written by staff in the NHS. The case study authors were asked to describe a recent IT development they had experience of in their working role. They were asked to categorise the development as having been either successful or unsuccessful, and to identify the human, organisational and technical factors that contributed towards that particular outcome. Overall, the case studies cover a variety of health care settings and functions, including ambulance service management, cancer care, electronic patient records and in-patient bed management.

The first step in the case studies analysis was to understand the factors that led to IT failure in each of the case studies, and to see if there were any commonalities that could suggest suitable early warning signs for IT failure in general. To do so, we followed the strategy introduced in section 4.3 on page 58 which is inspired by the inductive approach of thematic analysis. More specifically, we followed the following steps:

1. **Data familiarisation:** We familiarised with the data by reading all case studies.
2. **Data extraction:** We extracted from the case studies the relevant information into an excel data extraction matrix. Because of confidentiality reasons we can not publish the data extraction matrix, we do, however, list the types of details we captured for each of the case studies:
   (a) No: An itemised number to keep track of the case studies.
   (b) Development context (patient pathway): The pathway of the patient in which the new development is used, or was planned to be used.

   (c) Success / Failure / Partly Failure: Whether the described IT system was successful in realising the expected benefits, or not. In the case that only some of the expected benefits were realised we marked the case study as a partly failure.

   (d) Failure factors: A list of factors that according to the authors of the case studies contributed towards the failure of the IT development.

3. **Priori grouping:** Based on the data extraction matrix, we grouped the case studies into successful implementations of IT systems or not. We combined failed and partly failed case studies into one group, as both groupings are needed for further analysis of the failure factors.

4. **Extraction and initial coding:** We reviewed the case studies of the failure category again, and for each one we extracted the reasons that the authors of the case studies described were the roots of the IT system failure into a data extraction sheet. To do so, for each case study we did:

   (a) Highlighted in the text the failure factors.

   (b) Extracted each factor into an excel data extraction sheet removing any identification details.

   (c) Coded each factor using a restriction of several (five - six) words summarising the factor.

5. **Searching for themes:** Once all extracted factors were coded, codes were sorted into broader, more conceptual categories to create themes.

6. **Reviewing themes:** We reviewed the extracted factors and the themes they relate to, to determine whether the created themes satisfactorily capture the raw data.

7. **Data analysis:** Finally, we calculated the prevalence of each factor in the set of 18 case studies to identify the most common ones.

Of the 18 case studies, only three were described by their authors as having been successful. The remaining 15 were categorised as having completely or partly failed to deliver the expected benefits.

We identified 7 theme categories of failure factors: people-oriented factors, data-oriented factors, system-oriented factors, organisation-oriented factors, governance-oriented, requirements-oriented, and politically-oriented factors. Table 5.1 shows the factors that we found have contributed to the overall failure of the IT developments, as retrieved from the case studies. A brief description of each factor is given, along with the number of case studies in which the factor is mentioned as having played a role in the failure. In the table, the failure factors are grouped into the themes we identified which describe the nature of the factors.

Figure 5.1 (on page 72) shows the factors in decreasing order of occurrence, with the factors reported by most case studies shown on the left and those reported the least shown on the right.

Contributory Factor	Further Details	Case studies
	People-Oriented Factors	
Resistance to change in processes/ procedures	The introduction of new IT systems can require staff to follow new processes and procedures, which they may resist. Staff can also fear job losses and degrading of their roles.	14
Reluctance to engage with new technology	Staff may resist a change from familiar to unfamiliar technologies.	2
Varying IT literacy levels among staff	This factor can reduce or delay the benefits of the new IT systems, until the necessary skills are acquired.	4
Clash of grammars [175]	Specialist vocabulary used by one set of individuals (e.g., clinicians, the IT team) may not be completely understood by another set (e.g. clinicians in other departments, secretaries), leading to delays and data quality problems.	3
Lack of trust between staff	Staff often do not trust the IT implementation team and do not share knowledge, domain expertise and ideas with them.	3
Incorrectly identified stakeholders	Key users of the system were not identified as stakeholders and hence were not involve in any decision making (e.g. medical secretaries).	4
Insufficient stakeholder engagement	Communication and engagement may vary between stakeholder groups (e.g. secretaries, clinicians, GPs, developers).	6
Lack of shared vision	Lack of a shared vision between the stakeholders.	5
Insufficient resources for support and training	Lack of training and support for the users can cause resistance to use the new system.	5
	Data-Oriented Factors	
Numerous data sharing agreements	Exchange of information within or between organisations is controlled by multiple sharing agreements, each controlling a different, narrowly defined subset of clinical data (sometimes even within a single record).	1

Contributory Factor	Further Details	Case studies
Conflicting formats/ data structures	Data may need to be transformed before they can be used. Extraction of data from legacy systems in the required format may only be possible through the back end database, or following manual data transformation.	5
Disconnected data silos	Clinicians tend to trust their own legacy databases more than centralised shared ones. This can lead to duplicated, fragmented and inconsistent information.	5
Paper-based information	Paper-based information transfer is still widespread, and brings many additional costs. E.g. Patient clinical notes must be firstly physically found and then entered before the data can be reused.	2
System-Oriented Factors		
System extensibility not considered during development	Extending an existing system with extra functionality can require significant unforeseen re-configuration that adds to workload.	1
Over-optimistic system reuse	When a system designed for one organisational unit is adopted by another, considerable reconfiguration and process redesign is often required.	1
Inadequate system performance	System performance problems, especially over the long term, can have a de-motivating effect on users.	5
User-unfriendly system design	Often, new IT systems are not as easy to use as the users expect. They can be complex and inflexible. Systems may be wholly or partially unused, or workarounds may need to be developed.	4
Costly system sustainability and maintainability	Systems must support their users over the long term, in the face of requirements change, or become decreasingly useful.	1
Top-down system design	Managers may take decisions without consulting the people on the ground. The needs of specific regions or departments may not be considered.	3
Lack of required equipment	If insufficient or inadequate equipment is provided, stakeholders may not observe the value they expected from the new development.	4
Organisation-Oriented Factors		
Organisation has a complex structure	The NHS is not a single homogenised organisation but a coalition of many of individual organisations (e.g. Foundation Trusts, GP practises, commissioning groups, social care), each with its own politics, culture, technologies and structures.	2
Regulations	Organisational policies may conflict with the processes imposed by new IT developments, and vice versa, leading to reduced/delayed benefits and need for reconfiguration.	1
Organisational redesign	IT developments often demand some degree of organisational redesign (forming new teams, adopting new processes, etc.), which is time consuming and often unwanted by staff.	6

Contributory Factor	Further Details	Case studies
Resistance to collaborative projects	Staff may be reluctant to participate in collaborative IT projects, fearing loss of data/process ownership, control and influence.	2
Governance-Oriented Factors		
Governance	Risk-averse organisations resist sharing data with other organisations because of concerns regarding information and corporate governance, data ownership, privacy and confidentiality.	4
Responsibility	When a new integrated IT system is created, it may be unclear which organisational unit is responsible for developing/procuring and maintaining it. Ownership of the data may also be unclear.	1
Requirements-Oriented Factors		
Design-reality gap	The new IT system may not work as envisaged by its users, or may work in a way that contradicts processes and practises previously in use. When a new system interferes with the current working practises of staff, workarounds and shortcuts are devised.	8
Engagement	If the implementation team does not engage with clinicians and other key stakeholders to capture user requirements, the system functionality may not meet user needs.	2
Effect on quality of care	Clinicians can feel that the need to input data, or more generally interact with the new system cuts into time that would otherwise be spent focusing on the patient, and reduces the quality of care provided.	2
Politically-Oriented Factors		
Loss of political influence	A departmental unit may fear loss of influence and power when a new IT system is introduced.	3
Financial pressures	Projects are often expected to deliver benefits with a tight budget and limited/insufficient resources, leading to short-cuts that cause problems in the future.	4
Varying political power	Conflicting priorities on the requirements of IT systems by different agencies lead to conflicting requirements and unrealised benefits.	1

Table 5.1: Factors contributing to IT failure in the NHS case studies.

From figure 5.1, we see that the most common causes of IT failure in our case studies are related to people and how they interact in order to complete tasks. Some of the failure factors relate to technology, but these are fewer in number compared to the social (human and organisational) factors. Of the 32 factors identified, less than a quarter are primarily technical in nature: conflicting formats/data structures, disconnected

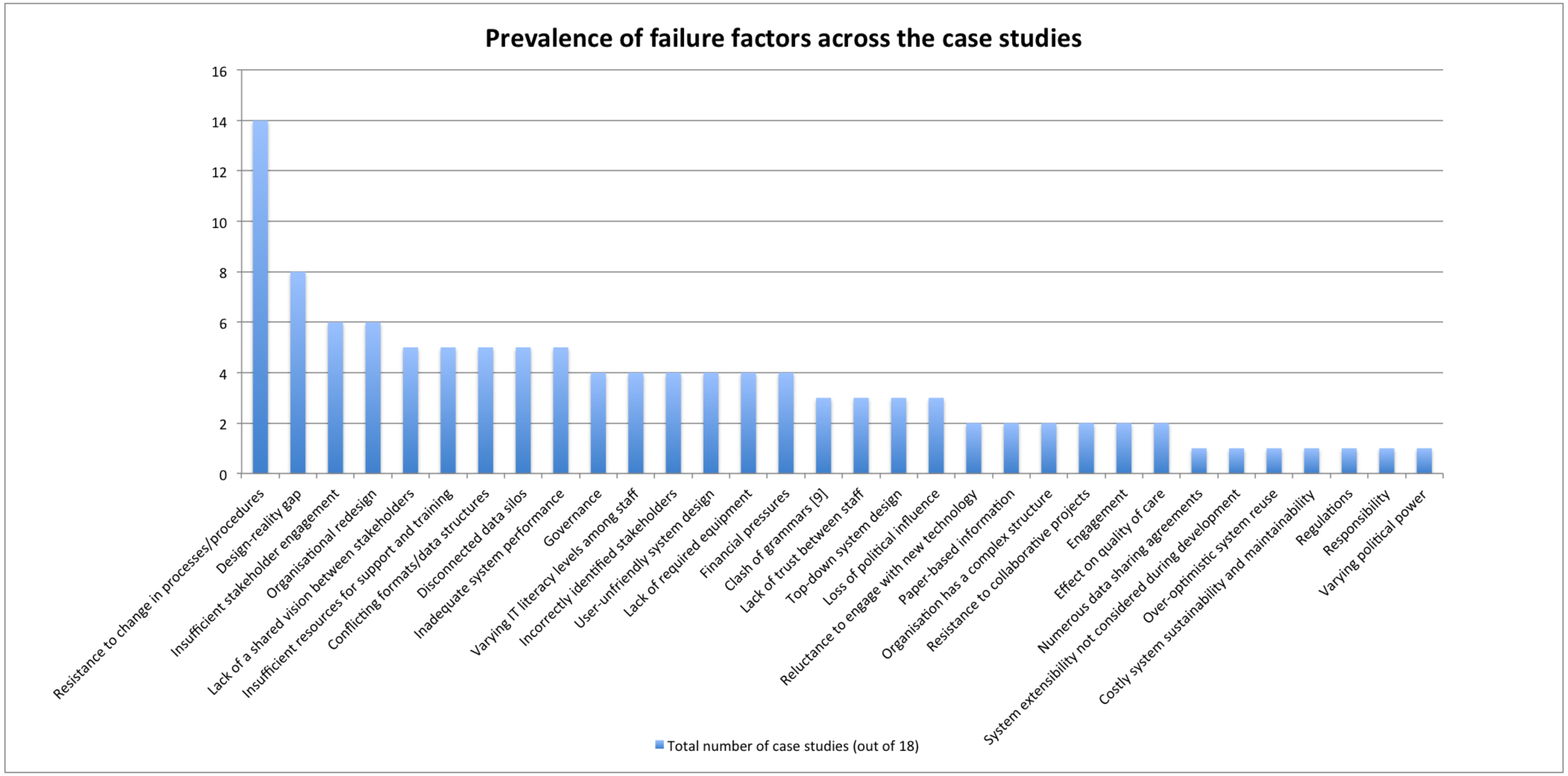


Figure 5.1: Prevalence of failure factors across the case studies.

data silos, inadequate system performance, paper-based information, system extensibility not considered during development, over-optimistic system reuse, and costly system sustainability and maintainability.

Clearly, the technical challenges of data movement are implicated in many of these failure factors. Costs introduced by the need to transform data from one format to another have long been recognised, and tools to alleviate the problems have been developed. However, other social factors, like fear of change, lack of trust, miscommunication, power games, are more complex.

Looking more closely, we can see that data movement is implicated in many of the non-technical failure factors, too. Many of the factors come into play because data are moved to allow work to be done in a different way, by different people, with different goals, or to enable entirely new forms of work to be carried out using existing data. For example, changes that end-users resist come about when work is moved from one part of an organisation to another (and with it, the data needed to complete that work), or when tasks formerly done by others must now be done by the end-user (with the help of the data we have moved to allow this). The design-reality gap, a term coined by Richard Heeks [87], refers to incorrect and misleading assumptions the design team may hold about how the system will work, and which conflict with the reality of how the work is actually done in practice. The risks of such assumptions are amplified when data are shared between teams, potentially bringing a design-reality gap for each team involved with the design team, and even between end-user teams. All these factors can be present in the context of a single team, and a single application, but they are amplified when data are moved between teams, between systems, and between organisations.

A dominant cost found in the case studies is the administrative cost of complying with data sharing agreements when data are shared with an organisational structure, other than the one that created the data. An example retrieved from one of the case studies specifically mentions that the sharing of data between the foundation trust and the GP is controlled by a myriad of data sharing agreements, each with specific services for a tightly agreed subset of the clinical record. The case study author explains that the high number of sharing agreements caused the administration of the IT system in place to be a significant overhead and ultimately a disadvantage for the whole IT project.

A type of risk found in the case studies is the misinterpretation of the data when shared with organisations or staff other than the creator of the data. There were numerous instances in the case studies where information captured by clinical staff (e.g. GPs, radiologists, etc.) had to be transcribed into the systems by clerical staff (e.g. medical secretaries). However, information that is obvious for the clinical staff might not always be known to staff with a different background and expertise. This often causes data not to be input correctly into the target system leading to decrease data quality and increased risk of misinterpretation, or sometimes not to be input at all. Particularly, a case study in a cancer care setting reported a reduced 20% coverage of the cancer stage (one of the main indications for cancer survival analysis) after a move of ownership to another department which utilised secretarial resources to capture this information. In this case, the stage of cancer was naturally inferred by the other attributes within the form, however, secretarial staff did not have the knowledge, nor the expertise to determine, and casually left the field empty. Other costs found in the case studies are reduced data quality, transformation costs, transportation cost, etc.

In conclusion, from the case studies analysis we found that data move not only through the technical infrastructure of databases and networks, but also through the human infrastructure, with its changing rules, vocabularies and assumptions. This suggests that data movement could be a proxy for some of the non-technical risks and costs the case study authors experienced, as well as the technical costs and challenges. The question therefore arises as to whether we can use the presence of data movement as the backbone for our model to identify cost and risk. If we can abstract the details of a new IT development into a map of the new data movements that would be required to realise it, can we quickly and cheaply assess the safety of those new movements, combining both technical and social features to arrive at our assessment of the risk? To answer the preceding question, we need to understand the specific features of data movements that can indicate the presence of cost and risk. We return to our case studies, to look for examples of data movement that were present when the IT development failed, and to generalise these into a set of data movement patterns that could become the basis for our method. The results of this second stage of the analysis are described in the following section.

5.3 Data Movement Anti-Patterns

Having examined the case studies, we found that data movement is an indicator of most of the IT failing factors. In this section, we propose a catalogue of data movement anti-patterns, each describing movements of data that might introduce some type of cost or risk to the development. We also give the conditions under which a pattern causes a failure, and the type of cost or risk it might impose on the organisation.

An anti-pattern is defined by Ambler to be “the description of a common approach to solving a common problem, an approach that in time proves to be wrong or highly ineffective” ([6], p.20). Similar definitions are given by Budgen and Koenig [38, 107]. Though the term is more normally associated with software design, it can also be applied more broadly. In our context, a data movement anti-pattern is a commonly occurring movement of data that appears to solve the problem of conveying needed data to a consumer, but which produces higher-than-expected costs in the longer term, that can reduce the overall value obtained from the project as a whole.

To develop the data movement anti-patterns catalogue, we followed the strategy introduced in section 4.3 on page 58. In particular we followed the following steps:

1. **Data extraction:** We read through the case studies again, looking for any type of data movement. We extracted any found instances into a data extraction sheet noting the following:

    - The no of the case study.
    - The producer or number of producers of the data to be moved.
    - The receiver or number of receivers of the moved data.
    - The data to be moved.
    - The type of movement.
    - The format of data to be moved.
    - The barriers obstructing the efficient and effective movement of data.

    Above data were captured in a simplistic notation we adopted from UML modelling notation in the format of diagrams. Also, we extracted from the case studies as movements of data those that were described by the authors as needed but not yet implemented since if properly developed would solve an existing problem and lead to a more effective patient care.

    The case study authors had not been asked to provide this information explicitly in their assignment, and therefore we used our own judgement to infer data movements involved based mostly on information present in the case studies. In the cases that important information was left out, as it was considered to be implicit by the authors of the case studies, we turned to the web and literature to fill in.

2. **Searching for themes:** In order to identify themes across the data we revised each extracted movement and took apart any domain-specific details originating from the case studies, but without altering the topology structure or any of the barriers that made them a problematic movement. We then transformed those examples into a set of generic anti-patterns by collating similar examples together.

3. **Reviewing themes:** Finally, we reviewed the extracted data movements and the themes they relate to, to determine whether the created themes satisfactorily capture the problematic data movements identified in the case studies.

All of the case studies involved some kind of data movement and it was commonly the case that the data movement was at the heart of the part of the development that failed. Although there were many examples of movement of data between computer systems we found a richer variety of movement patterns between people, from people to systems, and *vice-versa*. We describe below the anti-patterns we identified from the case studies. For each pattern we give an identifying name, define the context in which it can happen, and provide the conditions that should hold for costs to arise. Any examples given are taken directly from the case studies (but with identifying details removed). The anti-patterns are illustrated in figure 5.2 and further described in the following sections. Other potential costs, risks, and problematic data movement patterns may exist, here we present those reported in the case studies. All the anti-patterns are summarised in table 5.2 at the end of the chapter for easier future reference.

5.3.1 Change of Media

Often, a change of medium is required when data are moved from the producer to the consumer of the data. This is straightforward in the case of electronic data, which can usually be converted into physical report form, for document generation and printing. But the situation is more complicated when data on paper must be entered into a destination software system. We illustrate this pattern in figure 5.2 (a), and we describe it as follows:

Context: When data move from a source ‘S’ to a target ‘T’ of a different media (e.g. physical to electronic), then a transformation cost exists, either before or after the transportation of the data, that can lead to decreased quality at the ‘T’ side.

Cost and Risk: Data entry of physical data type into an electronic target system is a time consuming process, typically done by administrative staff who may not

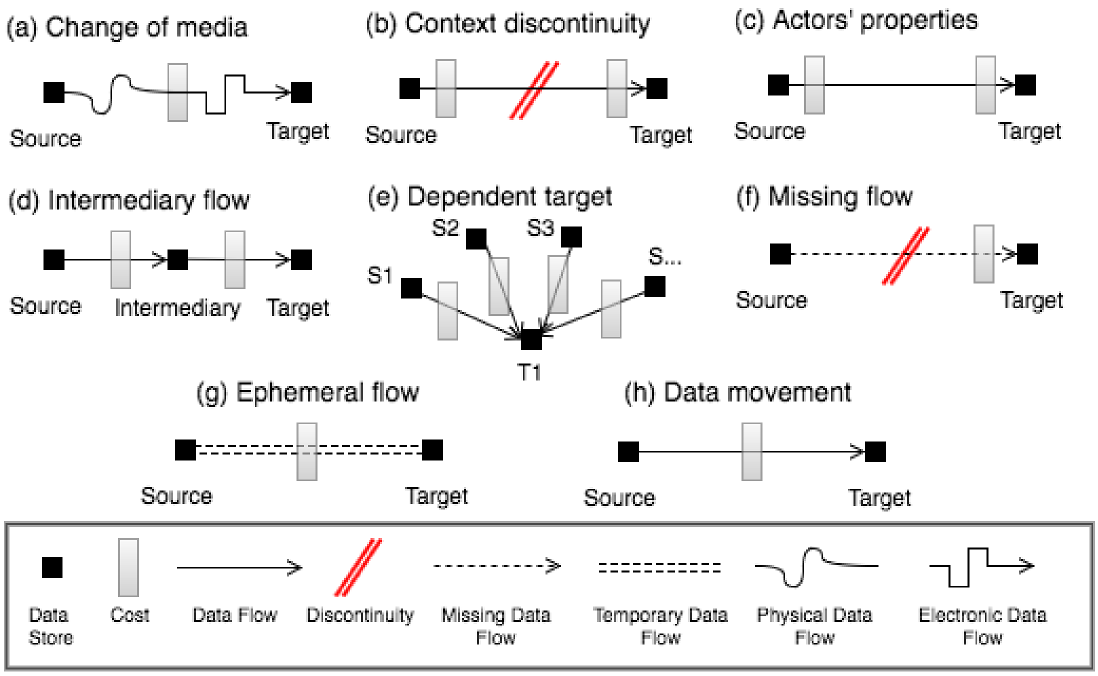


Figure 5.2: Data movement anti-patterns retrieved from the case studies.

have a strong understanding of the meaning of the data they are entering. Errors can easily be injected that may significantly reduce the quality of the information stored at the target side.

5.3.2 Context Discontinuity

Often data need to be used in a context other than the one they were originally designed for. Whenever there is a discontinuity in the movement of data caused by a change in context, costs will be imposed on the movement. The context discontinuity pattern is showed in figure 5.2 (b) and described as:

Context: When data move from a source ‘S’ to a target ‘T’ of a different context (e.g. organisation, geographical area, etc.) and a discontinuity exists in the flow, then a bridging cost is imposed to either or both sides of the flow.

Cost and Risk: Sharing data outside the immediate organisational unit can result in a number of administrative costs, such as reaching and complying with data sharing agreements, as well as complying with wider information governance and ethical requirements. Also, staff reluctance to share ownership of data may exist on both sides of the movement.

5.3.3 Actors' Properties

Costs can also be introduced by key heterogeneities in the properties of the consumer and producer of the data. Differences in system requirements, business processes, governance, and regulations between producers and consumers of data create transformation costs that must be borne either at the source or the target location (or both). Integrating data from "data island" sources (sources that have not previously been shared up to this point) can have high costs; such sources typically have limited external connectivity, and are tailored for use by one type of user, bringing a risk of data quality problems at the target side. Additionally, if the source of data belongs to a different context than the target, then there is the risk of "clash of grammars" (the meaning of the data moved being altered by the change of context because of cultural differences, experience differences, or other type of reason [175]), and a cost of lower data quality at the target side. For example, data entered into a system by secretarial staff can contain errors if they are not explicit and requires medical knowledge/vocabulary that the staff lack to fully understand. The actors' properties pattern is illustrated in figure 5.2 (c) and described as:

Context: When data move from a consumer 'C' to a producer 'P' (system or human), a difference in a property of either the source or target (e.g. clash of grammar, data islands) introduces a transformation cost to the movement.

Cost and Risk: There is the risk of clash of grammars, low connectivity, and the cost of lower data quality at the target side.

5.3.4 Intermediary Flow

Intermediary systems or staff may be introduced with the aim of reducing some upfront cost (such as the use of lower-paid staff to enter data on behalf of higher-paid staff), but can actually create downstream costs in the longer term such as those caused by lower data quality or missing data. The intermediary flow pattern is illustrated in figure 5.2 (d) and described as:

Context: When data move from a source 'S' to a target 'T' through an intermediary step, a cost is introduced to either data movement side.

Cost and Risk: There is the risk of missing data, and the cost of lower data quality.

5.3.5 Other Data Movement Anti-Patterns

- *Dependent target:* Often, data needed in a target location originally exists in several sources. If the business processes of the target depend on the data of the sources, then the cost of transformation is usually done on the target side. When data move from multiple sources to a target, and the target depends on the data in the sources then a cost of extraction, transformation and integration appears in each of the flows (figure 5.2 – e).

- *Missing flow:* Technical or governance barriers can often introduce a prohibitive cost that obstructs the implementation of the flow. Data needed by a consumer exists at a source, but are not able to reach the consumer (figure 5.2 – f).

- *Ephemeral flow:* Ephemeral is a flow from S to T that exists for a short period of time (i.e. migration purposes) and is planned to be deleted in the near future. Ephemeral flows are often created cheaply, with a short-term mindset, but then become part of the system, leading to future costs and complexity (figure 5.2 – g).

- *Data movement:* Apart from all the above cost and risk, whenever data move from its source to a destination there is the accumulative cost of extracting, transforming and loading the data from the source to the target. This cost often includes staff training and support, and can be in either side of the flow (figure 5.2 – h).

5.4 Discussion and Conclusion

In this chapter, we described our analysis of a set of 18 case studies from the NHS domain. From the case studies, we extracted the factors that the authors of the case studies identified as contributing towards the failure of the IT developments. We found a complex mix of failure factors. We found technical factors arising from sharing or integrating information, often stemming from the diverse data sources involved. But we also found other, often neglected factors stemming from the social aspects of the organisation; its people, policies, processes, governance, etc. This is consistent with results from other sources (e.g. [45, 72, 128]).

Other sources in the literature also discuss the variety of social factors that can affect an IT development [7, 8, 11, 20, 69, 87]. People are reluctant to change their current processes to use the new system in place, or user requirements are not met because of conflicting organisational policies and governance issues. For example, Greenhalgh *et al.* show the importance of human factors affecting the integration of electronic patient record (EPR) systems [80]. They state that the lack of consideration of the human factors is detrimental when bridging the design-reality gap in EPR systems.

Further analysing the IT failure factors, we found that most of them involved some kind of data movement either between people, systems or organisations. For example, in people-oriented factors we found that miscommunication between important stakeholders was the dominant failure factor, i.e. information that should have been shared between them, was not moved clearly or often not moved at all. Also, in data-oriented factors the medium in which information was captured and sharing agreements obstructed the efficient movement of data to the related recipients. Similarly, in governance-oriented factors data movement is avoided because of the risks of complying with corporate governance and data ownership guidelines. Additionally, in the requirements-oriented factors, system performance and functionality requirements are often not communicated clearly leading to a design-reality gap. Finally, organisation-oriented factors state that information that is needed in external teams or agencies are often not available because of the complex structure of the NHS and the myriad of agreements to comply with.

Extracting examples of data movements that lead to IT failing factors, as described in the case studies, we devised a catalogue of generic data movement anti-patterns that under some certain conditions can impose cost and risk in the IT development.

From the patterns, we found that cost and risk are likely to arise when data are moved between two entities that differ in some key way. When data are moved from producer (or holder) to consumer, it typically needs to be transformed from one format to another. Data values that make sense in the producer environment need to be converted into values that will be interpreted equivalently in the consumer environment. However, this conversion process is often difficult to apply correctly and completely, as the knowledge that is required is often stored tacitly in the heads of the data producers and consumers, rather than being explicitly declared in an easily accessible form. We can never store an item of data in its entirety in a database; instead, a considerable part of the meaning of a piece of data are stored outside the database into which they are loaded, in the local context conventions and assumptions of the people who create it. When data are moved out of its system of origin, this local, tacit part of its meaning is left behind. If this part of its meaning is not needed by the new consumer, then no harm is done. Or, if the local context and conventions of the new consumer match those of the data collector then, again, no harm is done. In all other cases, the portability of the data [1] is reduced and its interpretation altered making the data less useful after the movement than before.

Where data are sensitive (as health care data often is) there are also governance issues to be considered. Data often cannot be shared unless they have been appropriately aggregated or otherwise anonymised. Data may need to be filtered before they are moved, or moved through a particular set of systems, purely to be cleared for export to the real data consumer. There are myriads of data governance guidelines and protocols that must be met before sharing a medical dataset with a third party.

Governance and cultural challenges are just one part of the ways in which low information portability converts itself into a cost for data consumers. Technological differences between the source and target of a data movement step are an obvious factor (for example, where the source and target are represented using heterogeneous data structures and semantic conventions). But there are other challenges to be considered, that may be more costly overall and harder to predict in advance. Organisational, ethical, legal and other governance-related barrier can all come into play.

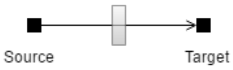
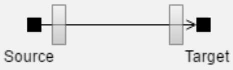
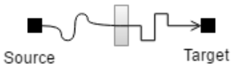
We are not claiming that the above set of patterns is complete; of course other types of data movement and a variety of other circumstances can bring cost and risk. It is, however, a starting point of a representative set of problematic data movements to help us understand the differences between source and target and the risk they might impose on the organisation, to help us devise early warning signs of cost and risk.

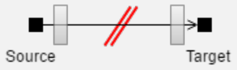
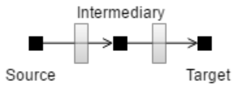
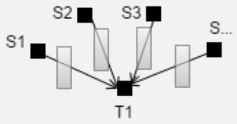
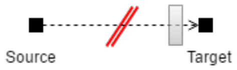
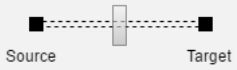
Apart from all the challenges, data movement is crucial to the functioning of most large organisations. While a data item may first be introduced into an organisation for a single purpose, new uses for that data will typically appear over time, requiring them to be moved between systems and actors, to fulfil these new requirements. Our societies now exist as networks of data, flowing from one system to another, and many crucial functions of government, industry, healthcare, education and science depend on the smooth functioning of these data flows. Can enterprises be viewed, at one level of abstraction, as networks of sub-systems that either produce, consume or merely store data, with flows between these sub-systems along which data travel?

When we plan to introduce new functionality into an enterprise, we must make

[1] Information portability is the ability of information to retain its meaning when moved to a context other than the one it was originally designed for (section 3.5).

sure that the data needed to support that functionality can reach the sub-system in which they will be consumed, so that value can be created from it. The costs of getting the data to their place of consumption must not be greater than the value generated by their consumption. Moreover, new risks to the enterprise will be introduced. The enterprise must evaluate the effects on its core functions if the flow of data is prevented for some reason, or if the costs of getting the data into place rise beyond the value that are produced. In the following chapter we investigate whether this abstraction could provide a lightweight early-warning indicator of the major cost and risk involved in introducing new functionality in an enterprise.

Data movement pattern name	Diagram	Context	Conditions, cost and risk
Data movement		Data move from a source (S) to a target (T) and there is a cost assigned to the flow of data.	If the movement is new, there is the cost of extracting, transforming and loading the data from the S to T. Also, staff might need training and support.
Actors properties		Data move from a S to a T and a property of the S or T introduces a cost to the movement.	For example, the S is a "data island" (a data source that has not previously been shared up to this point). Such sources typically have limited external connectivity, and are tailored for use by one user or group of similar users. Moving data out of a data island can be costly (because the mechanism for sharing has to be set up from scratch) and brings a risk of data quality problems at T systems. Additionally, if the S belongs to a different context than the T, then there is a risk of clash of grammars (the meaning of the data moved being altered by the change of context) and a cost of lower data quality at the T side.
Change of media		Data are moved from a S to a T of a different format of the data (i.e. from a physical type of data into an electronic one)	If the S stores the data in a different media from the T, then there is the cost of changing the media before or after they are transported. For example, if the S is paper-based and the T is digital, then there is the cost of data entry.

Discontinuity		Data move from a S to a T in a different context, introducing a discontinuity in the flow which imposes some cost to either or both sides of the flow.	Sharing data outside the immediate organisational unit can result in a number of administrative costs, reaching and complying with data sharing agreements, and complying with information governance requirements. There is also the risk of staff reluctance, on both sides of the movement. Finally, if the T system needs data in a different format to how it is stored at the S, then there is a transformation cost that must be borne, either by the S or the T system.
Intermediary		Data move from a S to a T through an intermediary step which introduces a cost to either flow.	Intermediaries may be introduced with the aim of reducing some up-front cost (such as the use of lower-paid staff to enter data on behalf of higher-paid staff) but can actually create downstream costs in the longer term (such as those caused by lower data quality or missing data).
Dependent target		Data needed in a T, exist in *several* sources.	If the data are moved from multiple sources, then a cost of extraction, transformation and integration might appear in each of the flows.
Missing data flow		Data are needed from a S to a T.	Often, there is a technical or governance barrier introducing a prohibitive cost that obstructs the implementation of the flow. Data needed by a consumer exists at a S, but are not able to reach the consumer.
Ephemeral flow		A data flow from a S to a T that exists for a short period of time (i.e. for migration purposes) and is planned to be deleted in the future.	Ephemeral flows are often created cheaply, with a short-term mindset, but then become part of the longer term system, leading to future costs and complexity.

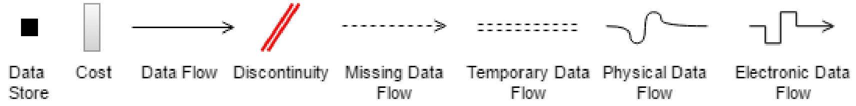


Table 5.2: Data movement anti-patterns likely cost and risk.

Chapter 6

Data Journey Model: Mapping Data Movement in Large, Complex Organisations

6.1 Introduction

Having identified a problematic set of data movements, the next step is to find a way to identify where the patterns occur in an information infrastructure to identify the places where high cost and risk are likely to exist. To do so, we need a way to capture the broad movements of data happening in the organisation. Specifically, we need a modelling technique that:

- allows us to model the movement of data within and between organisations,
- gives equal prominence to both social and technical factors affecting the movement of data, and
- is sufficiently lightweight to be used as a decision-making aid in the early stages of a development cycle.

Despite a plethora of modelling techniques and notations found in the literature for use during information systems design [2, 13, 16, 31, 49, 50, 156, 164, 186, 187], we only found a handful of methodologies that give equal prominence to both social and technical factors [140, 186, 187]. Of these, none were sufficiently lightweight to be used in early stage go / no go decision making.

In this chapter, we investigate whether a large organisation can be viewed, at a level of abstraction, as networks of sub-systems that produce, consume or store data, with flows between these sub-systems along which data travels. We examine whether this abstraction can give us a low-cost and lightweight way to identify problematic data movements in an information infrastructure that would potentially introduce cost and risk in the organisation. To test this hypothesis, we analyse the data movement anti-patterns looking for the minimum information that must be captured about an IT development and its organisational environment, in order to identify the presence of the patterns.

We propose a new technique of modelling the movement of data through and across organisations, aiming to identify the kinds of data movement that can lead to high risk and cost. The technique is lightweight as it abstracts away from the details of the business processes that use the data, and focuses just on the broad movements of data between significant organisational elements. The novel modelling technique, called data journey model, captures the movements of data within or between organisations through complex networks of people and systems.

We begin by describing the process we used to look for the minimum information to be extracted from the patterns (section 6.2). Next, we present the data journey model and notation (section 6.3). Finally, we propose a method for creating data journey models (section 6.4).

6.2 Data Movement in Large Organisations

Having identified the need for a new lightweight socio-technical technique (section 3.4), we set out to define a new modelling approach for data journeys, to capture the data movement anti-patterns we located in the case studies and described in chapter 5. In the sections that follow, we describe and justify the model we produced.

The question that now arises is whether we can capture the data movements happening in an organisation and identify the existence of the anti-patterns. Can we view an organisation, at an abstraction level, as a collection of “good” and “bad” movements of data? What is the least information we can capture about an organisation to be represented as a collection of data movements?

To address the above questions, it was necessary to understand the variety of data movements that exist in large organisations. We returned to the NHS case studies and looked for examples of data movements that the authors described. We found the following types of data movements:

- Data movement between people. For example, a GP sharing information with the Radiologist, clerical staff need information produced by clinical staff, etc.
- Data movement between systems. For example, patient's blood test results stored in the pathology lab database need to travel to the GP's database.
- Data movement between people and systems. For example, clerical staff have to input appointment details into the reception's database, or transcribe notes of clinical staff into the system.
- Data movement between organisational structures and teams. For example, data created and stored in the cancer hospital database needs to be shared with an external research agency as part of a new government guideline.
- Electronic and physical data movement. For example, information can be moved through a physical media (i.e. envelope, patient folder) or an electronic media (i.e. through an internet connection).
- Missing data movement. For example, a new flow of data is needed, but does not currently exist because of a prohibitive cost stemming from either a governance or technical barrier.

All examples involved the movement of a data entity between two places, the source and the target. The source is where the data entity originates from. It is the context in which the data entity was designed for, and is used. In some cases the source is a producer of new data entities. A producer is either a computer system or a human. Similarly, the target, is the end destination of a data movement step, to which the information travels to achieve a value-creating step. However, usually a collection of data movement steps are needed to accomplish bigger values. For example, for the GP to collect all information needed to decide on a patient action plan, several data movement steps are needed to transfer information from a variety of sources and producers.

The data entity is moved from the source to a target in a variety of means. In the case studies we identified two types, physical and electronic. In the physical one, data entities are captured in paper format and hence moved around the organisations in folders. In the electronic type, data are sent to the target location through a network connection. Although, naturally we would assume the electronic data movement to be the most dominant, we observed increase in the use of physical movement in the healthcare setting.

Having identified the above types of data movement, we can see the need for a conceptual model that can capture movement of data not just through the technical infrastructure of an organisation (the databases, systems and physical connections in place), but also the human infrastructure: the people interacting with the systems in place to create new information, transform and consume existing data.

6.3 Data Journey Modelling

In this section, we propose a new modelling technique, called data journey model, that captures data movements in large complex organisations. We begin by providing the requirements for our model, motivate its use and present its components and meta-model.

6.3.1 Requirements for Data Journey Modelling

The core requirement of the model is to identify the points in an information infrastructure where data are moved between two organisational entities which differ in some way significant to the interpretation of the data. These are the places where the portability of the data [1] are put under stress, where errors can occur when the differences are not recognised, and where effort must be put in to resolve the differences. The model must therefore have the following properties:

- Must capture the movement of data across an information infrastructure, including the entities which “hold” data within the system, and the routes by which data moves between them; we call this landscape.
- Must capture the points at which key differences in the interpretation of data occur, both social and technological.
- Must be low-cost to produce, since it is intended for use during early stage decision making.

[1] Information portability is the ability of information to retain its meaning when moved to a context other than the one it was originally designed for (section 3.5).

6.3.2 The Data Journey Model

In this section we describe a new model that captures an abstraction of an organisation (or collaborating group of organisations) designed to make explicit the kinds of problematic data movement patterns we identified from the NHS case studies, while hiding irrelevant details. We call this model a "data journey model", since it shows the high-level journeys data takes through the organisation in order to deliver value.

We use the term *data journey* to describe the movement of one or more data entities through the landscape, from its point of entry to its point of use. Data journeys are purposeful, implying that the data are needed at its destination for some value-creating step. For example, suppose a GP requests a patient blood test from a nearby pathology lab to decide on a further care plan. To do so, data needs to travel from the GP organisation (in the form of a request card and blood sample) to the hospital porter's pigeon holes, to the lab staff, to the lab's database (where results are input by the lab analyst), and back to the GP's database to await discussion with the patient. All these steps together make up the data journey that must be supported for the blood tests results to reach the GP and effective treatment decisions to be made.

The aim of a data journey model is to model the broad movements of a set of data entities through complex networks of people, systems and organisations. Data journey models do not attempt to provide a complete representation of an organisation or its processes; nor are they expected to model the complete set of data movements. Rather, they provide a simplification of reality showing the journeys of the entities of interest within the information infrastructure.

6.3.2.1 Data Journey Model Components

The components of a data journey model (and the notation we propose for them) are given in figure 6.1. We describe each component below.

We call the information infrastructure through which the data of interest moves the ***data landscape***. It includes both people and technical components that contribute to the creation, storage, transportation and use of information. We distinguish components that store data, from those that interact with data (creating them and consuming them).

Data containers are places where data can "rest" (be stored) on its journey through the data landscape. A container can be in electronic form (e.g. a database, an Excel file, a word document) or in physical form (e.g. file cabinets, desks, pigeon holes). We denote electronic data containers with the database icon and physical ones with

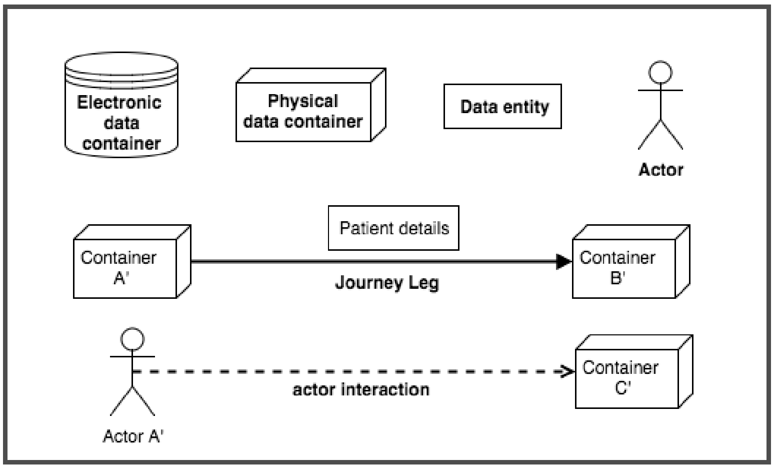


Figure 6.1: The notation of the data journey model.

a rectangular box, as shown in figure 6.1. In the pathology lab example previously introduced, the containers are the GP's desk, the GP reception desk (both storing the request card and blood sample), the pigeon holes of the hospital, the pathology lab secretary's desk, and the pathology system database (storing the blood test results).

Data stored in a container can travel to another container through some already established route. In the data journey model we call these container-to-container movements***journey legs***. A journey leg connects two containers if there is a medium through which they can share data. It allows data to move from a source to a target container to be used for a value-creating step or to await further onward movement. We denote journey legs with a straight line arrow connecting two containers. The direction of the journey leg shows the movement of a set of data entities from the source to the target container, as shown in figure 6.1.

Each journey leg, moves a ***data entity*** from a source data container to a target data container. We represent data entities on the model using a rectangle. Within the rectangle we write the data entity being moved.

The other type of component modelled in the data landscape are ***actors***. These are the people or IT systems that interact with containers to create, consume or transform the data stored in them. Actors are denoted using the actor symbol of the UML notation [156], and their interactions with the containers are shown as a dotted arrow beginning from the actor and ending to the container with which interacts.

Taken together, the legs between the components show the data journeys that are currently supported by the data landscape that is modelled. A ***data journey*** is a sequence of legs that makes a connected path through the data landscape for a set of

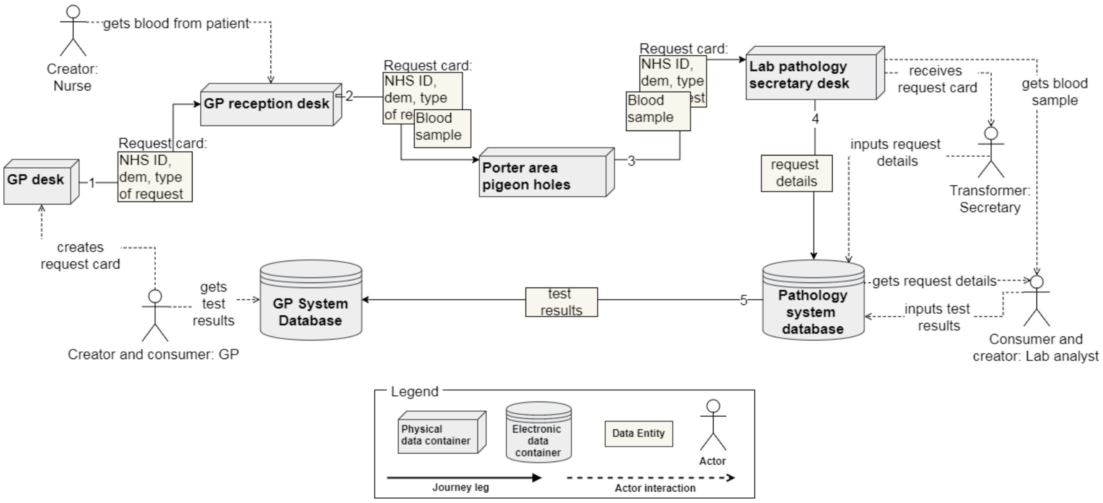


Figure 6.2: Data journey diagram of a GP requesting blood test results from a pathology lab.

data entities of interest, from a point of origin (a point at which the data first enters into the landscape) to a final destination (a point at which the data are used to deliver some value for some stakeholder). The way points on the journey may themselves be producers or consumers of this and other data items, or they may merely hold data. Journeys may be simple (traversing just one or two legs within a single department) or complex (covering a network of cooperating organisations across wide geographical or organisational distances). They may describe movements happening in an existing infrastructure, or they may describe planned movements for a proposed new IS development (in which case, certain legs may be missing in the existing data landscape, and need to be added to allow the proposed data journey to take place).

Figure 6.2 shows the data journey model of the pathology lab example previously introduced. The journey begins when a GP requests a patient's blood test. The results of the test are produced in the pathology lab.

6.3.2.2 Data Journey Meta Model

Figure 6.3 shows the meta model of the data journey model, expressed in UML class diagram. The meta-model indicates the relationships between the components of the data journey model, and the rules and constraints of the model. In the figure, each rectangular box denotes a component of the data journey model (actor, container, data entity, etc.). The top compartment of the box contains the name of the data journey model component, and the bottom compartment contains the attributes.

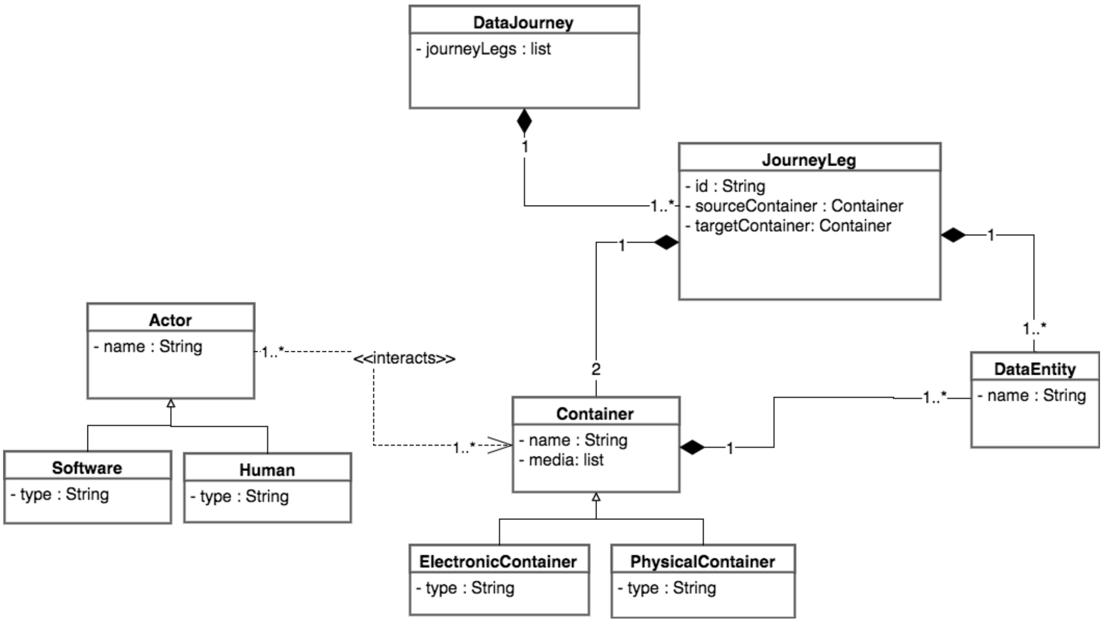


Figure 6.3: The data journey meta model.

In the figure, a line with a filled diamond shape indicates a composition, or ‘has a’ relationship. For example, in the data journey meta model a data journey is composed by a sequence of one or more consecutive journey legs. A journey leg is composed by two containers (a source and a target container), and a set of one or more data entities. Similarly, a container stores one or more data entities.

A solid line arrow in the figure indicates an inheritance, or an ‘is a’ relationship between two components. For example, a human is an actor. Also, a container can be either electronic or physical container.

Finally, a dotted line arrow in the figure indicates a dependency relationship. Such a relationship represents the dependency of one data journey model component to another. For example, an actor depends on a container to consume, transform or create new data entities.

6.4 Creating Data Journey Models

In this section, we propose an approach to create data journey models. In general, models can be constructed using top-down, or bottom-up techniques. Here, we present a bottom-up approach indicative of our experiences creating data journey models.

The first step of our approach, is to identify the scope of the movement we want to model, i.e. the set of data entities of interest that we want to model its movements.

For example, returning to the pathology lab example introduced above, the scope of the movement that needs to be modelled is the collection of the data entities needed for a GP to decide on an action plan. The data entities of interest in this example are the blood sample, the test result and the patient's details.

The next step is to construct the model's landscape. This is the information infrastructure in which the data entities of interest are stored and consumed, i.e. the data containers and the actors interacting with the containers to consume, transform or create data.

Finally, the last step is to add to the landscape the journey legs that move data entities from a source to a target container. Having completed this last step, the journey of the data of interest is formulated showing the origin of the data, the journey through the information infrastructure of the organisation and the destination.

Figure 6.4 shows a working example of the steps we took to create the data journey model of the pathology lab example. We further explain each step below:

1. We identified the data entities of interest.

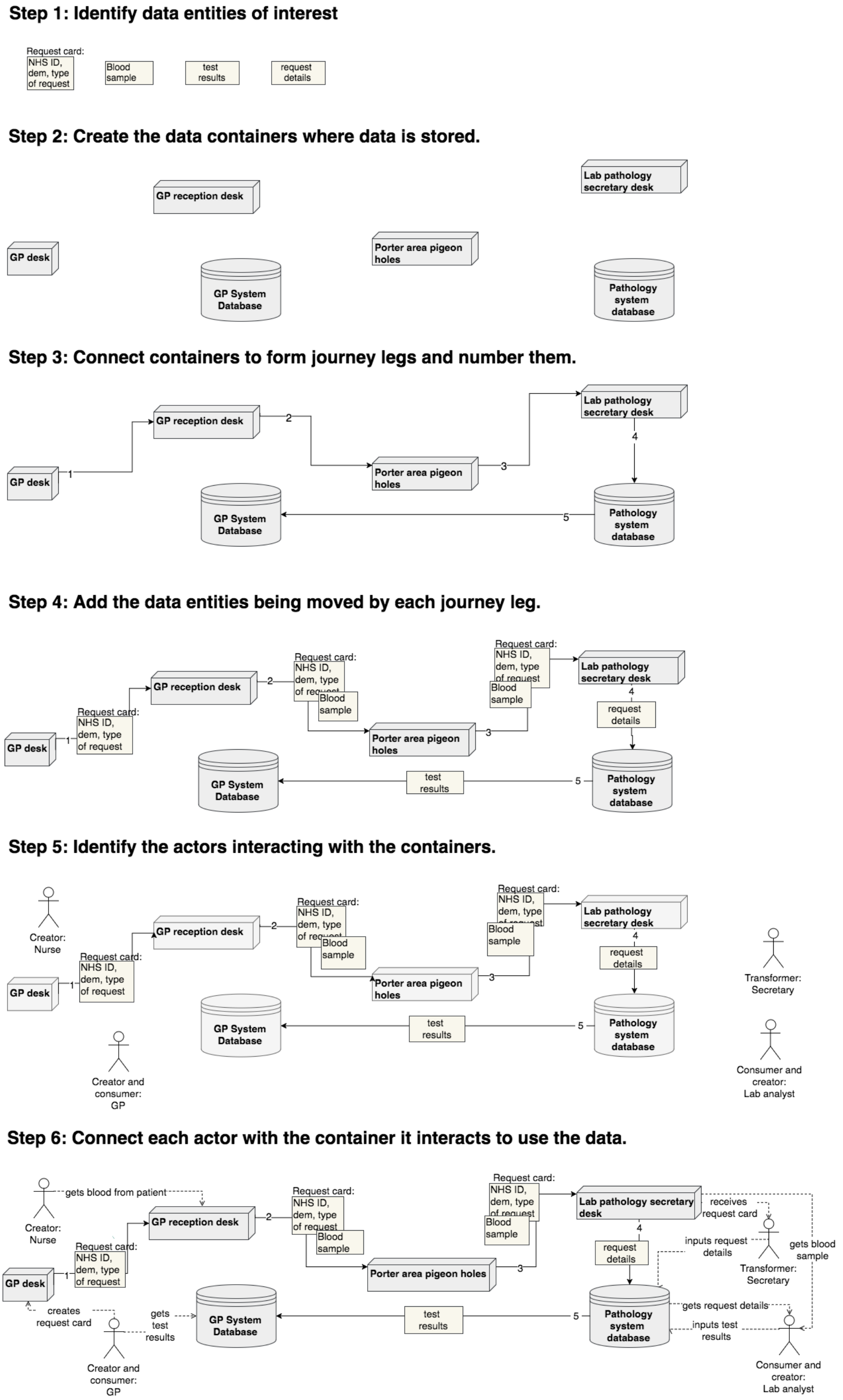
2. We identified the containers in which the data of interest are first entered into the landscape, and those where they are consumed to generate value, and those where they reside *en route* to their eventual consumption. We then constructed the identified containers to the model using a rectangular box for the physical ones and a cylinder for the electronic, as illustrated in figure 6.4.

3. We identified the routes by which data entities move between containers. Then, we added the routes to the model using straight line arrows to form journey legs. The direction of the arrows denote the movement of data from a source to a target container. We numbered the journey legs for future reference.

4. For each journey leg, we added to the model the data entities they move from the source to the target container.

5. We identified the actors interacting with each container by looking for the following inclusion criteria:

   - is either a person or system,
   - interacts with an identified container to create, consume or transform the data entities of interest (identified in step 1),



Step 1: Identify data entities of interest

Step 2: Create the data containers where data is stored.

Step 3: Connect containers to form journey legs and number them.

Step 4: Add the data entities being moved by each journey leg.

Step 5: Identify the actors interacting with the containers.

Step 6: Connect each actor with the container it interacts to use the data.

Figure 6.4: Working example constructing the data journey model of the pathology lab example.

   - the interaction with the container creates some value that lies within the scope of the journey.

   Then, we added the actors to the model, and labelled each actor with its role in either creating, consuming, or transforming the data, and its position in the organisation.

6. Finally, we connected each actor with the container it interacts with, using a dashed line arrow beginning from the actor and ending at the container labelling their interaction.

6.5 Conclusion

In this chapter we proposed a novel modelling technique that captures data movements between organisations, through complex networks of people and systems. Data journey models are lightweight and low-cost to create as they capture the broad movements happening in an organisation. We then described the components and notation of a data journey model and finally, we outlined a method for creating new models.

Chapter 7

Applying Socio-Technical Boundaries to the Data Journey Model

7.1 Introduction

In the previous chapter, we explored whether an organisation can be viewed, at an abstract level as a collection of data movements, and proposed the data journey model. The data journey model captures the movements data make, within the technical and human infrastructure of an organisation, to reach its destination. Having modelled the data journeys, the next step in our project is to identify any problematic movements and identify high cost and risk they might impose on the organisation.

These are the places in the journey where some aspect of the use of data changes, in a way that might cause the portability of data to be put under stress. In particular, we are looking for the places in the journey where the data move away of their original context and their quality is endangered.

In this chapter, we propose the new concept of "data journey boundaries", a low-cost way to identify socio-technical places in the journey of data where the portability of the data is put in danger. We devise a lightweight method that uses data journey boundaries to identify key differences between the source and target of a journey leg, that may alter the original interpretation and use of the data and ultimately impose high cost and risk to the movement.

More specifically, our main contributions in this chapter are:

- An extension to the data journey model, called data journey boundaries, that based on socio-technical information can identify key differences between the source and the target of a journey leg.

- A lightweight method that layers socio-technical boundaries onto a data journey model and identifies places in the journey of potential high cost and risk.
- A proposal of a method for the low-cost up-front identification of new data journey boundaries in other domains.

Following sections motivate the new concept of a data journey boundary (section 7.2), describe how it can be used in relation with the data journey model (section 7.2.2), and present an extension of the data journey meta-model (section 7.3). Then, we propose a new low-cost method that identifies places of high cost and risk in new developments (section 7.4). Finally, we discuss other modes of use of the data journey model and our method (section 7.5), and give a proposal for identifying new boundaries when applying our method in other settings (section 7.6).

7.2 Identifying Boundaries in Data Movement

In this section we address the following questions and propose the new concept of boundaries to help us tackle them:

- What types of differences between the source and the target of a data movement step can cause cost and risk to the IT development and risk the failure of the planned IT systems?
- How can we identify these differences in the journey of data and identify the places where cost and risk are highly likely to exist?

From the analysis of the NHS case studies, we found that significant differences between organisational elements can impose high cost and risk on the movement of the data, and can ultimately cause the development to fail (i.e. the costs turning out to be higher than the planned benefits).

Technical differences between the source and the target of a journey leg can introduce a variety of challenges. For example, moving data between systems operating with different data types requires the transformation of the source data to match the target's data types. Other technical differences include format mismatches, database schema inconsistencies, system infrastructure differences and many others [83].

Other challenges stemming from organisational and human differences are often harder to identify. Examples are: information is moved between staff of different expertise, staff with clashing grammars, different governance and ethical guidelines, organisational structures, geographical hierarchies, etc. [8, 20, 80].

In order to identify places of cost and risk, we need a way to identify above differences in the journey of data noting the places where data travel to a context other than the one it was originally designed for. These are the places where the "portability" of the data [1] will be put under stress, where errors can occur when the differences are not recognised, and where effort must be put in to resolve the differences.

Examples of the most common socio-technical differences between a source and a target of a data movement step found in the case studies are:

Context discontinuity pattern: a difference in the organisation in which the source and the target belongs to, may cause the administration costs of complying with numerous governance and ethical guidelines.

Change of media pattern: a difference in the media between the source and the target of a data movement step, can introduce high transformation costs to the IT development.

Actor's properties pattern: a difference in the properties and culture of the actors using the data at the source and the target, can reduce the quality of the data at the target side.

To look for the existence of the above differences between the elements of the data journey model, we first examine existing techniques that can help us and then propose a new approach to apply them on the data journey model.

7.2.1 Boundaries in Information Sharing

In this section we investigate uses of the term boundary in the information sharing community to see how is used to reveal challenges of information sharing. In the context of information sharing, a boundary is described as a discontinuity, representing incoherence and gaps between two entities. In the context of cross-boundary information sharing and integration in the public sector, a boundary mainly represents organisational boundaries between different public agencies. Pardo *et al.* uses the term boundary to denote the sharing of information across different organisational structures [141]. They describe a boundary as a discontinuity, representing incoherence and

[1] Information portability is the ability of information to retain its meaning when moved to a context other than the one it was originally designed for (section 3.5).

gaps between two entities. They investigate sustainable cross-boundary information sharing across organisational teams.

Organisational boundaries in the information sharing and integration literature have been studied a lot [11, 23, 32, 51, 184]. In particular, Carlile *et al.* and Espinosa *et al.* support the idea that organisational boundaries exist among different organisations and departments because of the differences in expertise, experience and regulations of different domains [39, 69]. In the public domain, organisational boundaries exist due to legislation causing differences in creating departments and organisations and the visions and missions of each [39].

In the information sharing and integration context other types of boundaries have been identified and used. Yang *et al.* discuss boundaries of personal, sectoral (between different organisational sectors), process and geographic type [183, 184]. They categorise the boundaries into two dimensions: horizontal and vertical. In the vertical dimension, information travels across government agencies at different levels (i.e. from the central department of an organisation to the local team of the same organisation), whereas in the horizontal dimension information is shared among parallel government agencies at the same level (i.e. central team of one organisation to the central team of a different organisation).

Zheng *et al.* define a boundary as the combination of the above types of boundary. More specifically, they propose a framework that categorises boundaries in terms of organisational elements that share information. These are: departments, processes, people, and operations. However, they use the term "boundary" in relation to the term "barrier". Zheng *et al.* differentiate the two terms; they define the term boundary as a neutral area that can cause the existence of barriers [184, 189]. Barriers are attributes of a boundary and what prevents us from crossing it. For example, barriers of a people boundary can be lack of trust, personal conflicts and others. Barriers can be overcome and/or eliminated, but boundaries tend to exist for a long period of time unless significant changes happen. Zheng *et al.* use a metaphor to better distinguish between the two terms.

> "A boundary is like a crossroad, and the difficulty of passing it depends on the potential barriers around it; like the weather, cars, pedestrians, etc. The crossroad can exist for a long time and may be easy to cross when conditions are favourable. However, barriers are the conditions that obstruct the driver to cross and can change in time. Drivers must be aware of both the boundaries and the barriers in every crossroad to cross safely."

([189], p. 9)

In this thesis we adopt the definitions given by Zheng *et al.* [189]. We use the term boundaries as elements that exist in the nature of an architectural landscape. They are lines clustering similar elements together, like departments of the same organisation, people of the same position role, systems with same interface, data with same data format, etc. Boundaries are neutral, they have no effect in the landscape; they are hidden. However, their existence is revealed when we have to cross them by moving a data entity to a location outside of the boundary. For example, if we want to move a set of data entities to another organisation we have to cross the boundary clustering all the elements of the source organisation. Crossing the organisational boundary we may have to face the cost of governance issues. The organisational boundary existed in nature and was revealed when we moved the data entities outside boundary's cluster.

7.2.2 Data Journey Boundaries

In order to identify significant differences between the components of a data journey model that can impose high cost/risk on the journey, we propose the new concept of ***data journey boundaries***. In the rest of this thesis we refer to this concept as 'boundaries'.

The aim of a boundary is to identify key differences between the source and target of a journey leg that can alter the interpretation and usage of the data at the target side. We combine our findings and the existing literature and use the term 'boundary' as followed:

Definition 7.2.1. Data Journey Boundary: Data journey boundary is the dividing line that clusters together data journey model elements with similar properties based on a certain condition.

A boundary indicates a context in which data can move without risking a lower data quality at the target side of a journey leg and introduce high cost and risk to the organisation. For example, an organisational boundary can cluster all containers that belong to the same organisation, department or team. Any journey legs moving data entities within the organisational boundary will not have the cost of complying with respected data sharing agreements and information governance protocols. However, a journey leg that moves data across the boundary will have the extra cost of complying with the protocols.

Another example, is the clash of grammar issue identified above. A boundary clustering actors of the same role in an organisation indicates the safe movement of data between actors of the same role. A journey leg that moves data across the boundary to an actor of a different role and possibly expertise will have the risk of data misinterpretation and possible lower data quality at the target side.

We use the term ‘barrier’ as the cost that is assigned to the organisation when a data entity crosses a boundary. The cost can be trivial (transportation costs from the source to the target), or can be immense. Identifying and quantifying the cost for each crossed boundary is a very challenging and expensive process which does not fit well with the lightweightness and cost-effectiveness requirements of this project.

The presence of a boundary increases the risk of data crossing it, and a barrier to impose a cost to the organisation. Therefore in this project, we abstract away from the expensive details of predicting the barriers in the journey of data and mainly focus on finding the boundaries. Identifying the boundaries in the data journey will reveal to the modeller the places where risk is increased and costs are likely to arise. These are the places where further analysis is needed by the decision makers to find the barriers and predict the type of cost and risk.

Having introduced and defined the concept of a boundary, we need a way to identify the boundaries in the journey of the data. Since there is little value in predictions that cost a significant fraction of the actual development costs to create, a boundary must have the following properties:

- A boundary must rely on easy-to-acquire information.
- A boundary must be low-cost to apply on the data journey model, so that we can quickly obtain results.
- Finally, a boundary must identify the places where data travel to a context that significant differences can reduce the interpretation of the data and add costs to the IT development.

The next step is to create easy-to-acquire and apply boundaries that can be overlaid on the data journey model and reveal the existence of differences. To do so, we collect information regarding the data journey elements to each categorise in a boundary cluster. In some cases, the information is readily available. For example, it is normally well known to stakeholders when information is stored on paper, in a filing cabinet, or in electronic form.

However, other properties, like cultural and staff grammar differences are less obvious. For these properties we use a *proxy*; some piece of information which is cheap to acquire, and approximates the same relationship between the actors and containers as the original factor. For example, we use salary bands as a proxy indicator for the difference in actors' vocabularies, on the grounds that a large difference in salary bands between actors probably indicates a different degree of technical specialism.

We use the following rules and proxies for indicating the presence of the boundaries as follows:

Organisational boundary: clusters together all data journey model elements that belong to the same organisation. An organisational boundary is crossed when the source container of a journey leg belongs to a **different organisational unit** than the target container:

$$sourceContainer.organisation \neq targetContainer.organisation \tag{7.1}$$

Containers boundary: clusters together all containers of the same medium (e.g. all the physical containers together, and all the electronic). A containers boundary is crossed when the medium of a source container of a journey leg is different from the **medium** of the target container:

$$sourceContainer.media \neq targetContainer.media \tag{7.2}$$

Actors boundary: the actor creating the data at the source container of a journey leg has a **different salary band** than the actor consuming it at the target container:

$$sourceContainer.actor.role \neq targetContainer.actor.role \tag{7.3}$$

Other, more costly, types of boundaries are also likely to exist. Here, we use the ones that we have, so far, found to be readily available and easy to acquire the information for.

7.3 Data Journey Model Boundary Extension

In this section, we extend the data journey meta-model to include the new concept of boundaries. As we have already seen previously, we retrieved three examples of boundaries from the data movement anti-patterns. The three boundaries are the organisational, actors and containers boundaries. Here, we use those three boundaries, as an example to extend our data journey meta-model.

The new extended model must be able to capture:

- The socio-technical information needed to form boundaries to indicate differences between the source and the target of a journey leg.

- The conditions that boundaries are based on to cluster data journey model elements together.

Figure 7.1 (on page 104), shows the extended meta-model of the data journey model that captures boundaries, expressed in UML. We explain each new concept below.

A data journey model can have a collection of boundaries. A boundary can be of a variety of types (e.g. organisational, actors properties, container media). Each boundary can be overlaid on the data journey model to form *groupings* of data journey model elements (containers, and/or actors) of similar properties.

Each grouping clusters data journey elements together based on a predefined condition. One boundary type can have multiple groupings overlaid on the data journey model, based on a predefined rule. The rules define the type and the number of groupings we need to cover the boundary.

Figure 7.2 on page 105 shows an example of the organisational and containers' media boundaries overlaid on the pathology lab data journey model. The blue line box labelled 'GP organisation' denotes a grouping of the organisational type of boundary and groups all the data containers in the model that belong to the GP organisation.

For example, considering the rule of the containers boundary given in equation 7.2 on page 102, and knowing that there are two types of media in the data journey model (physical and electronic):

$$container.media = \{physical, electronic\} \tag{7.4}$$

we will need a total of two groupings, one for each type of container media, to form the containers type of boundary.

Similarly, an organisational type of boundary (equation 7.1 on page 102) has one grouping for each organisation the containers belong to (i.e. the GP, the F.T., etc.):

$$container.organisation = \{GP,\ F.T.,\ clerical\ area,\ clinical\ area,\ etc.\} \tag{7.5}$$

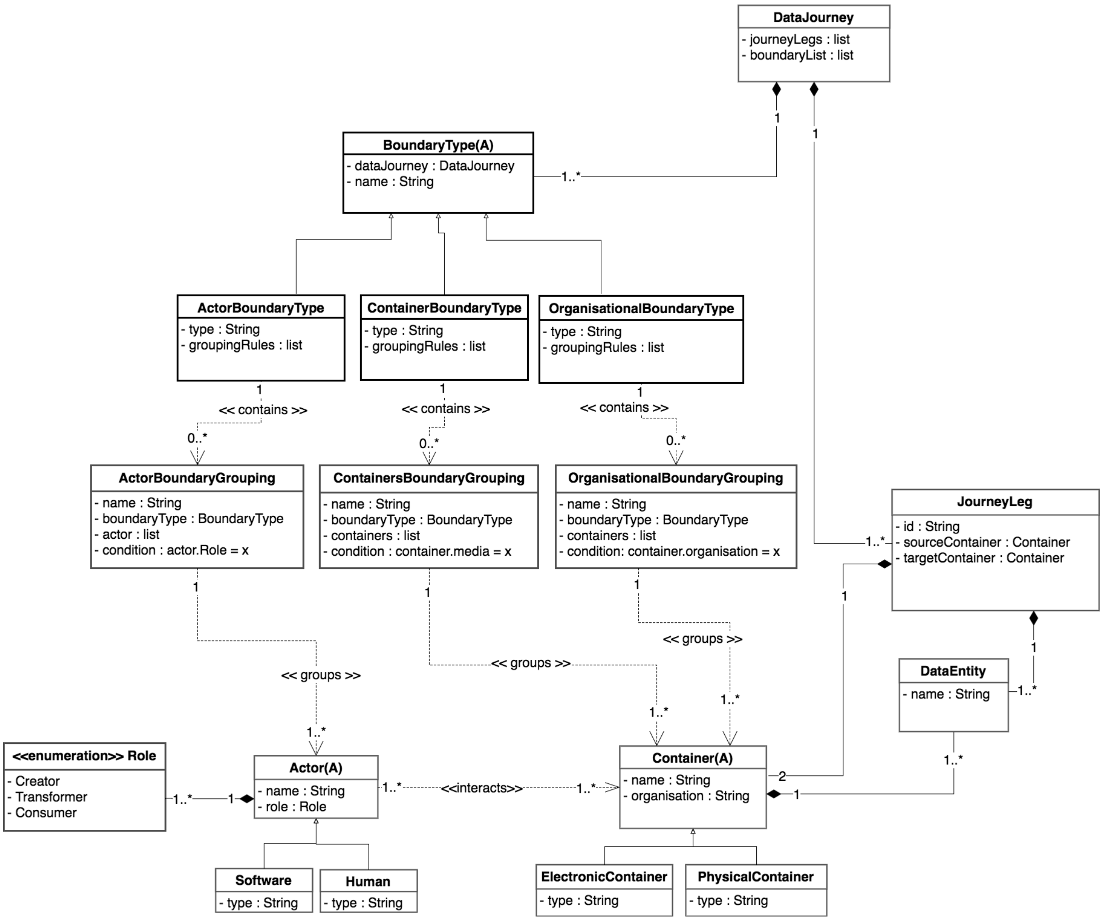


Figure 7.1: Data journey meta-model extension capturing boundaries.

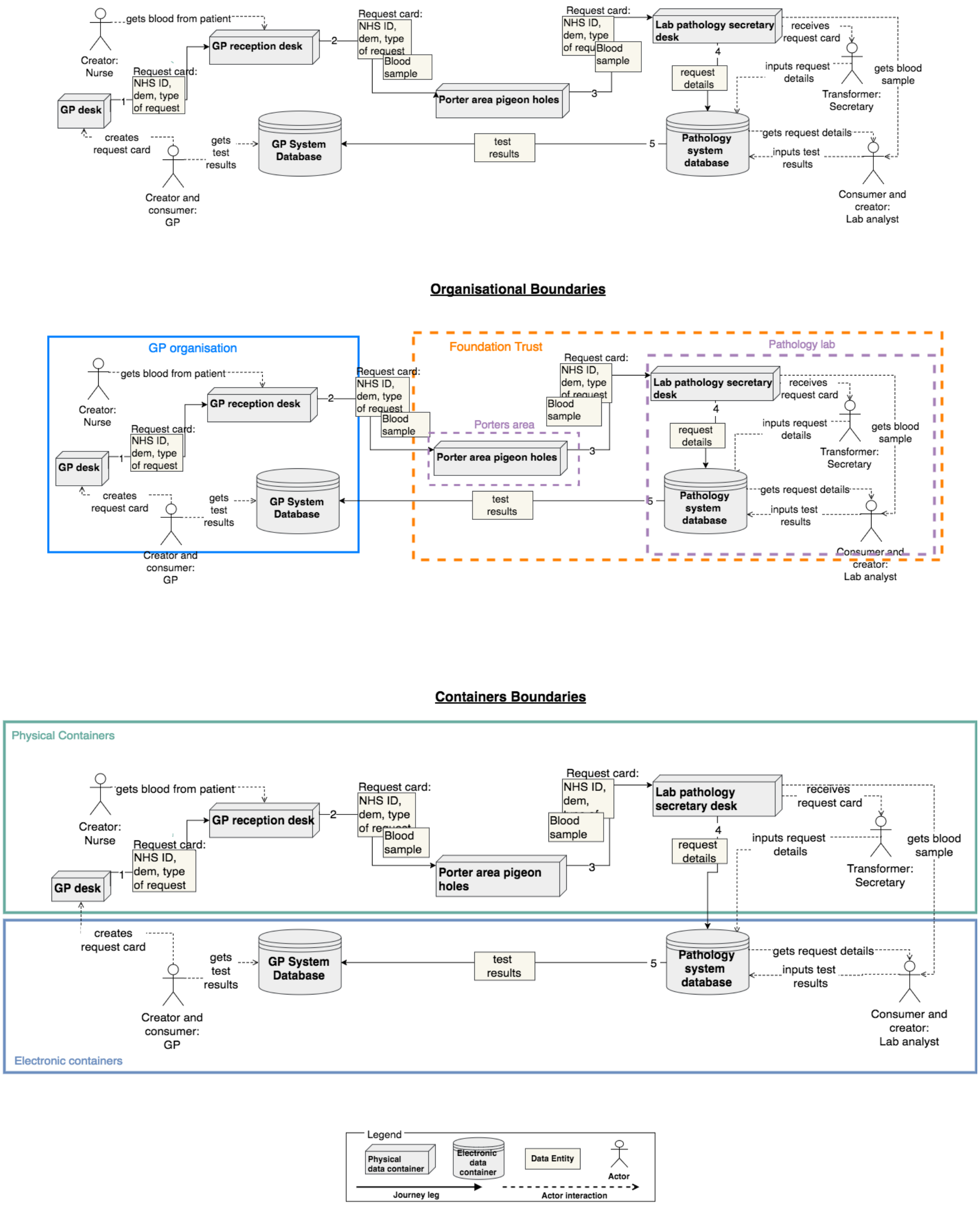


Figure 7.2: Organisational and containers boundaries on the pathology lab data journey model.

Although the boundary rules are predefined, the collections of the different types of media or organisations might not be known to the modeller. This is the type of information that the modeller will have to acquire from the domain experts. However, as already mentioned, the boundaries are designed to require only easy-to-acquire information not to burden domain experts' valuable time.

7.4 Identifying High Cost and Risk in Data Journey Models

In this section we explore how we can use the new concept of data journey boundaries to identify the places in the data journeys that pose the highest risk of imposing costs on stakeholders. We propose a new method that overlays boundaries on the data journey model to reveal the journey legs that move data across a boundary, and hence increase the risk of a barrier obstructing the smooth movement of data to its destination.

We need a method that guides us in:

- Modelling the necessary parts of the existing information infrastructure of an organisation, but also the parts of the planned new IT development to be integrated in the infrastructure.
- Modelling the movement of data from a point of entry in the existing infrastructure to the point of use by the new consumer.
- Identifying the places in the journey that because of some socio-technical boundary can impose high cost and risk on the new development.
- A mechanism to report back to managers the places of the movement with the higher cost and risk.

Our method begins by identifying the set of existing data entities that must travel to the new IT development. We then model the information infrastructure in which data entities currently exist and the journeys already happening within it, using the data journey model. Next, we add to the model the new development and the planned new journey. Once we have modelled both the existing and new data journeys we overlay on to the model socio-technical information to help us identify places in the journeys that can cause high costs. Figure 7.3 on page 108, outlines the five phases of our method. The following sections further describe each of the phases and provide figures illustrating steps to be followed using the pathology lab example introduced earlier.

7.4.1 Phase A: Model Existing Landscape

The first phase of our method involves modelling the existing information infrastructure of an organisation and the data movements that already happen within it. To do so, we follow the steps given in chapter 4 (section 6.4) on how to create a data journey model and illustrated in figure 6.4 (page 94).

7.4.2 Phase B: Model the Planned New Journeys

Next, we add to the model the planned new IT development and the journeys that would move data from the existing landscape to the new consumer.

For example, suppose that in the pathology lab model, a new external government agency requires demographics data for all GP requested tests, to assess effectiveness of workload sharing.

We add to the model the new development and journeys as follows:

1. *Add* to the model's landscape the containers of the new development that would store the required data.
2. *Connect* the new containers to form potential journey legs denoting the future movement of data from a source to a target container.
3. *Label* each journey leg with the data entities it will potentially move.
4. *Add* to the model the actors interacting with the new containers to create some value.

Figure 7.4 (on page 109) illustrates this phase of our method using the pathology lab example.

7.4.3 Phase C: Overlay Socio-Technical Boundaries

To overlay the boundaries on a data journey model, we group together the elements of the data journey diagram with similar properties based on the predefined rules and conditions. For example, we group together all the data journey elements belonging to

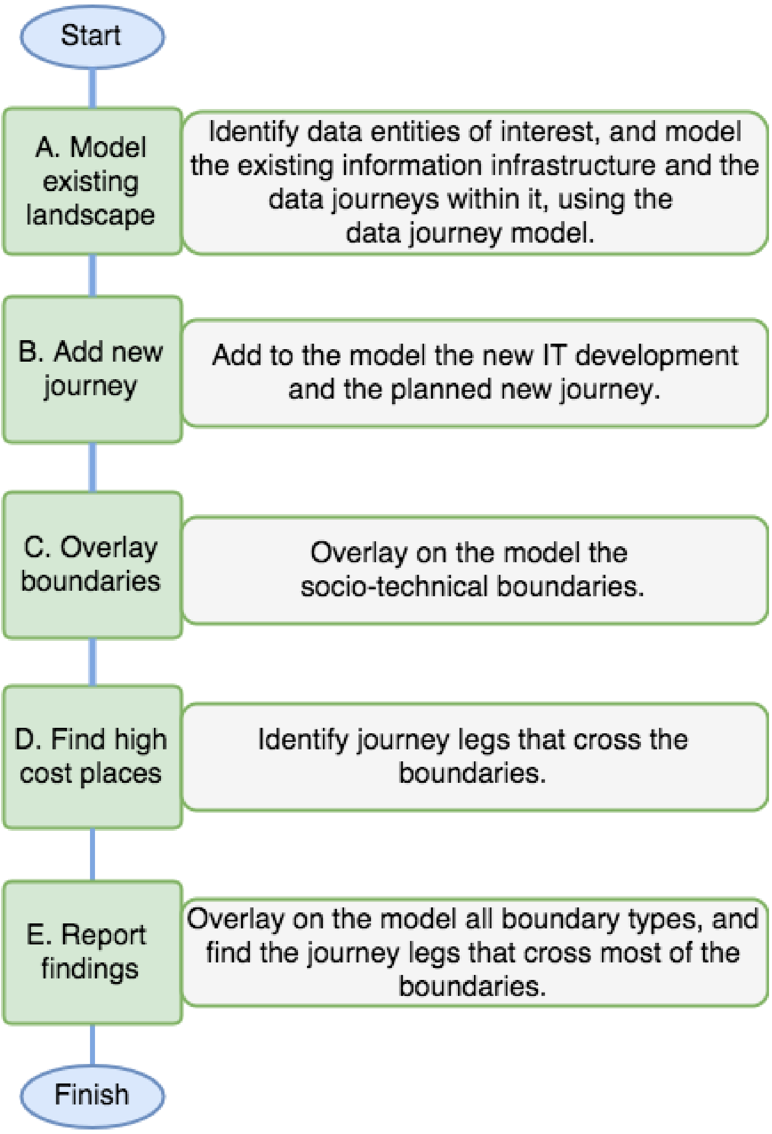
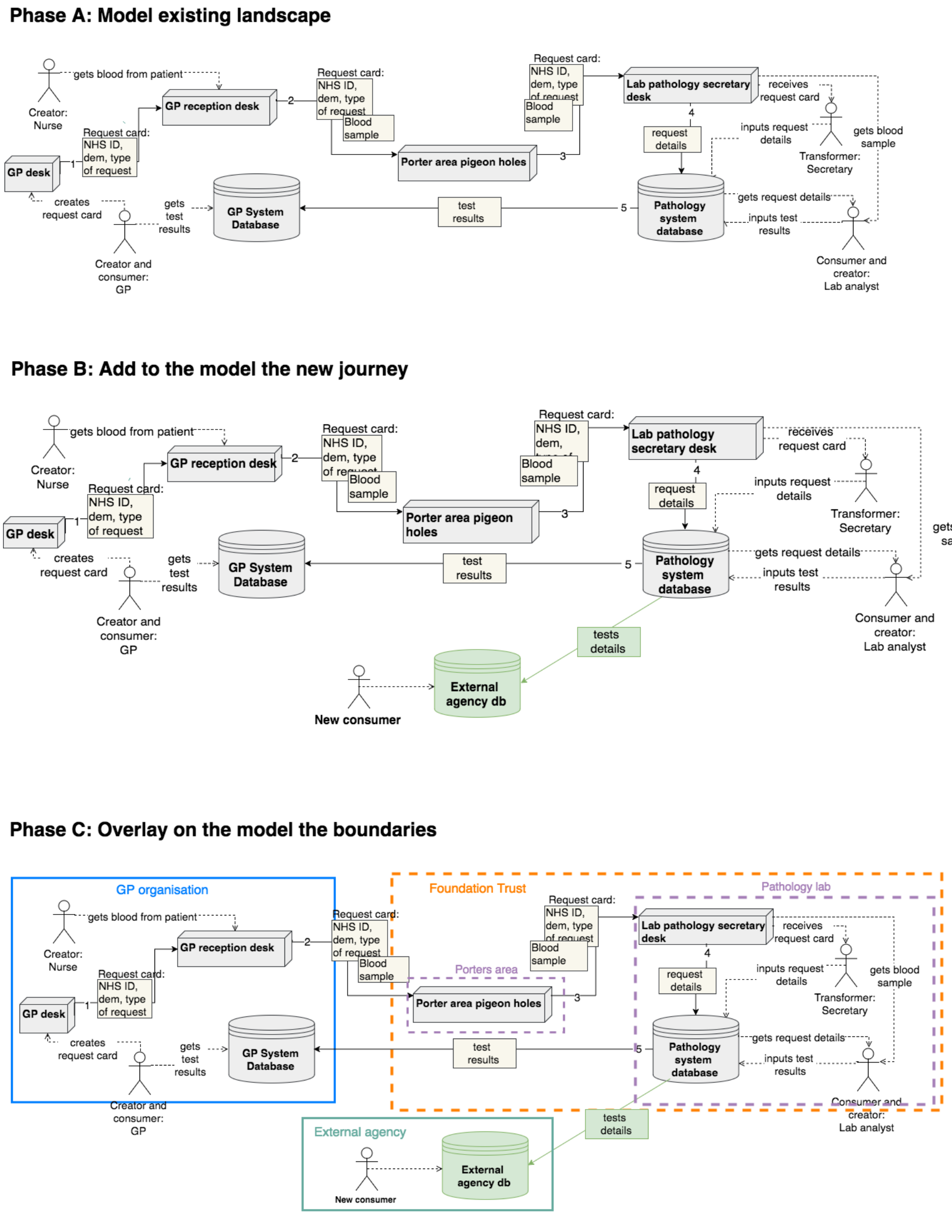

Start
A. Model existing landscape
Identify data entities of interest, and model the existing information infrastructure and the data journeys within it, using the data journey model.
B. Add new journey
Add to the model the new IT development and the planned new journey.
C. Overlay boundaries
Overlay on the model the socio-technical boundaries.
D. Find high cost places
Identify journey legs that cross the boundaries.
E. Report findings
Overlay on the model all boundary types, and find the journey legs that cross most of the boundaries.
Finish

Figure 7.3: Overview of our method to identify places of high cost.


Phase A: Model existing landscape

Phase B: Add to the model the new journey

Phase C: Overlay on the model the boundaries

Figure 7.4: Risk/cost identification in the pathology lab example (phases A to C).

the same organisation. We then overlay the groupings on top of the data journey model to form boundaries.

Phase C in figure 7.4, shows the organisational boundaries of the pathology lab example. The data journey elements belonging to the GP organisation are shown within the blue line box boundary, whereas the ones belonging to the hospital within the orange dotted line box. Within the hospital boundary we have the two in-depth boundaries; the containers belonging to the hospital porters area and the container that belong to the hospital's pathology lab.

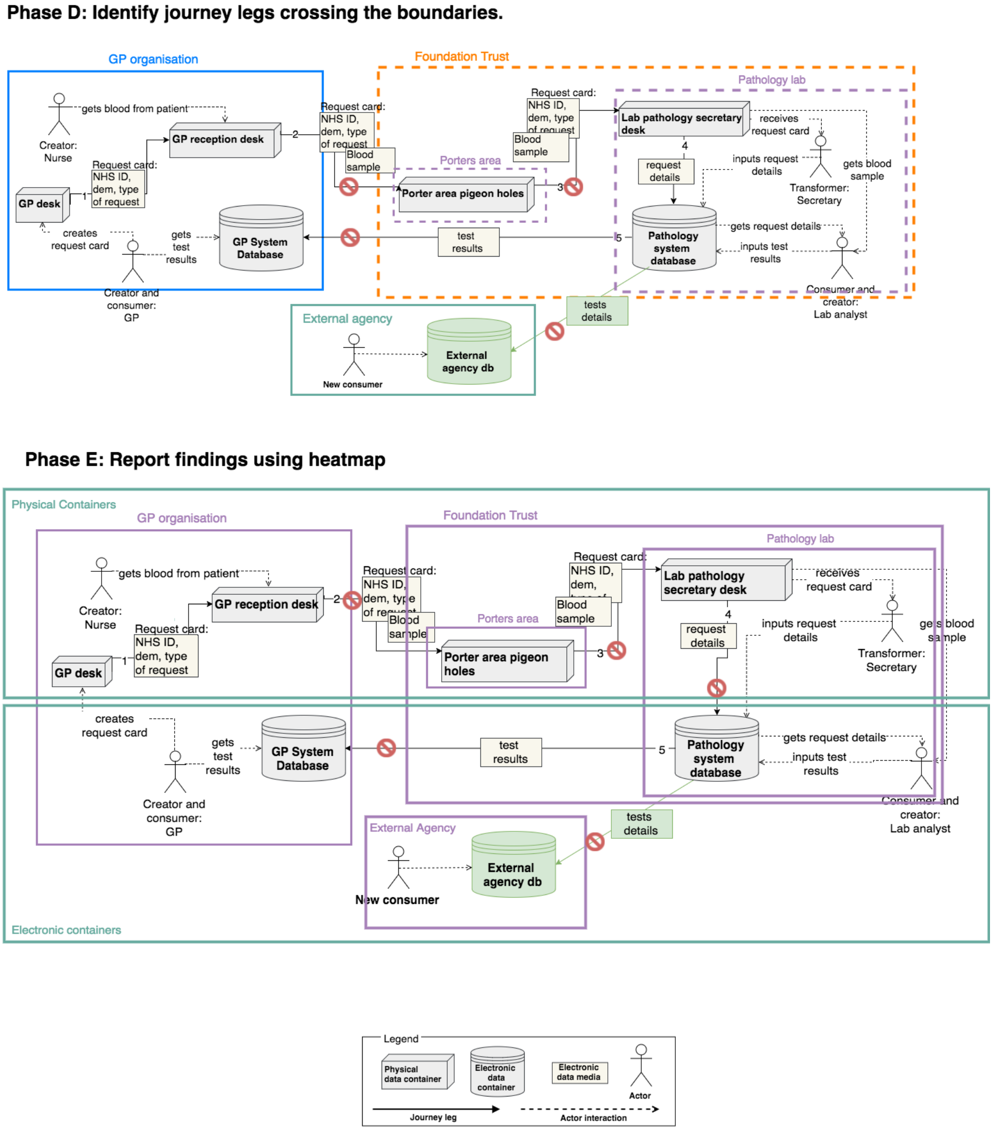
7.4.4 Phase D: Identifying Points of High Cost/Risk

Next, we identify the places in the journey of likely high cost and risk. To find those places, we identify the journey legs that move data across a boundary. If the source container of a journey leg belongs to a different grouping than the target container, then a boundary has been crossed that may impose high costs to the new IT development. Phase D step in figure 7.5 shows the costly journey legs (that crossed a boundary) with a red warning sign.

7.4.5 Phase E: Report Findings Using Heatmap

With several boundary types, there may be many points where a journey leg crosses a boundary. Are all to be considered equally costly/risky? We hypothesise that to identify the places with the highest possibility of cost and risk is to find the journey legs with the most crossed boundaries. Those are the places with the most differences between the source and the target container. We therefore hypothesise that the greater the number of differences in a journey leg, the highest the possibility of that leg to introduce cost or risk to the development.

To verify above hypothesis we devised a method to identify the journey legs with the highest cost and risk. We propose the use of the graphical method of heatmap. The ***data journey model heatmap*** graphically illustrates all the identified types of boundaries on the data journey model to highlight the places with a higher chance of risk. To do so, we overlay all groupings of all boundary types on the model. Then, the modeller can easily find the journey leg(s) with the most boundary crossings, denoted by the places with the most warning signs. Phase E part of figure 7.5 shows the full heatmap created for the pathology lab example, with the organisational and container media boundary types overlaid onto it.



Phase D: Identify journey legs crossing the boundaries.

Phase E: Report findings using heatmap

Figure 7.5: Risk/cost identification in the pathology lab example (phases D and E).

Having identified the journey leg(s) with the most boundary crossings, stakeholders and managers of the organisation have a list of risky data movement places and the type of cost to be expected. The list contains the places in the journey of data from an existing system to the new development that have been predicted as with the highest probability of cost and risk based on the number of boundary crossings (i.e. List: journey leg no, boundary(/ies) crossed, possible cost and risk).

7.5 Modes of Use

Our proposed method uses the data journey model and the new concept of boundaries to identify points of cost and risk when existing data move to a new IT development. It assists stakeholders in large, complex organisations make better informed decisions on whether adding new functionality to an already crowded infrastructure is worth pursuing or not. However, this is not the only way in which data journey model can be used. In this section, we outline two further modes of use of our method and the data journey model.

7.5.1 Nowcasting Optimisation Points

Another mode of use is nowcasting optimisation points of existing developments. Nowcasting, a recently introduced term in the area of economics, refers to the prediction of the present, the very near future and the very recent past [9, 10].

Often, organisations have imperfect views of the movements of data happening within and between them. People usually know the current state of the data they rely on, but not the whole picture of where the data came from, or the stops it made throughout their journey. As a consequence, decision makers have to “forecast the present” and build a clear idea of what the present looks like before they identify any points of optimisation in their current systems and infrastructure.

Our method can be used to help decision makers to nowcast the state of the current data movements happening within or between organisations and potentially identify opportunities for cost savings in existing IT developments. It can be used to highlight places in existing systems where optimisation is possible and the costs reduced. Our method can capture the data journeys already happening within or between organisations. By overlaying onto the model the socio-technical boundaries, we can identify the journey legs that move data across a boundary to another context, risking a lower data quality at the target side. By doing so, we can identify points in the current journeys of data where costs can be reduced and the journeys optimised.

7.5.2 Checking Compliance with Frameworks and Guidelines

Organisations often have no choice but to comply with regulations and guidelines (i.e. set by the government). Another mode of use of the data journey model is to check whether an organisation can comply with such compliance programmes. For example, the method can be used to assess organisational readiness to implement clinical guidelines set by government for the management of chronic conditions, like diabetes.

The guidelines can be modelled as sets of data journeys to check whether the organisation implements or not. If the organisation does not implement a data journey guideline, a cost of not complying with the related guidelines and/or regulations will be imposed on the organisation. Similarly, the method can allow teams to assess their readiness to comply with standardised care pathways or clinical guidelines.

7.6 Identifying Boundaries in New Domains

In this section, we propose a method to identify additional boundaries to use when modelling data journeys in a new domain. It is an up-front approach that takes place before any data journey modelling is initiated, and low-cost, so as not to jeopardise the lightweight nature of the data journey modelling technique. The process we propose for this is as follows:

1. "Grumble Analysis": hold a brainstorming session with stakeholders and domain experts to capture socio-technical challenges stemming from technical, organisational, regulations, guidelines, and social aspects, that they face in their everyday work.

2. "Good/Bad Analysis": for every challenge identified, check whether an already established boundary matches it. If not, a new boundary may be waiting to be discovered. Ask the experts to suggest characteristics that differentiate participant organisational elements causing the problems described by the challenge, from those which do not. Primarily, we look for binary properties of the participants ("good" or "bad" characteristics) that are simple and easy-to-acquire. For example, if data shared from external sources are often incomplete, we might

distinguish sources based on their data admin response times. “Good” suppliers respond within two days, “bad” suppliers respond less promptly.

3. For each such characteristic, we look for easy-to-acquire surrogates that can be used to form boundaries on the data journey model. An easy and quick way to do this is the tee-shirt agile approach [56]; categorise properties into simple broad classes, such as large, medium, small. Having categorised the properties of the participant organisational elements, we can group together the elements with similar size to form boundaries. Whenever data crosses a boundary it indicates the movement of data from a source with a good property to a target of a different property that would impose costs to the journey.

Having identified a pool of potential boundaries, we then apply them on the data journey models we create within the new domain to check their usefulness in identifying cost and risk in the new domain. So, over time, we can build up a core set of boundaries to be used within that particular domain.

7.7 Conclusion

In this chapter, we described an expansion of the data journey model that captures socio-technical information, called boundaries, that when overlaid on the model can identify potential places of high cost and risk. Boundaries group together data journey elements (containers and/or actors) that share similar properties, representing a context in which data can move without risking lower data portability.

Next, we developed a new low-cost method to identify places of high cost and risk when existing data move to a planned new IT development. The method overlays easy-to-acquire socio-technical boundaries on top of the data journey model to identify potential points of cost/risk. The method uses a heatmap approach to identify the places with the highest predicted cost/risk.

Finally, we explored other potential modes of use of both the data journey model and our proposed method. Our method may be used to identify opportunities for cost savings in existing systems, and assess organisational readiness for various compliance programmes. For the use of our method in other settings, we propose a method for the identification of a set of boundaries valid in the new domain.

Chapter 8

Applying Data Journey Modelling in the NHS

8.1 Introduction

In the previous chapters we devised a method that models the journeys data make through complex networks of people and systems, and identifies the places in the journey where high cost and risk are likely to exist. In this chapter we describe how we evaluated the method, present the results we obtained, and discuss the significance of these results. Specifically, we set out to answer the following questions based on the thesis research questions given in section 1.4:

- How accurate is our method in identifying places of cost and risk in the journey of data? Can the method produce accurate enough findings for a reasonable amount of time and effort invested by the domain experts?

- How 'stable' is our method across different organisations and domains? Is our method's three core boundaries (organisational, containers media, and actors role) that we have identified in our previous work likely to be capable of identifying points of cost and risk in other settings, or will each new setting require us to identify a new set of boundaries?

- How 'complete' is our method in identifying all significant places of cost/risk? Does the method systematically omit some types of cost/risk? Even if the three core boundaries are stable across domains, that leaves open the possibility that some important cost and risk are not being identified by our method, because they cross boundaries that we are not modelling.

- How expensive is it to determine possible important boundaries for new settings, before full data journey modelling has taken place? Even if we identify a set of boundaries that are stable and (largely) complete across domains, there is still the possibility that some particular organisation or context might have highly specific requirements that should be taken into account in the identification of cost and risk.

To answer above questions we applied our modelling technique in real-world case studies in the healthcare domain. We worked with clinicians from five NHS Foundation Trusts across the UK to model the movements of data of recently introduced IT systems and identified points of high cost and risk.

We followed a variety of methods in the case studies, each chosen to fit the circumstances of the opportunities we had to work with domain experts (some of which arose at short notice). In each case, we began by defining a procedure to follow, and took care only to document domain expert responses, and not to lead them to answers we might have wished they had given.

As with any large commercial or government institution, some aspects of the details of the case studies are confidential. Although the models used to evaluate our model and method are created based on the actual case study, in this thesis we present and use a more general model, typical of those used in a range of NHS hospitals. However, the results of the evaluation presented here are the original ones produced from the actual case studies.

We begin by presenting our evaluation of the method's accuracy in the radiology department of a nearby FT (section 8.2), and the method's stability in the domain of Clinical Genomics (section 8.3). Next, we evaluate the method's completeness in the full Clinical Genomics patient pathway (section 8.4) and finally discuss our results (section 8.5).

8.2 Evaluation of Method's Accuracy in an NHS Hospital

In this section we describe our efforts to answer the first research question outlined in this chapter on evaluating the accuracy of our method's findings.

8.2.1 Study Design

In this study we are looking for evidences to support or oppose the hypothesis that our method can quickly and cheaply provide accurate findings of points of cost and risk. To do so, we conducted a retrospective evaluation based on a case study in the healthcare domain. In the case study, we examined movement of data from GP organisations to the radiology department of an NHS Foundation Trust in the UK. Prior to the case study, hospital staff had identified costs and delays in their IT system handling the appointments and patient data in the radiology department, and had recently introduced a new improved system.

To evaluate our method we followed the retrospective approach outlined in section 4.5. We looked back at the old system before hospital staff made any improvement efforts to reduce delays and costs. We applied our method to the old system, without knowing what improvements were made, to identify high cost places in the data journeys. Then, we modelled the new system in place and compared our findings with the changes made by hospital staff. More specifically, we followed the steps below:

1. Conducted semi-structured interviews with two NHS domain experts working in the Informatics department of the Trust in the roles of Application Specialist and Integration Analyst. We structured the interviews not to reflect any preconceived theories or ideas. We had some predetermined interview questions aiming to gather domain knowledge on the data movements, organisational structures, available information systems, data formats and staff roles in the FT. However, we kept the structure of the interview open. We started with an opening question such as 'Can you tell me about your experience using the old system in place?' and then progressed based upon the initial response while kept an eye on gathering information based on our predetermined interview questions.

2. Modelled the data journey model of the *old* system (before improvements were made by hospital staff).

3. Applied our method on the 'old' model to identify journey legs of likely high cost and risk.

4. Modelled the data journey of the *new* system (after the hospital staff improvement efforts).

5. Identified the journey legs that hospital staff improved by comparing the two models ('old data journey' and 'new data journey').

6. Compared the predicted costly journey legs of the old model with the improvements made by staff to assess the accuracy of our predictions, with the help of the NHS domain experts.

When comparing our predictions with the improvements made by hospital staff, we assume that staff have correctly identified a set of cost improvements in the old system. However, we also assume that there are other places of potential cost savings that were not changed by staff for various reasons, such as limited time and resources. To fill this gap, we consulted the two NHS domain experts to assess the feasibility of our predictions that have not been improved in the new system.

8.2.2 Study Success Criteria

Before undertaking any evaluation exercise we define the criteria that will be used to determine whether the results support the hypothesis or provide evidence against it.

We do not expect our model to identify all the changes made by the domain experts in the new system, since that would require detailed and complete modelling, and more investment from domain experts. Instead, we are evaluating whether our lightweight model can cheaply and quickly identify high cost places where savings can be made without investing extreme resources (time, effort, money). We defined our goals for assessing the lightweight property of the method and the accuracy of the predictions as follows:

Domain expert's time We kept track of the time invested by hospital staff to provide us with domain information needed to make the models. We set the threshold to be one working day (up to 7 hours) of interview/modelling time for each 'old' and 'new' model, since hospital staff's time is valuable. This includes one initial meeting for setting up the scenes, an hour or two of offline checking gathering unknown domain expertise and a follow up meeting to confirm models to be produced.

Modelling time We also monitored the person-hours required to create the 'old' model and identify places of high cost and risk. Building the model and predicting costs must take a small fraction of the system development time, so based on the scale of this case study we set the threshold to one working day (up to 7 hours) for this, too.

Prediction accuracy We set a success threshold for the method of doing as well as the hospital staff in identifying points of cost and risk. That is, we would regard the method as successful if it could perform *at least as well* as the hospital staff in identifying points of cost and risk. The new processes and IT systems had been in place at the hospital for some time when the evaluation was carried out, and the hospital staff had seen evidence that the benefits they hoped for from them had been realised. We therefore took the pragmatic decision to treat the hospital staff had correctly identified a set of cost improvements in the old system. However, we also assumed that there might have been other potential cost saving measures that were not implemented by staff (perhaps because of limited resources, or because the opportunity was not identified at the time).

Since our model aims for a good-enough answer quickly, we do not expect it to be accurate and complete. However, every inaccurate prediction has the cost of further investigating it. Therefore, we set the conservative goal of considering the model to be accurate if it produces fewer wrong predictions than correct predictions.

8.2.3 Modelling the Old Hospital System

In the case study we examined the movement of data that occurs when a GP needs to decide on an action plan for a patient who may have a fractured bone. In the old system, before the hospital staff's improvements were implemented, the GP requests an X-ray to be taken at the local hospital's radiology department by filling in a request card. The request card is then sent to the hospital's radiology department by a courier. When the patient arrives at the radiology department for their apartment, a radiographer takes an X-ray image of the patient. A radiologist reviews the image and dictates a report regarding the X-ray findings. Then, the secretary transcribes the report into the system, prints it and sends it to the GP by post. Then the GP will decide on an action plan.

We conducted semi-structured interviews with the NHS domain experts to gather information on the movements of the data entities needed by a GP to decide on an action plan. These are: the patient's identification details and the report containing the X-ray findings. Appendix A.1 on page 169 gives a typical example of data movement processes in an NHS FT similar to the one of our case study. With the help of the expert's domain knowledge, we designed the data journey model given in figure 8.1 (page 121). The processes we followed to create the data journey model is given in appendix A.2 on page 170.

The data journey model produced has been validated by the PACS (Picture Archiving and Communications System) Imaging Informatics Manager of the hospital to ensure that the movements shown are representative of the hospital's processes.

8.2.4 Identifying Points of High Cost/Risk

Once we have modelled the journey of the data, we overlaid onto it the socio-technical boundaries to identify places in the journey of likely cost and risk. We overlaid the three core boundaries of our method as follows:

- Organisational boundary: groups containers and actors that are part of the same organisational unit: GP organisation, Community, NHS F.T., Archives, and Radiology department (as provided by the domain experts).
- Containers Media boundary: groups containers that use the same media for information transfer and storage: physical containers (storing the request card, patient folder, patient packet, x-ray, cassette), and electronic container (storing the report).
- Actors pay-scale boundary: groups actors who are at the same or similar salary level (clinical staff, clerical staff).

The data journey models with the overlaid boundaries are given in Appendix A.3 on page 173. Figure 8.2 overlays all three boundaries to form a boundary heatmap. The heatmap highlights the journey legs that cross a boundary. Journey legs with a red line arrow cross one type of boundary, whereas legs with a red double-line arrow crossed two types of boundaries. We found no leg that crossed all three boundaries in this case study example. Based on the heatmap (figure 8.2) the journey legs that crossed the most boundaries, and that therefore are predicted to have the highest possibility of costs, are: 2 and 4. Table 8.1 lists all the journey legs that crossed one or more boundaries, and describes likely cost and risk the crossed boundaries might impose on the movement based on the data movement anti-patterns.

8.2.5 Modelling the New Improved Hospital System

In the old system, X-rays were captured on large X-ray films and stored in the Archives room. This caused delays in transporting the films to the place of need, while investing

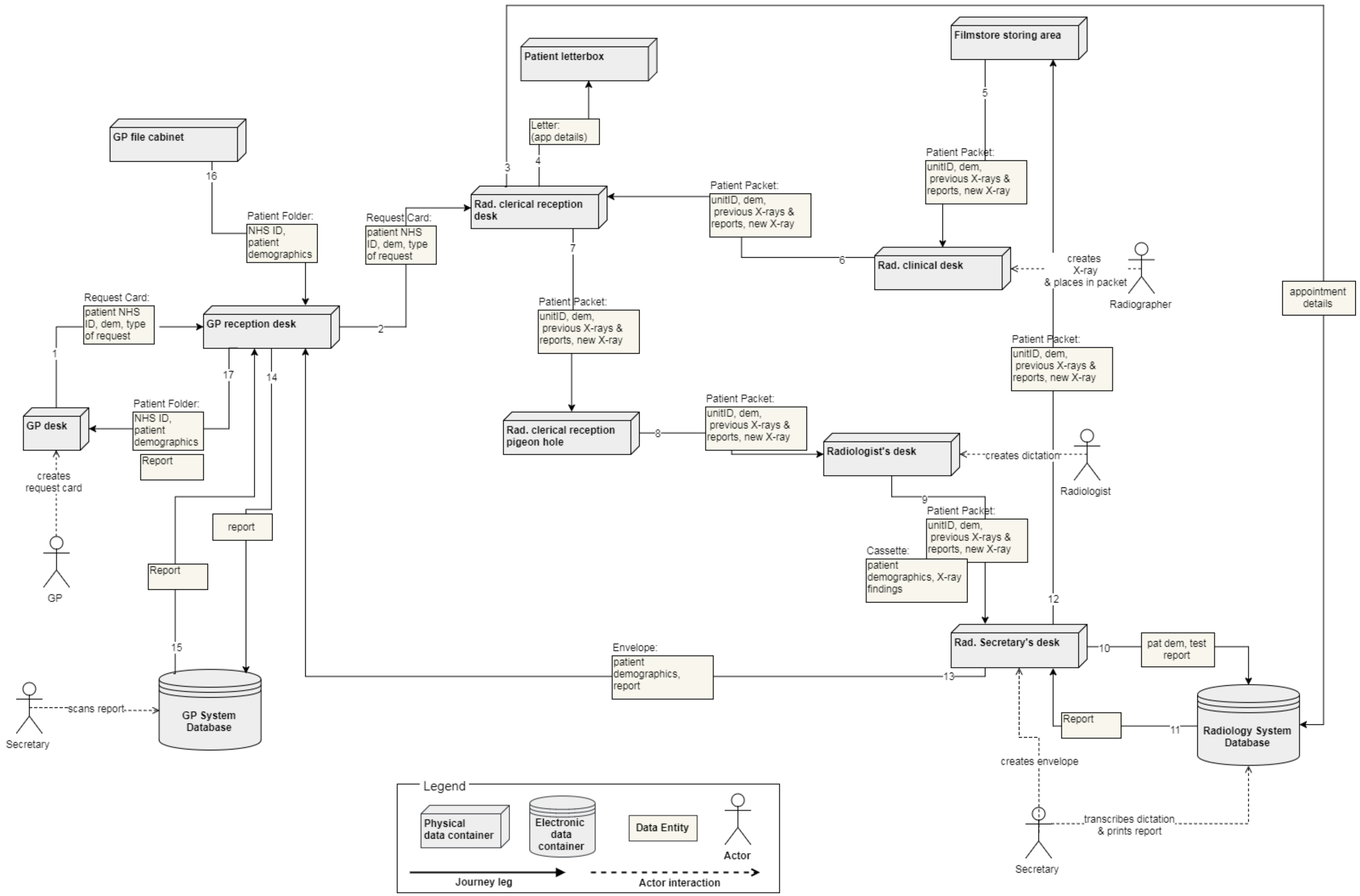

GP file cabinet
16
Patient Folder:
NHS ID, patient demographics
Request Card:
patient NHS ID, dem, type of request
1
GP reception desk
2
17
14
GP desk
Patient Folder:
NHS ID, patient demographics
Report
creates request card
GP
report
Report
15
Secretary
scans report
GP System Database
Request Card:
patient NHS ID, dem, type of request
Rad. clerical reception desk
3
4
Letter:
(app details)
Patient letterbox
7
Patient Packet:
unitID, dem, previous X-rays & reports, new X-ray
Rad. clerical reception pigeon hole
8
Patient Packet:
unitID, dem, previous X-rays & reports, new X-ray
Radiologist's desk
creates dictation
Radiologist
9
Patient Packet:
unitID, dem, previous X-rays & reports, new X-ray
Cassette:
patient demographics, X-ray findings
Patient Packet:
unitID, dem, previous X-rays & reports, new X-ray
6
Rad. clinical desk
creates X-ray & places in packet
Radiographer
Patient Packet:
unitID, dem, previous X-rays & reports, new X-ray
5
Filmstore storing area
Patient Packet:
unitID, dem, previous X-rays & reports, new X-ray
12
appointment details
Envelope:
patient demographics, report
13
Rad. Secretary's desk
10
pat dem, test report
Report
11
Radiology System Database
creates envelope
Secretary
transcribes dictation & prints report
Legend
Physical data container
Electronic data container
Data Entity
Actor
Journey leg
Actor interaction

Figure 8.1: The journey of the data needed by a GP when a patient has a fracture (old system).

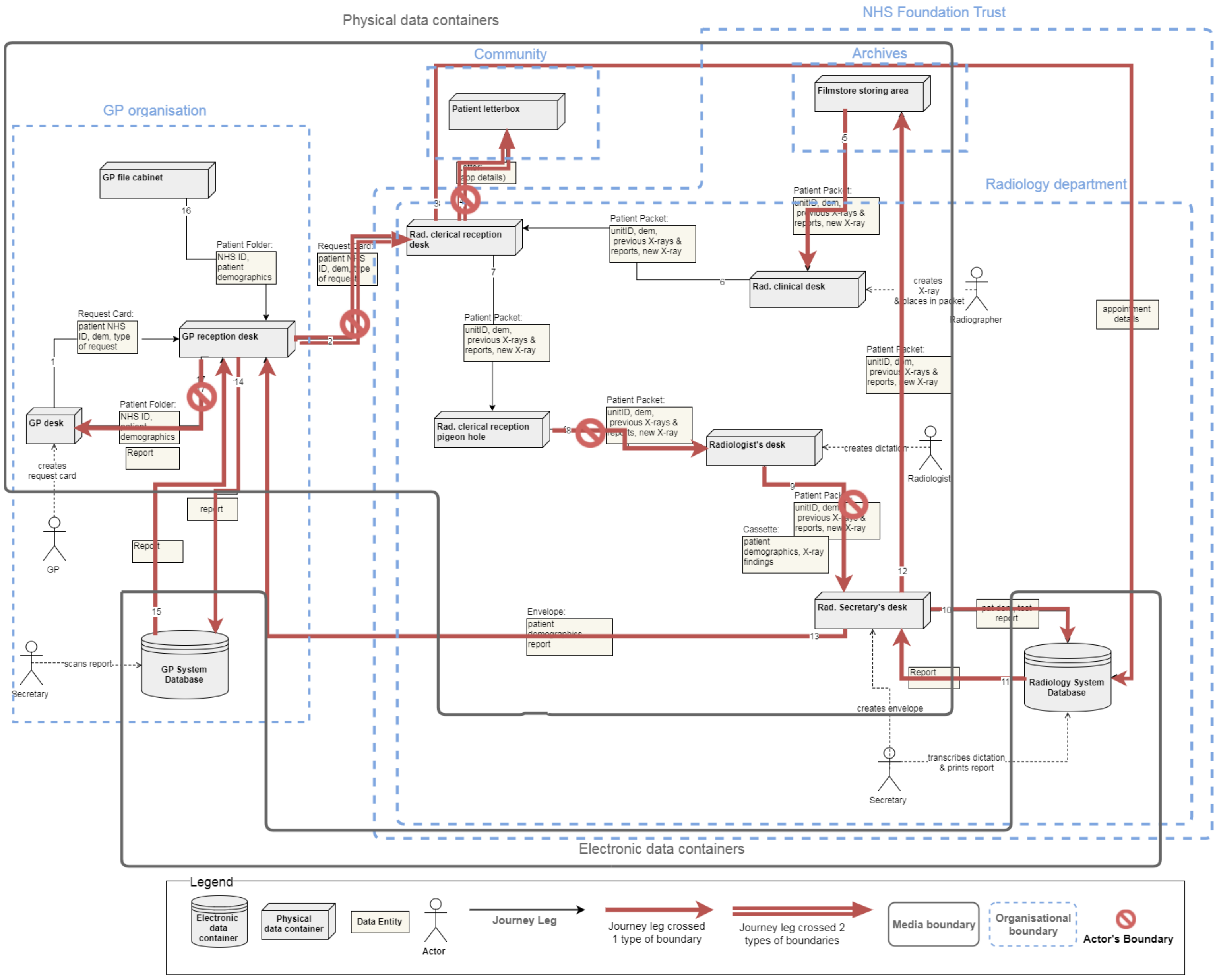

Physical data containers
NHS Foundation Trust
Community
Archives
GP organisation
Radiology department
GP file cabinet
Patient letterbox
Filmstore storing area
Rad. clerical reception desk
Rad. clinical desk
GP reception desk
GP desk
Rad. clerical reception pigeon hole
Radiologist's desk
Rad. Secretary's desk
GP System Database
Radiology System Database
Patient Folder: NHS ID, patient demographics
Request Card: patient NHS ID, dem, type of request
Patient Packet: unitID, dem, previous X-rays & reports, new X-ray
Cassette: patient demographics, X-ray findings
Envelope: patient demographics report
appointment details
app details)
Report
report
creates request card
creates X-ray & places in packet
creates dictation
creates envelope
transcribes dictation & prints report
scans report
GP
Secretary
Radiographer
Radiologist
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
Electronic data containers
Legend
Electronic data container
Physical data container
Data Entity
Actor
Journey Leg
Journey leg crossed 1 type of boundary
Journey leg crossed 2 types of boundaries
Media boundary
Organisational boundary
Actor's Boundary

Figure 8.2: Heatmap of the organisational, containers media and actors role boundaries overlaid on the old system's data journey model.

Journey Leg	Crossed boundary	Likely cost and risk
2	Organisational and Actors role	Data move away from the GP organisational unit to the Foundation Trust indicating the cost of complying with data sharing agreements, governance and ethical guidelines. Also, data created by the GP are used by another actor of different position role (the radiology secretary), implying a risk of clash of grammars, leading to lower quality of data moved to the target.
3	Containers media	Data move from the physical container of the clinical reception desk to the electronic container of the radiology's system database causing a data entry cost. There is also the risk of injecting errors that can reduce the quality of the information at the target side.
4	Organisational and Actors role	Data move between two organisations, the FT and the community indicating the cost of complying with data sharing agreements, governance and ethical guidelines. Also, data created by the secretary of the FT are used by a user of different role (the patient) risking clash of grammar and lower data quality.
5	Organisational	Data move from the archives department of the FT to the radiology department indicating the cost of complying with data sharing agreements, governance and ethical guidelines.
8	Actors role	Data created by the radiographer are used by another actor, the radiologist. The risk of clash of grammars exists that can cause lower data quality to the radiologist.
9	Actors role	Data created by the radiologist are used by another actor, the secretary. The risk of clash of grammars exists that can cause lower data quality to the secretary.
10	Containers media	Data move from the physical container of radiology secretary's desk to the electronic container of the radiology's system database causing a data entry cost. There is also the risk of injecting errors that can reduce the quality of the information at the target side.
11	Containers media	Data move from the electronic container of the radiology's system database to the physical container of the radiology secretary's desk. There is the cost of printing and transferring the data to the target side.
12	Organisational	Data move from the radiology department of the FT to the archives department indicating the cost of complying with data sharing agreements, governance and ethical guidelines.
13	Organisational	Data move from the radiology department of the FT to the GP organisation indicating the cost of complying with data sharing agreements, governance and ethical guidelines.
14	Containers media	Data move from the physical container of the GP reception desk to the electronic container of the GP's system database causing a data entry cost. There is also the risk of injecting errors that can reduce the quality of the information at the target side.
15	Containers media	Data move by the electronic container of the GP's system to the physical container of the secretary's desk. There is the cost of printing and transferring the data to the target side.
17	Actors role	Data created by the GP's system are used by an actor of different role, the GP. There is a risk of clash of grammars that can cause lower data quality to the radiologist.

Table 8.1: Cost and risk identified by the method.

time and resources in filing, capturing, and transferring the films and the patient's packet. Also, there was a small risk of losing the films. In the new system the old X-ray machinery was replaced with state-of-the-art electronic equipment that captures and stores X-ray images in digital form. X-ray images are now uploaded to the PACS system, and are no longer printed on X-ray films. The new digital images can be easily modified to highlight and magnify the area of interest, are quick and easy to transfer around the hospital, and can be in more than one place at the same time.

The PACS system is integrated with the Computerised Radiology Information System (CRIS) responsible for receiving referrals, booking appointments, and managing patients. CRIS replaced the old radiology system and is fully integrated with key hospital information systems such as the Patient Administration System (PAS), the Order Communications, and the Electronic Patient Records system (EPR). The data journey model of the new system in place is given in figure 8.3 (page 125).

To evaluate the predictions of our method when applied on the old data journey model, we had to identify the improvements hospital staff made to the journeys of the data. To do so, we compared each journey leg of the old model before the improvements were made by hospital staff, with the new data journey model to find whether the predicted costly movements have been removed or replaced in the new system. Table 8.2 shows the differences between the two models noting the changes made to the movements of the old system and the corresponding new journey legs.

8.2.6 Results and Discussion

In this section we present the results of this evaluation exercise and discuss what they tell us about the accuracy of our method at identifying places of cost and risk in a data journey model. By comparing the old data journey model with the new model we identified the set of journey legs that hospital staff assessed as costly and replaced in the new model to reduce the overall costs of the system. We assessed a prediction as accurate if the predicted journey leg changed in the new model on the assumption that the hospital staff would not have gone to the trouble and expense of making changes if they did not see an opportunity to cut costs/risks, since no new functions were to be supported by the new version of the system.

While it seems reasonable to assume that hospital staff correctly identified potential cost saving improvements we cannot also assume they found and implemented all cost saving opportunities that might exist (i.e. due to hospital staff limited time and resources). We therefore cannot assess a prediction to be incorrect if it does not match

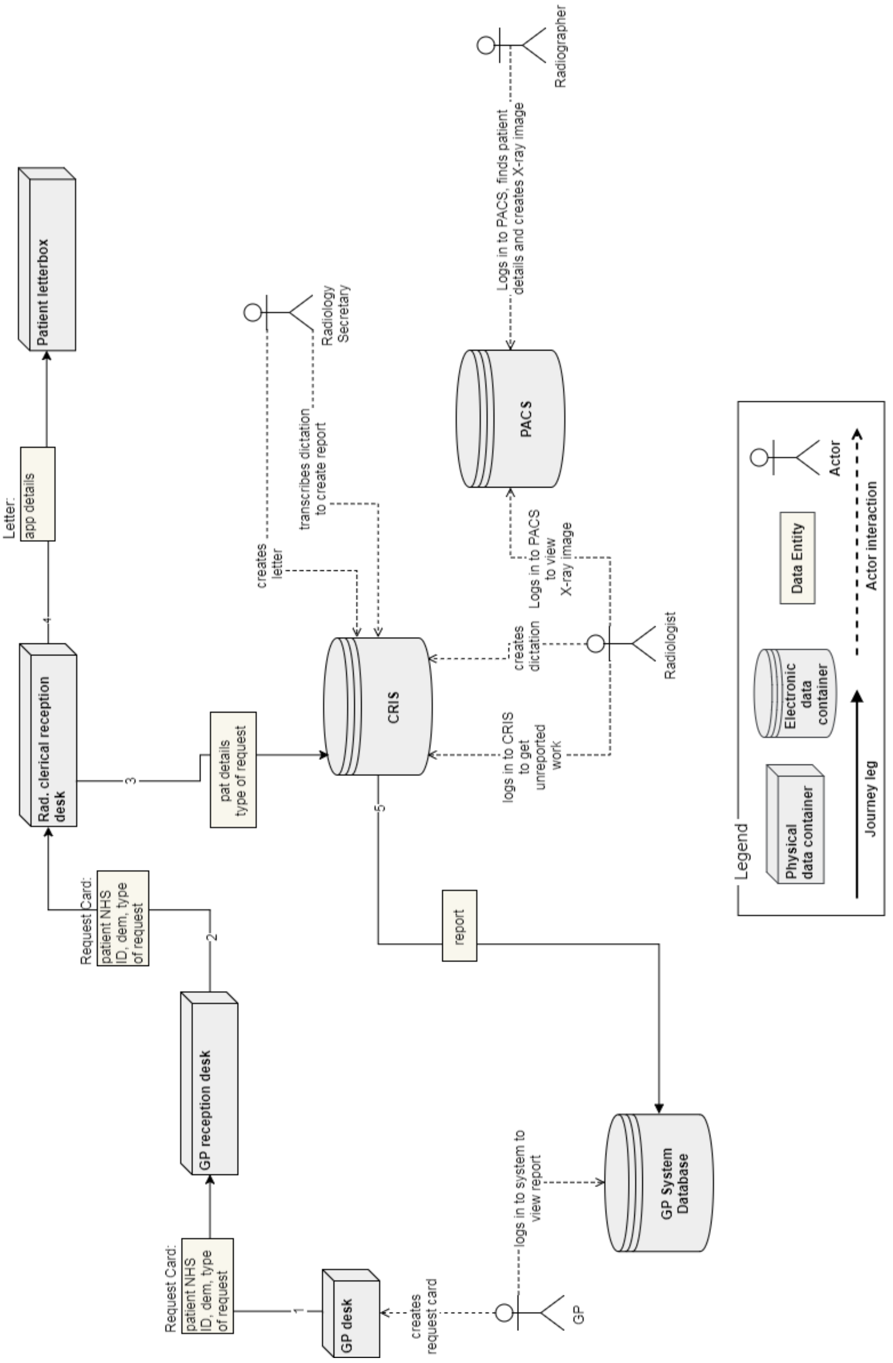

GP desk
1
Request Card:
patient NHS ID, dem, type of request
GP reception desk
2
Request Card:
patient NHS ID, dem, type of request
Rad. clerical reception desk
3
pat details
type of request
4
Letter:
app details
Patient letterbox
CRIS
5
report
GP System Database
PACS
GP
creates request card
logs in to system to view report
Radiology Secretary
creates letter
transcribes dictation to create report
Radiologist
logs in to CRIS to get unreported work
creates dictation
Logs in to PACS to view X-ray image
Radiographer
Logs in to PACS, finds patient details and creates X-ray image
Legend
Physical data container
Electronic data container
Data Entity
Actor
Journey leg
Actor interaction

Figure 8.3: Data journey model of the new hospital system.

Old leg	Changes made by hospital staff	New leg
1	No change.	1
2	No change.	2
3	Different target container, the radiology system is replaced by CRIS system.	3
4	No change.	4
5	Leg removed by replacing physical packet with electronic data saved in PACS.	-
6	Leg removed by replacing physical packet with electronic data saved in PACS. Radiographer creates electronic X-ray image in PACS.	-
7	Leg removed by replacing physical packet with electronic data saved in PACS.	-
8	Leg removed by replacing physical packet with electronic data saved in PACS.	-
9	Leg removed by replacing physical cassette with electronic data saved in CRIS. Radiologist accesses patient details through PACS, and dictates report in CRIS.	-
10	Leg removed by replacing physical cassette with electronic data saved in CRIS. Secretary accesses dictation and transcribes report in CRIS.	-
11	Leg removed. The report is not printed, as will be electronically sent to the GP.	-
12	Leg removed by replacing physical packet with electronic data saved in PACS.	-
13	Leg replaced by using electronic report sent directly to the GP system.	5
14	Leg replaced by using electronic report sent directly to the GP system.	5
15	Leg removed. GP accesses report directly from the GP system.	-
16	Leg removed. Archives still exist, but not used in everyday processes.	-
17	Leg removed. GP accesses report directly from the GP system.	-

Table 8.2: Changes made to the old system by hospital staff.

the changes made by the hospital staff.

To evaluate the accuracy and feasibility of the rest of our predictions we asked the two NHS domain experts to review them. The experts assessed predictions as 'valid' if they could see a clear potential for cost saving if the predicted journey leg was eliminated, and otherwise as 'not valid'. We also asked the domain experts to comment on the cost and risk they faced while working with the old system.

Table 8.3 gives the assessment of our method's predictions in this retrospective evaluation. It shows whether the predicted costly journey legs were removed or replaced in the new model. For each prediction, the table presents the views of the NHS domain experts on the cost and risk imposed by the respective boundary, along with an assessment of the validity of each prediction. Finally, we give a summary assessment of the accuracy of each prediction. We categorise each final assessment in the table as follows:

True Positive (TP) A prediction is assessed as a TP if the predicted costly leg was either removed in the new model *or* assessed as 'valid' by the domain experts.

False Positive (FP) A prediction is assessed as a FP if the predicted costly leg was not removed in the new model *and* the domain experts assessed it as 'not valid'.

False Negative (FN) A journey leg is assessed as a FN if it was not included in our predictions but the domain experts foresaw significant likely cost and risk associated with it.

True Negative (TN) A journey leg is assessed as a TN if it was not included in our predictions *and* it was not changed in the new model *and* the domain experts did not foresee any likely cost and risk associated with it.

Old journey leg	Crossed boundary	Journey leg removed in new model?	Domain experts' assessment	NHS domain experts view on cost and risk	Final assessment
1	None	-	-	-	TN
2	Org.	No	Valid	There is a transportation and postage cost transferring the request card from the GP to the radiology department.	TP

2	Actors role	No	Valid	Since the request card is filled in by a different person than the one using it, there might be a missing information cost. In the case of an uncompleted request card (happens frequently), the process is disrupted. Costs include time to call the GP chasing missing information, time and effort to complete another request card, resources of the replacement card. Also, these extra costs imposed on the process can presumably damage the reputation of the GP and FT.	TP
3	Containers media	No	Valid	Inputting data into the system requires time and effort of clerical staff. Also, it introduces the risk of injecting errors. The risk of injecting errors into the system is higher since data were created by a user other than the hospital clerical staff (i.e. the GP secretary). Also, there is duplication cost since data already existing on the request card are duplicated into the system.	TP
4	Org.	No	Valid	There is a postage cost imposed when sending the letter to the patient. Resources are needed like paper, stamps, envelopes and the postal franking machine.	TP
4	Actors role	No	Valid	Time and effort of the clerical staff to print and prepare the letter. Also, since the appointment date and time are selected by the radiology secretary but actually used by the patient, there is the risk of the unavailability of the patient and the cost related with the patient cancelling or rescheduling the appointment.	TP
5	Org.	Yes	Valid	There are searching and transferring costs. Time and effort of clerical staff to find the patient's packet from the archives and transferring it to the radiology clinical area. Also, resources are needed to transfer the big packets across units, mostly by using a trolley.	TP
6	None	-	-	-	TN
7	None	-	-	-	TN
8	Actors role	No	Not valid	The radiographer creates the X-ray image which is then used by the radiologist. Mistakes made because of the different actors are possible, but unlikely.	FP

9	Actors role	No	Valid	The dictation is created by the radiologist but transcribed into the system by the secretary. This introduces the cost of mistakes inputted into the system. This is actually a common phenomenon since secretaries do not share the same knowledge and experience of radiologists to always fully comprehend what they meant to say in the dictation.	TP
10	Containers media	Yes	Valid	There is the cost of the time and effort of the secretary typing data into the system and the risk of injecting mistakes and errors.	TP
11	Containers media	Yes	Valid	There is the stationery, time and effort cost of the secretary printing the report.	TP
12	Org.	Yes	Valid	There is a transportation cost transferring the packet with the patient's information back to the archives. Time and effort of clerical staff are needed. Also, resources like the trolley are required.	TP
13	Org.	No	Valid	There is a transportation cost transferring the enve lope containing the report to the porter's area of the FT, staff time and effort.	TP
14	Containers media	Yes	Valid	There is the cost of the time and effort needed by the GP secretary to scan the letter into the GP system. Also, there is the risk of duplicating information already existing at the FT.	TP
15	Containers media	Yes	Valid	There are printing costs.	TP
16	None	-	-	Cost of retrieving and filing patient records.	FN
17	Actors role	No	Not valid	The report is created by the FT radiologist which is now used by the GP. Often, there is the cost of a follow up appointment with the patient. However, this is not caused by the actors barrier.	FP

Table 8.3: Assessment of the accuracy of our method at predicting costly/risky places.

From the table above, we see that the old data journey model has 17 journey legs. 13 of the 17 legs cross a boundary, and hence were predicted as costly. However, two legs crossed more than one boundary. In the above table, those two legs are presented as separate predictions (since only one of them might be valid) making the total number of predictions 19.

More than half of our predictions (13 out of 19) are assessed as true positives. Of these, 6 were removed in the new model, while the remaining indicate boundaries that still exist in the current system and were assessed by the NHS domain experts as 'valid' predictions.

Of the 19 predictions, two were assessed as false positives, since they were not removed in the new model and the domain experts assessed them as 'not valid'. Although our method predicted them as costly, the costs that the domain experts have identified for the respective journey legs did not arise from the crossed boundaries. Both journey legs crossed the actors boundary; data moved from the radiographer to the radiologist and from the radiologist to the GP. Although these actors have different pay scales, they share the knowledge and experience needed to comprehend the information shared among them. This indicates a limitation of the salary band proxy used to identify the cost of clash of grammars between actors and points to a need for further research. However, we do not expect such a easy-to-apply method to work with high accuracy. Importantly, the actors pay scale was a low-cost proxy, any other proxies pursued which might produce fewer false positives will only be a good replacement if it is as easy and lightweight to apply as the actors pay scale.

Four out of 17 journey legs had no barrier predicted. The domain experts did not identify any costs or risks related with three of those four legs. However, we have no evidence that these journey legs result in no cost at all. Since there will always be room for optimisation we can not have an accurate percentage of true negatives. However, the domain experts did find additional cost and risk in one of these four legs (journey leg number 16). Leg 16 was assessed as a false negative, since it crossed no boundary, but domain experts identified a cost of retrieving, filing and transferring the patient folder. Our method did not identify the cost of transferring physical data, since it only captures costs of transforming data from physical to electronic format and vise versa. Other transfer costs in the model (e.g. journey legs 2 and 5) were captured by the organisational boundary, as different organisational units tend to be in different physical locations, suggesting further exploration is needed on the intersection of organisational and transportation costs. Table 8.4 shows the ratio of true positives, false positives, true negatives and false negatives in our predictions.

	Positive	Negative
True	68%	16%
False	11%	5%

Table 8.4: Contingency Table Showing Evaluation Results

To assess our methods ability to distinguish between different degrees of cost/risk, we asked the domain experts to comment on the most costly/risky prediction that if eliminated will have the biggest long term benefit for the organisation. Both domain experts *independently* assessed actors boundary on journey leg 2 to be the most costly. This is one of the two journey legs (numbers 2 and 4) that our method predicted to have crossed the most boundaries and hence be most costly, as shown in the heatmap (figure 8.2). The prediction of journey leg 4 was also ranked high by the experts (fourth and fifth most costly leg).

Our method identified a majority of the journey legs of the old model (14 out of the 17) as involving physical media. Physical legs have significant stationery, printing and transportation costs as well as the risk of data quality loss (duplication, timeliness, incompleteness, consistency, etc). This type of cost was the one targeted by the hospital staff and was reduced to only three physical legs in the new model, as can be seen in the new journey model (figure 8.3).

According to the domain experts, another significant cost is imposed on the organisation when journey legs cross organisational boundaries. Five journey legs of the old model crossed organisational boundaries, of which only two did not have any governance issues (numbers 5 and 12), since they move data within the same hospital unit. According to the experts the legs with no governance barrier were the ones to be eliminated because governance is one of the hardest and most complex boundary to remove. Organisational barriers are one of the hardest to resolve because of the myriad of governance regulations that organisations need to comply with, especially when the data moved include sensitive patient information. Change is easier and quicker within an organisational unit than across. Also, the three journey legs in the GP organisation of the old model were not changed in the new model, for the obvious reason that the changes were driven by hospital staff in the radiology department who had no jurisdiction to enforce change at the GP organisation.

Finally, we interviewed the domain experts on the types of cost and risk that affect their organisation. Their comments reflect on the findings of our method:

> *"Costs are indicative of both the process and the actors involved in the process. For instance, costs would be higher in respect of processes involving higher skilled personnel, such as time taken by GPs to complete request cards at their practice, and the actual X-Rays performed by the Radiographers. Similarly, from an administrative perspective, costs would be higher where either duplication or other errors are introduced (due to*

> *mis-communication, mis-understandings, etc) by medical secretaries. At the lower end of the cost spectrum would be the lesser skilled personnel such as clerical staff and porter staff to transport data/information from different end-points such as from the GP practice to the Secondary care facility (i.e. Acute hospital)." - NHS Domain Expert #1*
>
> *"The majority of the total cost is the human resources. For example the time the GP needs to fill in the request card, and so on. Another major cost is the acquisition and maintenance of clinical and clerical equipment, like the X-ray machine, cassettes, etc. The least expensive costs to the organisation are the stationery and printing costs." - NHS Domain Expert #2*

A property of our method is to identify costly places in a lightweight manner. Although, more information invested in the method will likely provide more accurate results, the time required to acquire this information will be taken away from the domain experts', clinicians', or managers' other activities (time spent treating patients). Our aim was to develop a lightweight method that does not demand much time to be spent on acquiring needed information, designing the models, applying the prediction method, interpreting and checking the results. Instead, we looked for information that is easy to acquire, but also points out areas of cost and risk. Because of the lightweight property of our method we do not expect all our predictions to be accurate or complete. However, every inaccurate prediction when reported to potential managers or stakeholders has the cost of further investigating it. Therefore, as a pragmatic minimal success criteria we consider the model to be accurate if it produces *more* correct predictions than wrong predictions. The results of our evaluation showed that we obtained significantly more correct predictions than false ones.

Apart from the accuracy of the predictions, another criterion of the success of our method was the time needed to collect required information and the effort put to identify barriers. Records of our interviewing times with the NHS domain experts show that they are within our set threshold of one working day. We had three meetings of an hour each with both experts present (domain expert #1, and #2), and two further one-hour meetings with just one of them (expert #2), totalling 8 hours of combined time. However, the records of the time spent modelling turned out not to be useful. This is because we refined the model while working on the case study, and so ended up repeating a significant amount of the modelling work. We cannot therefore assess the success of our method against this criterion from this study. Later work has suggested that modelling does not add much to the time needed to acquire the domain information, and indeed can be carried out by domain experts themselves after less than one hour's training [66].

Results on both time taken and accuracy of the predictions are within the set thresholds, indicating that our method is lightweight and can identify places of high cost and risk in the journey of data among organisations. Table 8.5 summarises our success criteria, and actual times and results.

	Success Criteria	Actual outcomes
Accuracy	The accuracy of the predictions. At least 50% of the predictions must be TP, while FP predictions must be fewer than TP.	13 out of the 19 predictions were TP (68%), while only two were FP (11%). The rest 21% were TN and FN.
Time	The time invested by the staff of the organisation to give us domain information. We set the threshold to one working day.	We had three meetings of approximately one hour each with NHS domain expert #1, and five one-hour meetings with domain expert #2.

Table 8.5: Evaluation against success criteria

8.3 Evaluation of Stability of the Core Boundaries

Having evaluated the accuracy of our method in the previous section, in this section we assess whether our method is useful in settings other than the ones used before. We are investigating whether our method's three core boundaries identified in the previous chapter are capable of identifying places of high cost and risk in other domains. Will each new setting require a new set of specific boundaries to identify places of cost/risk? If the data journey modelling method is to be truly low cost to apply, then we need to have a stable set of boundaries that can be used and give good results across many domains.

To answer above questions, we worked with health care practitioners (HCPs) in the new area of Clinical Genomics. Clinical genomics is a branch of medicine in which the genome of the patient is sequenced, and interpreted by a multi-skilled team of experts, in order to assist with diagnosis of hereditary conditions, inform treatment decisions, and to determine the likelihood of conditions or symptoms appearing in the future [59].

The Clinical Genomics patient pathway[1] is more complex than those we have

[1] A "patient pathway" is the route that a patient takes from their first contact with an NHS member

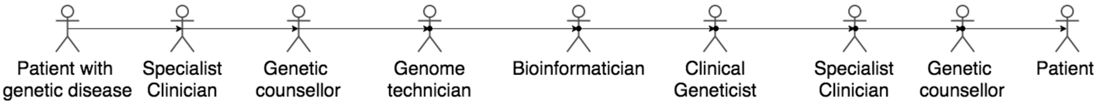


Figure 8.4: Actors' pipeline in the genetic testing patient pathway.

worked with before, as it requires sharing of a variety of information between numerous actors with very different skill sets, coming from different organisational departments. Figure 8.4 shows the complexity of the interactions that occur throughout the genetic testing patient pathway. A detailed description of the processes are given in Appendix B.1 (page 177), along with a typical example of the pathway processes represented using swimlane diagram (figure B.1 on page 179).

During the period covered by this evaluation, we were able to gain access to three separate groups of HCPs in this domain, working in numerous hospital foundation trusts across the UK, and having quite different roles in the clinical genomics patient pathway (varying from managerial positions to specialist practitioners along the pathway, and IT developers).

We begin by presenting the methods and the success criteria we used in the studies. Then, we describe the three NHS case studies in which we assess the stability of our method.

8.3.1 Methods

In the following three studies, we followed the approach described in section 4.5, a method inspired by action research, but adapted to fit the circumstances of the opportunities we had to work with domain experts, some of which arose at short notice. In each case study, we began by defining a procedure to follow, and ensure to only document domain expert responses, and not to lead them to answers we might have wished they had given.

Our research question for this group of studies is: are the core boundaries stable across domains? That is, if used alone, can our core boundaries identify some costs/risks deemed significant by the domain experts in domains different from the ones from which they were originally identified?

of staff (usually their GP), through referral, to the completion of their treatment. It can also cover the period from entry into a hospital or a treatment centre, until the patient leaves.

8.3.2 Success Criteria

In setting out to answer our research question, we did not require that the core boundaries find *all* cost/risk points, only that significant ones could be identified. We do not expect our model to identify all the points where cost/risks can be reduced, since that would require detailed and complete modelling, and more investment from the HCPs. Instead, we are evaluating whether our boundaries can cheaply and quickly identify places where savings can be made without investing extreme resources (i.e. time, effort, money). We define the following success criteria to assess our method's stability:

Boundary Accuracy For a boundary to be considered stable in the new domain, must identify at least 50% true positive predictions (points of cost/risk that the hospital staff has identified as costly/risky, or agreed could be promising places for cost savings) in each of the three case studies.

Method Stability For our method to be considered stable in the new domain, *all* three boundaries must be stable in accurately predicting points of cost/risk.

8.3.3 Clinical Genomics Study A – Patient Phenotype

In the most extensive study of the three, we worked with staff in a Genomic Medicine Centre department of a Foundation Trust hospital in Greater Manchester, which had recently undertaken a re-engineering project to improve the efficiency of their processes.

In this part of the pathway, the phenotype of the patient is captured by a clinician. Phenotypic information is gathered from retinal images using specialist devices. The clinician examines these images and writes a report capturing the phenotype of the patient as hand written notes. Clerical staff then type up the reports.

8.3.3.1 Study Design

To evaluate the stability of our method's boundaries in this domain, we carried a retrospective analysis of the data journeys needed to capture the patient's phenotype (the set of observable characteristics of the patient's genotype) as described in section 4.5. We modelled the journeys of data *before* and *after* the information infrastructure redesign to assess the predictive power of our method's boundaries, using the following approach:

1. We conducted semi-structured interviews with an NHS domain expert working as a Bioinformatics Developer at the Genomics department of the trust, to gather domain knowledge on the journeys of data. To do so, we followed a similar approach to previous study. We started with the opening question ‘Can you describe how you use the system(s) to examine phenotype images and write the final report?’ and then progressed to some predetermined interview questions aiming to gather domain knowledge on the data movements, organisational structures, available information systems, data formats and staff roles in the FT.

2. We modelled the *old* data journey before any improvements were made by hospital staff and overlaid our core boundaries to identify the journey legs with likely high cost and risk.

3. Then, we modelled the data journey of the *new* system after the hospital re-engineering team improvement efforts.

4. Finally, we compared the identified costly journey legs of the old model with the staff’s improvements to assess the stability of our method’s boundaries in identifying places of high cost and risk.

8.3.3.2 Identifying Cost/Risk Before Re-engineering

Before the re-engineering phase, several technical and social challenges caused considerable costs to the department. Diagnostic data were stored on a very old computer not connected to the network. To retrieve data, staff had to physically go to the room where the devices are stored and retrieve data using a USB drive, or sometimes even a floppy disk. Moreover, patient phenotype information was often needed by bioinformaticians and other actors working in the pathway, but no official sharing protocol existed, obstructing the dissemination of vital information. Another expensive issue was that phenotype data were not always coded (converted into standard medical codes) on input to the local computer system, making machine processing quite challenging.

Interviewing the NHS domain expert, we gathered knowledge needed to create the data journey model of the old system prior to the re-engineering phase, such as the IT systems in place, the processes that move data across the organisation, and the people who use those data to create some value from the journey. Figure 8.5 [2] shows the data

[2] As with many governmental institutions, aspects of this case study are confidential. Although the models used to evaluate our boundaries are based on the actual case study, here we present a more general model, typical of those used across the NHS.

journeys needed to collect information from the imaging devices, send to the specialist clinician to write the phenotype report which is then sent to clerical staff to transcribe into the local genomics system.

Once the domain expert approved the model, we overlaid on it the three boundaries of our method to identify the journey legs with likely costs/risks. Figure 8.6 shows the data journey model and boundaries prior to the re-engineering phase. The organisational boundaries are denoted with a green solid colour line, the change of media boundaries with a red dotted line, and the change in pay-scale boundary with a large red warning sign.

The journey legs that cross any of the boundaries indicate places where the model predicts that movement of data to a context where the portability of the data can be reduced and costs/risks can be imposed on the journey. These are journey legs 6, 7, 8, 9, and 10, indicated in the figure with a red warning sign. The place of highest cost/risk as identified by our method is journey leg no 8, since it crosses two of the three boundaries (organisational and media boundaries).

8.3.3.3 Modelling the New System

Having predicted the problematic journey legs of the old system, we worked with the domain expert to model the current data journeys happening after the re-engineering project. Old data journeys were replaced by a new computer system for storing patient phenotype data. An on-line questionnaire guides the specialist clinician to capture the patient's phenotype, which is automatically coded and stored in a central database for access by the genomics team. Data captured by the retinal imaging devices are also uploaded to the central database. Figure 8.7 shows the data journeys in place after the introduction of a new computer system for storing patient phenotype data.

8.3.3.4 Study Results

As shown in table 8.6, comparing the two models, we see that in all but one of our predictions the predicted costly journey legs in the old model have been removed in the current model. The one that remains (journey leg #8) is under the control of the wider Foundation Trust, and according to the domain expert is harder to optimise, because of the governance restrictions imposed by the organisational boundary. However, the media boundary of the leg has been removed.

In total, 83% (five of six) of the predictions made by the model were confirmed as true positives by the domain expert based on the definitions of TP and FP we set in the

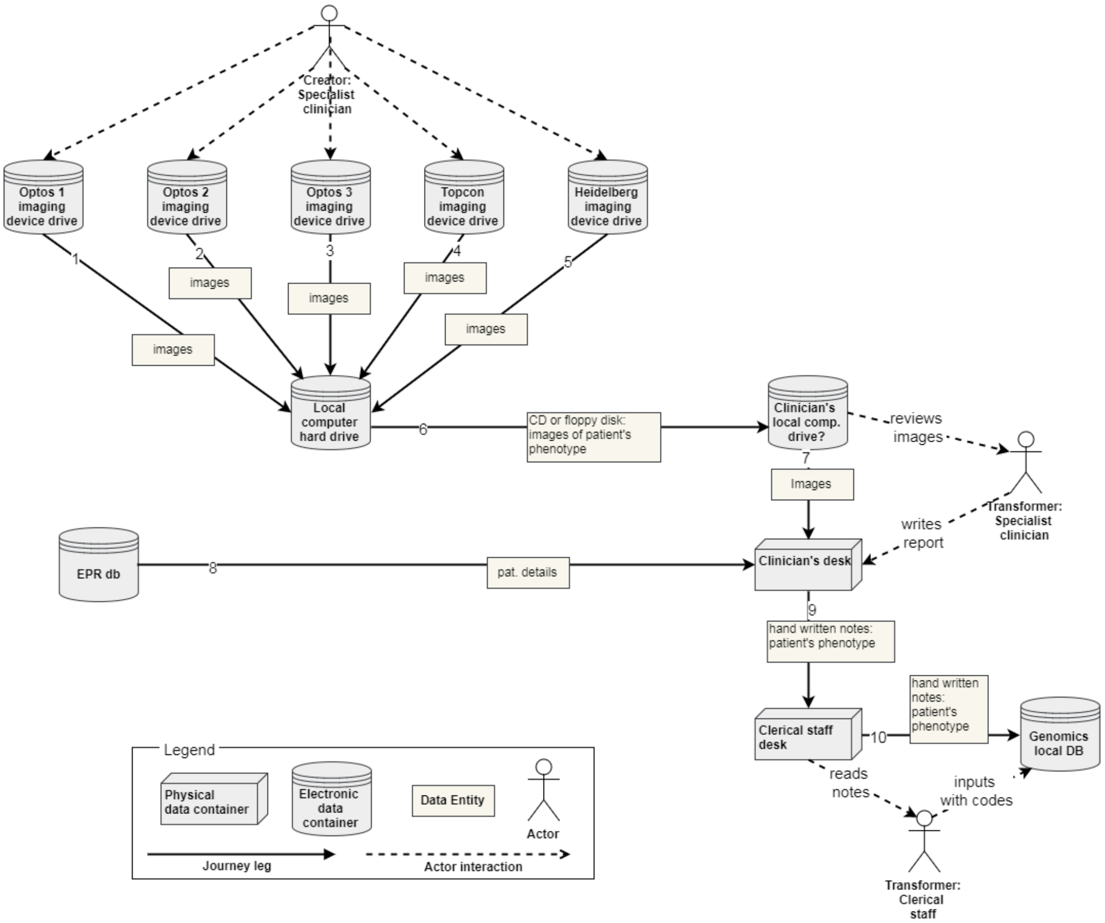

Creator: Specialist clinician
Optos 1 imaging device drive
Optos 2 imaging device drive
Optos 3 imaging device drive
Topcon imaging device drive
Heidelberg imaging device drive
1
2
3
4
5
images
images
images
images
images
Local computer hard drive
6
CD or floppy disk: images of patient's phenotype
Clinician's local comp. drive?
reviews images
Transformer: Specialist clinician
7
Images
writes report
EPR db
8
pat. details
Clinician's desk
9
hand written notes: patient's phenotype
Clerical staff desk
10
hand written notes: patient's phenotype
Genomics local DB
reads notes
inputs with codes
Transformer: Clerical staff
Legend
Physical data container
Electronic data container
Data Entity
Actor
Journey leg
Actor interaction

Figure 8.5: Data journey model *before* the re-engineering process.

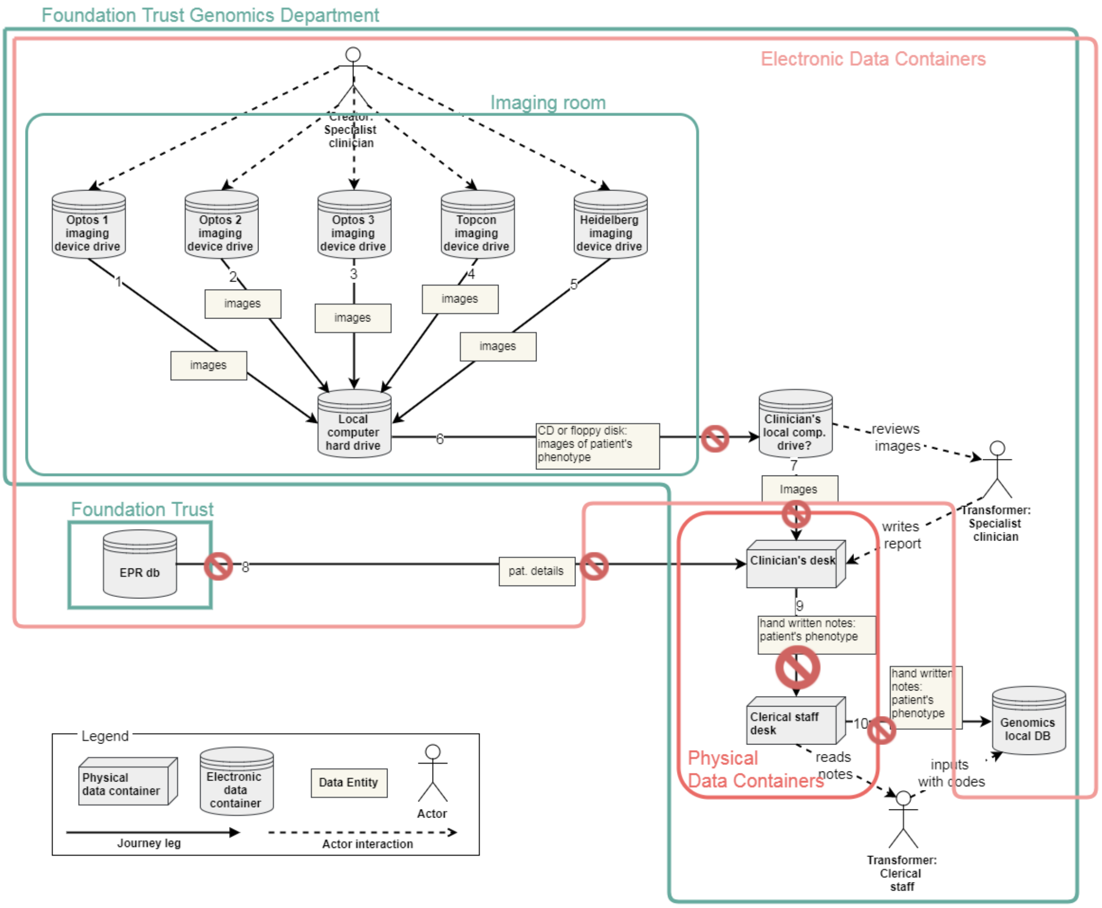

Foundation Trust Genomics Department
Electronic Data Containers
Imaging room
Creator: Specialist clinician
Optos 1 imaging device drive
Optos 2 imaging device drive
Optos 3 imaging device drive
Topcon imaging device drive
Heidelberg imaging device drive
1
2
3
4
5
images
images
images
images
images
Local computer hard drive
6
CD or floppy disk: images of patient's phenotype
Clinician's local comp. drive?
reviews images
7
Images
Transformer: Specialist clinician
writes report
Foundation Trust
EPR db
8
pat. details
Clinician's desk
9
hand written notes: patient's phenotype
Clerical staff desk
10
hand written notes: patient's phenotype
Genomics local DB
Physical Data Containers
reads notes
inputs with codes
Transformer: Clerical staff
Legend
Physical data container
Electronic data container
Data Entity
Actor
Journey leg
Actor interaction

Figure 8.6: Data journey model heatmap *before* the re-engineering process.

previous hospital study (described in section 8.2.6). In the remaining one (journey leg #8) the domain expert agreed that cost could have been saved resulting to 6 of the 6 predictions to be true positives. We found no false positive predictions in this study.

Thus the three standard boundaries *were* stable and useful in this new domain, able to identify the points of cost and risk that hospital staff identified and replaced.

8.3.4 Clinical Genomics Studies B and C

The next two studies were undertaken in conjunction with four Clinical Consultant Managers from four different Genomics teams across the UK (all of which were unconnected with the Clinical Genomics study A). The NHS staff members were participating in the doctoral level academic programme for Higher Specialist Scientist Training (HSST) at the University of Manchester. All were facing challenges in implementing new functionality in their trusts, to either integrate systems, migrate information to

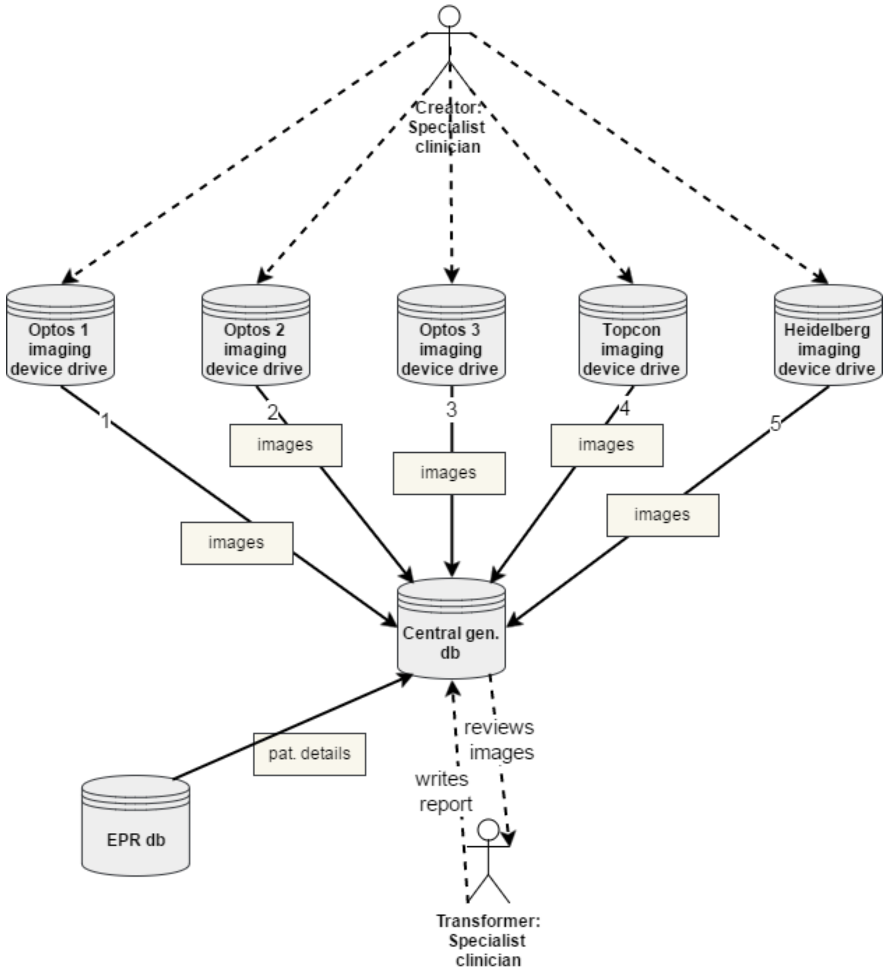

Creator: Specialist clinician
Optos 1 imaging device drive
Optos 2 imaging device drive
Optos 3 imaging device drive
Topcon imaging device drive
Heidelberg imaging device drive
1
2
3
4
5
images
images
images
images
images
Central gen. db
EPR db
pat. details
reviews images
writes report
Transformer: Specialist clinician

Figure 8.7: Data journey model of the 'new' Genomics team processes

new servers, or expand existing functionality.

8.3.4.1 Study Design

To assess the stability of our method's core boundaries in accurately identifying points of cost and risk we held a half-day workshop with the Clinical Consultant Managers. We followed an iterative and interactive agile approach in organising the workshop, aiming to maximise time the consultants spend interacting with our modelling process, while minimise lecturing. We divided the workshop into four sessions (brainstorming challenges, data journey modelling, identifying costs/risks, and reflection) capturing consultant's feedback at the end of each phase. We went through the sessions following the process below (the actual time taken in each step is given in the parentheses):

1. Hold a brainstorming session in which consultants discussed challenges they

Prediction	Journey Leg	Crossed Boundary	Cost Removed	Domain Expert Assessment
1	6	Organisational	Yes	–
2	7	Media	Yes	–
3	8	Organisational	No	Valid
4	8	Media	Yes	–
5	9	Actors	Yes	–
6	10	Media	Yes	–

Table 8.6: Our method's identifications on points of likely costs/risks and assessment by domain expert.

face while introducing new functionality to their current infrastructures (30 minutes).

2. Introduced the new data journey model as a new lightweight, agile tool to map the information infrastructure of an organisation and the data journeys happening within, without mentioning the notion of boundaries (45 minutes).

3. Asked the clinical consultant managers to create models of data journeys of their choice happening in their departments, and reflect on the process and results (45 minutes).

4. Before yet mentioning the boundaries, we asked them to note on their models the journey legs they think are the most expensive (in terms of time, effort, and resources) based on their experiences (10 minutes).

5. Only then we described the use of boundaries to identify places of high cost/risk (20 minutes).

6. The consultants applied the three boundaries on their models to identify the places the boundaries predict as most costly/risky. We then compared the places that clinical consultants assessed as most costly with the places our boundaries identified to assess our method's findings (30 minutes).

7. Finally, we reflected on the results, process and model (15 minutes).

8.3.4.2 Brainstorming and Modelling Sessions

The group of four consultants worked in pairs throughout the sessions. From the brainstorming session each pair identified an area of concern in their respective departments, in which problematic data movements cause problems to their workflows.

We introduced to the consultants the data journey modelling technique (without mentioning the boundaries) and show them the working example of creating the pathology lab data journey model (figure B.2 in Appendix B.2). Then, working in pairs the clinical consultants created data journey models corresponding to areas of concern in their respective departments. Each model was thus an amalgam of behaviours in two different Clinical Genomics departments in the UK. Each team focused (by their own choice) on slightly different parts of the patient pathway. Study B, focused on the data journeys needed to collect information from several actors across the pathway for the bioinformatician to process. Study C, focused on the data journeys needed to collect variant information from several external resources, like Decipher and ClinVar. The data journey models produced by the Clinical Consultants are given in appendix B on pages 184 – 186 (with identifying details removed).

Interestingly, all participants agreed that even this first step of creating the data journey model was in itself valuable. Prior to the experience, they had had a bioinformatics-centric picture of the work in their departments, since that was the focus of their everyday practices. The data journey model helped them to gain a different perspective on the processes and interactions taking place within their teams. They quickly gained an overview of how their departments use and share data, when in their daily work they were more used to seeing only the detail of small parts of the journey which were more involved with.

In particular, the study B pair was surprised by the number of journey legs needed to collect required information for exome assembly, just one task in the pathway. The study C pair emphasised the external data sources that bioinformaticians rely on to feed their computational analysis pipeline. Before they started modelling, they planned to model the journeys data make from the external data sources to the bioinformatician's machine. However, while modelling these journey legs, they realised that the legs depend on a bigger set of journey legs. Having created the data journey model, they recognised the value of other actors in the pipeline in getting accurate information to work with. The journey model brought home to all of them the complexity of the interactions between people and systems involved in their respective departments.

8.3.4.3 Identifying Costs/Risks and Reflection Sessions

Having completed the data journey models, and before the boundaries had been added, we asked both pairs to identify the journey legs where most effort (cost) was expended, and where they were most concerned to see an efficiency gain. Having done this, we introduced the three core boundaries of our method and showed the participants how to layer the boundaries over the model and draw predictions from them.

In both cases, the points of cost/risk identified by the boundaries differed from the problematic legs already identified by the teams. But after discussion, both teams agreed that the predicted legs *were* sources of highest cost; they just had not been aware of them beforehand.

In case study B, the pair identified the bioinformatician's part of the model as the most costly leg, where the heavy data processing is happening. The boundaries, however, identified a single point of failure in their current infrastructure — the majority of the journeys started and ended at a single system, maintained by one particular staff member. They had not realised the dependence of their processes on this one person before this exercise. Having identified it, they can look at re-engineering their processes and remove the risk.

In case study C, when the pair was asked to comment on the current costs of the model, the pair identified as most costly points the legs moving data from the external sources to the bioinformatician's workplace, because of the need to pay an access fee for this data. While the boundaries identified these points of cost, they identified a different point as being of highest cost/risk. This was the point of information handover from the bioinformatician to the genetic counsellor, which crossed all three boundaries. After discussion, both HCPs realised that the costly leg identified by the method happens on a daily basis (i.e. the cost of moving data accumulates daily), whereas the area they had been concentrating on happened more rarely (a one-time expense per year), hence is less expensive in the long-run.

8.3.4.4 Study Results

In both cases our three core boundaries were able to identify points of cost that the domain experts agreed with. In fact, in both cases, the predictions differed from the participants prior perceptions of the risks, but after discussion all participants agreed that the predicted legs had higher costs than those originally assessed as most costly. This suggests that the modelling method provided to the participants an insight into the needs of their departments that they did not have before.

Finally, the four sessions of the workshop for the studies B and C lasted three hours in total. Those three hours included training of the HCPs in data journey modelling, creating data journey models, and identifying points of cost/risk, showing the quick property of our modelling method.

8.3.5 Stability Evaluation Results and Discussion

In this section we discuss the results of the three case studies evaluation in the Clinical Genomics area. In study A, the organisational boundary identified two costly journey legs. One of the legs was removed by hospital staff and the other was assessed by the domain expert of the study as a potential place for cost savings. Hence, the two predictions of the legs that crossed organisational boundaries are both true positives. The containers media boundary identified three journey legs as costly. All three predictions are true positives as they have been removed by the hospital staff in the new system after the re-engineering phase. The actors pay scale boundary identified one costly journey leg which has been removed in the new system. All three boundaries predicted a full coverage of TP predictions, hence were proved to be stable in study A.

Studies B and C, were not retrospectively evaluated since the data journey models represented systems currently in place in the respective Trusts, and we do not have true positive evaluations from them. However, upon comparing the findings of our method's boundaries with the points of cost/risk the clinical consultants gave before the boundary analysis, all participants agreed with the findings of the method.

Regarding time taken to create the models and identify points of cost/risk, study A took a total of two hours: One hour meeting with the domain expert, and another one hour of offline checking of the models produced. Studies B and C both took three hours to complete.

Taken together, the three separate studies show that our method's boundaries are stable in the three FT in the Clinical Genomics area. Also, results indicate that even this simple set of our core three boundaries can have significant predictive power, at very low cost, in the Clinical Genomics domain. Table 8.7 summarises the results we obtained from the three studies.

Case Study	Organisational boundary	Containers media boundary	Actor payscale boundary	Time
Study A	stable (two TP of two predictions)	stable (three TP of three predictions)	stable (one TP of one prediction)	Two hours
Study B	stable (one prediction)	stable (three predictions)	stable (three predictions)	Three hours
Study C	stable (four predictions)	stable (one prediction)	stable (four predictions)	Three hours
Stability	stable	stable	stable	

Table 8.7: Evaluation results of the stability of core boundaries.

8.4 Evaluation of Completeness of the Core Boundaries

Having assessed the stability of the three core boundaries to accurately identify places of cost and risk in the Genomics domain, the next question to examine is whether the core boundaries form a complete set. Can they identify all the significant costly places or are there any other places of cost/risk that they are not identifying? In this section, we describe a fourth case study in which we investigated the entire Clinical Genomics patient pathway, looking for significant costs/risks not captured by our three core boundaries.

8.4.1 Study Design

For this study, we worked with several groups of people: the domain expert of study A coming from a nearby Clinical Genomics department in Greater Manchester, the group of HSST staff from studies B and C, but also another group of NHS staff attending the Clinical Bioinformatics Genomics Masters course at the University of Manchester. The latter came from various NHS Foundation Trusts in the Greater Manchester and Liverpool area, and worked in a range of positions within the Genomics patient pathway, such as Genetic Counsellor, Genome Technician, and Clinical Geneticist.

The research question we pursued in this study is: how ‘complete’ is the set of three core boundaries in identifying points of costs/risks? Can the boundaries identify all the significant points of costs/risks that domain experts found?

We used several methods to capture information needed to evaluate the completeness of our method’s boundaries. We used semi-structured interviews with domain experts and took an online course of clinical genomics[3] to gain knowledge on the processes and data journeys happening in a typical department across the full pathway. To assess the completeness of the core three boundaries in identifying significant costs/risks, we followed the steps below:

1. Participated in the online course of Clinical Bioinformatics to gather knowledge needed to create a data journey model for the full patient pathway.

2. Conducted a semi-structured interview with the domain expert of Study A to refine and verify the data journey model produced in the previous step.

3. Observed the Introduction to Clinical Bioinformatics Masters course[4] to collect socio-technical challenges NHS staff face in their everyday work (throughout the patient pathway) that cause significant costs/risks to their departments.

4. Categorised above challenges based on the type of cost they can impose to the organisation.

5. Assessed each challenge to see if it was identified by the three core boundaries of our method to see if the boundaries capture all above challenges.

8.4.2 Study Findings

In this section we describe our findings of the steps we took to evaluate the completeness of our three core boundaries.

8.4.2.1 Data Journey Model

Participating in the online course we gathered knowledge on the movements of data needed by a specialist clinician to create an action plan for a patient with a rare disease. We then developed a comprehensive data journey model for the full clinical genomics pathway capturing those movements. Figure 8.8 shows the data journey model of the full Clinical Genomics patient pathway as validated by the domain expert of study A.

[3] The online course of "Clinical Bioinformatics: Unlocking Genomics in Healthcare" was provided by the University of Manchester in FutureLearn. The course can be accessed by: https://www.futurelearn.com/courses/bioinformatics.

[4] Introduction to Clinical Bioinformatics (MEDN68300) is a Masters course offered by the Division of Evolution and Genomic Sciences of the University of Manchester (https://www.bmh.manchester.ac.uk/study/cpd/courses/informatics-biostatistics-cpd/?pg=2&unit=MEDN68300&unitYear=1).

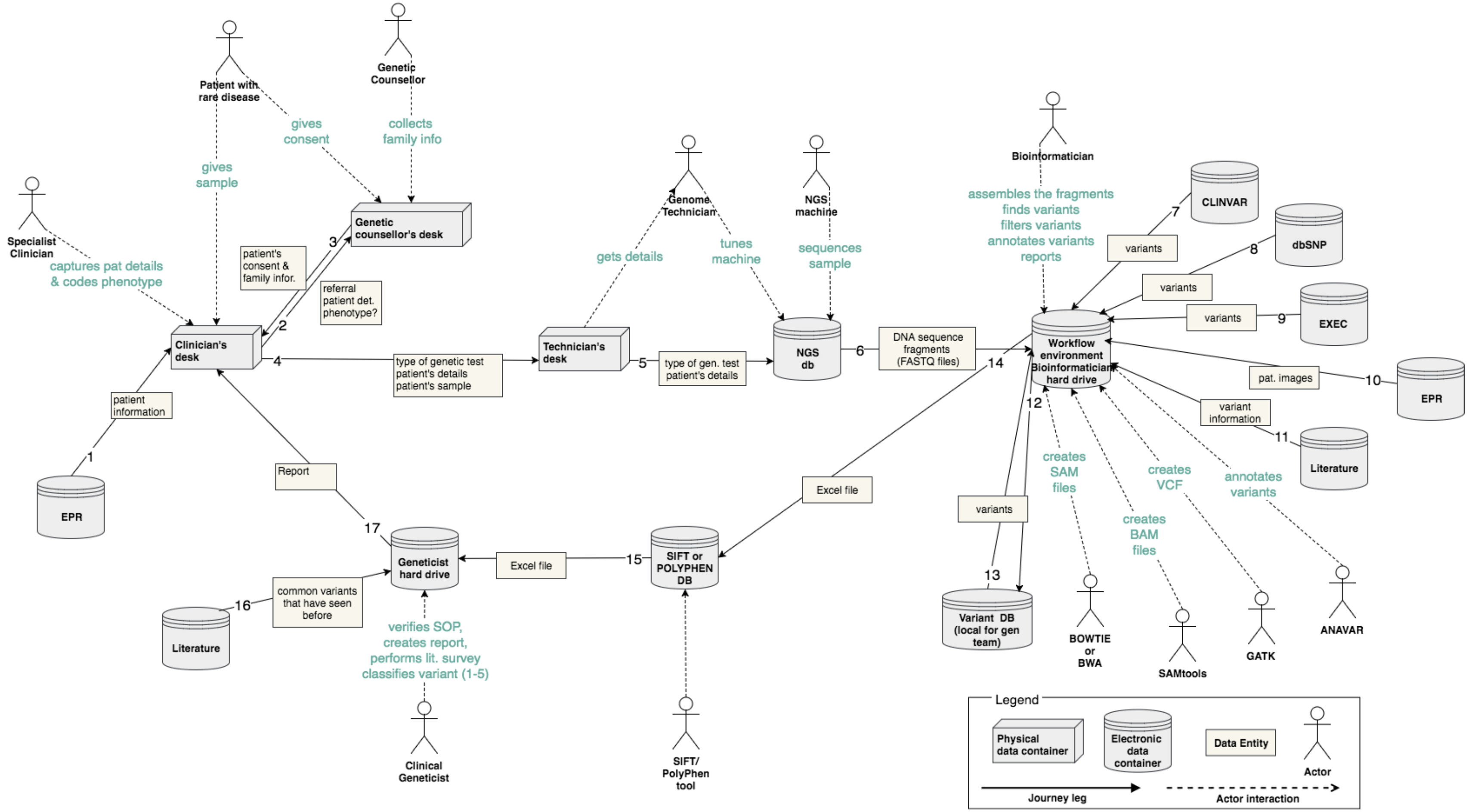


Figure 8.8: Data journey model of the full Clinical Genomics patient pathway.

8.4.2.2 Challenges in Patient Pathway

Next, we attended a session of the Introduction to Clinical Bioinformatics Masters course where we asked the participants to discuss challenges they face in their everyday work. We collected not just technical challenges stemming from the data and technologies used, but also social and organisational challenges that introduced some type of cost, risk, or extra effort to their everyday processes.

The challenges identified from this process are summarised in table 8.8. We categorised the challenges based on the type of obstacles they create. Then we examined each challenge in turn to determine whether costly/risky points would be picked up on the data journey model using our three core boundaries. To do so, we looked at the points on the data journey model we produced, where these challenges might materialise, to see if they coincided with the legs identified to be cost/risk hot spots by our three core boundaries. We give the boundary we deemed indicative of the cost described by each challenge in the second column of the table 8.8. Costly points not captured by any of the boundaries are noted with a question mark symbol.

From the table, we see that many of the challenges (62%) can be identified by our core boundaries. The people-oriented challenges were all identified by our actor boundary, as it identifies points where data move between members of staff with different kinds and degrees of specialist expertise. In our Clinical Genomics data journey model, this happens principally at the point of data handover from the clinical geneticists to the bioinformaticians, and *vice versa*. These groups of people have very different specialisms, and do not always share the same understanding of the data.

Similarly, the organisational boundary was able to indicate all the obstacles of sharing information with external sources. No change-of-media challenges were in evidence in the set of challenges we elicited from the domain experts in this case.

However, some challenges were not identified by our core boundaries. Specifically, those involving data volume and governance procedures were not captured by any of the core boundaries. Both these factors can limit the portability of data, and according to participants can introduce additional significant costs to the patient pathway. We explore each of them in the following sections.

8.4.2.3 Data Volume Challenge

The data volume is a major obstacle to the movement of data between key actors in the pathway. Sequencing a patient's genome can produce millions of data files, resulting

Challenges and potential costs/risks	Boundary
People - Oriented Challenges	
Data produced by teams of different backgrounds and expertise than the consumers are often misunderstood.	Actor
Data used by people other than those who captured them are often found to be inaccurate/incomplete. The external agency using them usually experiences a decreased quality of the data.	Actor
Lack of communication between stakeholders and the development team.	Actor
Staff are reluctant to share variants with other pipelines because of the governance frameworks in place.	?
Data Source - Oriented Challenges	
External data sources use different IDs. Some sources may have older versions of the same data entity. Also, different sources represent the same data in different ways.	Org.
There are governance issues whenever data are transferred between networks of different organisations (university and hospital network).	Org.
Information governance caused issues when integrating different parts of different pipelines (research and clinical).	?
Data Entity - Oriented Challenges	
Some data are 30–40 years old. Corrections to data over time cause duplicate versions of information (which are not explicitly marked as such at the source).	?
The sequencing machines produce very large volumes of data, causing storage and sharing problems.	?
Process - Oriented Challenges	
Different bioinformaticians use different workflows to process the data, leading to potentially different results.	Actor
Other Challenges	
Some data entities might not exist, like variants that havent been seen before. Hard to distinguish between non-existent data and absence from the source.	?
Information available in the literature is not always as accurate as claimed.	Org.
Clinical geneticists have only 10 minutes to comprehend findings produced by the bioinformaticians and take an, often, life-threatening decision.	Actor

Table 8.8: Challenges identified in the Clinical Genomics domain.

in large volumes of data (typically 10 – 100GB per patient). Given the complex nature of the pathway, with different actors typically working in different locations, and using their own dedicated software systems, large data volumes can be a real barrier to effective sharing and collaboration.

To validate the newly found obstacle of the volume of the data, we asked the two groups from case studies B and C whether they face challenges stemming from the movement of large volumes of data. Examples of volume-related challenges were experienced by participants in both studies:

- One of the participants reported the need to establish a new journey to move information from an old data repository to a newly created one. However, no connection had been established to migrate information between the two repositories (there were no governance barriers, since the movement was within the same hospital). Apart from communication problems between the stakeholders and the development team, the major problem faced by the participant's trust was the volume of data that needed to be shared. Since there is no direct network connection they currently have to copy each day's work (approximately 10GB of data) to the new repository, through external hard disks, every night.

- Another participant reported the need for exome data to move between two geographically distant sites (two cities in the UK). However, exome data sets are typically around 5GB in size, and attempts to transfer them cause the archive system used for transferring to crash.

- Moreover, in another data movement example in a university hospital, data needs to be transferred from the external university network to the internal NHS network. Both machines are in the same room, but are on different networks. The volume of data to be transferred is large, and the network is slow. The participant reported that sometimes they have to plug one machine directly into the other network to transfer the data.

8.4.2.4 Information Governance Challenge

The other type of boundary retrieved from the challenges relates to information governance. In domains where information is highly private and confidential such as clinical genomics, information sharing must be tightly regulated and controlled. Governance protocols must be established to ensure that patient's data are securely kept, and only used for agreed purposes. To complicate matters, governance protocols do not coincide with organisational boundaries. For example, within clinical genomics, there are two main protocols in use: a research oriented pipeline of processes and a clinical oriented pipeline. Each pipeline must follow the respective governance framework and guidelines. Audit information is captured along the pipeline based on the specific governance framework that applies. If governance protocols conflict, or are not fit for purpose, serious delays and additional costs can affect data sharing efforts.

8.4.3 Completeness Evaluation Results

This study identified two additional boundaries needed to root out the key obstacles to data sharing in a clinical genomics setting: data volume and information governance. Identifying the need for additional boundaries disproves our hypothesis that the set of three core boundaries of our method is complete. Our three core boundaries did not identify significant points of cost/risk produced by the large volume of data and information governance guidelines, suggesting that domain-specific requirements of organisations might drive the need for additional boundaries.

Although the two boundaries arose out of highly domain specific situations, they also seem generic enough to apply across a broad range of domains. In the era of big data, clinical geneticists are not unique in having to work with much larger data sets than their IT infrastructure is designed for. Nor are information governance protocols limited to the handling of genetic data. Hence, it seems reasonable to assume that these boundaries will have wider applicability than this one domain.

8.4.4 Volume and Governance Boundaries

The next step is to convert these high level concepts into actual “boundaries” that can be added to a data journey model. To be a useful boundary in our context, the information needed to decide which containers/actors are on which side of the boundary must be quick and low-cost to acquire. To apply the volume boundary on the model, we group together containers storing data sets of similar size. A simple and quick way to categorise the volume of the data entities is by using ‘tee shirt sizing’, in which size is described by broad categories (small, medium, large) [53]. Then, boundaries show where data move from containers handling large volume of data into containers set up to handle small volume, and *vice versa*. Cost and risk would arise when large volume of data must be processed by parts of the infrastructure set up to handle small volume of data. Similarly, to identify costs arising from the governance boundary, we group together containers and actors set to follow the research-oriented pipeline, and the clinical-oriented pipeline. Since containers and actors can work on both governance protocols, costs will arise when a journey leg moves data to a target container of a different protocol than those followed by the source.

8.5 Discussion and Conclusion

In this chapter we described our findings by applying the data journey modelling method to a variety of case studies in the healthcare domain. Table 8.9 combines and summarises the results of all five evaluation studies presented in this thesis.

Firstly, we retrospectively evaluated the accuracy of our method in identifying points of cost/risk in the radiology department of a Foundation Trust. The results of the evaluation showed that our method was able to cheaply and quickly identify most of the points of high cost/risk that the hospital staff had identified, along with several other possible directions that the staff did not identify for themselves, but agreed could be promising.

Secondly, we evaluated the stability of our method's three core boundaries in identifying points of cost/risk in the domain of Clinical Genomics, a more complex setting than the radiology study. Applying our method in three separate studies in three different Clinical Genomics Foundation Trusts, we found that the three core boundaries of our method are stable. Results indicate that even this simple set of our core boundaries can have significant predictive power, at very low cost in the Clinical Genomics setting.

Thirdly, we evaluated the completeness of our method's core three boundaries in identifying all significant points of cost/risk in the full Clinical Genomics patient pathway. Results of this part of the evaluation showed that our core boundaries did not identify all the significant points of cost/risk that health practitioners identified. However, we proposed the two additional boundaries of data volume and information governance to cover the gap. Findings suggest that domain-specific organisational requirements might drive the need for additional boundaries. For those cases we proposed a low-cost, up-front method that identifies additional boundaries in organisations with domain-specific requirements (described in section 7.6).

Our experience in applying our modelling method in five different studies showed

Success Criteria	Expected outcomes	Actual outcomes	Overall Result
Accuracy of predictions			
Radiology study accuracy	At least 50% of the predictions must be TP while FP predictions must be fewer than the TP predictions.	68% TP predictions and 11% FP predictions	✓
Cl. Genomics study A accuracy	At least 50% of the predictions must be TP while FP predictions must be fewer than the TP predictions.	100% TP predictions	✓
Cl. Genomics study B accuracy	Domain experts agree with the points of high cost/risk that our method identified.	Domain experts agreed	✓
Cl. Genomics study C accuracy	Domain experts agree with the points of high cost/risk that our method identified.	Domain experts agreed	✓
Lightweightness of method			
Radiology study domain experts time	One working day	We had three meetings of approximately one hour each with NHS domain expert #1, and five one-hour meetings with domain expert #2.	✓
Cl. Genomics study A domain expert time	One working day	We had one meeting of approximately one hour and another one hour of offline verification.	✓
Cl. Genomics study B domain experts time	One working day	Three hours (including training)	✓
Cl. Genomics study C domain experts time	One working day	Three hours (including training)	✓
Stability of core boundaries			
Organisational Boundary	stable across all three case studies	stable	✓
Containers Boundary	stable across all three case studies	stable	✓
Actors Boundary	stable across all three case studies	stable	✓
Completeness of core boundaries			
Cl. Genomics full patient pathway	Core boundaries should identify *all* significant cost/risk points.	Identified two more boundaries: Governance and Volume	X

Table 8.9: Overall evaluation results against success criteria.

that data journey models can typically be created in two–three hours (including training of domain experts in creating data journey models) depending on the complexity level of the study. Also, working with NHS practitioners of a variety of roles (i.e. IT developers, Integration Analysts, Clinical Consultant Managers, Genome Technicians, etc.), we found evidence that clinical staff can use our modelling technique as easily as those with an IT function.

Finally, although modelling the data journeys is only the first step in our technique, we found that it had value in itself, in helping health care practitioners to quickly gain an overview of how their organisation uses and shares data, when in their daily work they are more used to seeing only the detail of small parts of the journey which are more involved with.

Chapter 9

Discussion

> *"Simplicity – the art of maximising the amount of work not done – is essential."*
>
> *Agile Manifesto Principles*

9.1 Introduction

In this chapter we discuss the research questions we set to answer in the beginning of this thesis and provide the contributions we made to answer each question (section 9.2). We critically reflect on our contributions in terms of the overall body of work already conducted in the field of information systems and information sharing. Then, we discuss the significance of our results and the impact of our model and findings in the community (section 9.3). Also, we provide the strengths and weaknesses of our methodological approach as well as possible threats to validity coming from our evaluation (section 9.4). Finally, we provide a clear direction for future areas of investigation in terms of further evaluation areas, model refinement and model expansion (section 9.5).

9.2 Research Contributions

This thesis aims to aid the early stage decision making process of whether to proceed with a new IT development or not, by providing a modelling tool and method that helps managers of large, complex organisations identify places of potential high cost and risk that are likely to exist in a new development.

To begin with, we examined a set of 18 case studies from the health care domain written by staff coming from a variety of roles in the NHS across the UK. The case studies describe factors that contributed to the failure or success of recent IT developments in a variety of settings, such as cancer and diabetes care.

A common feature of the case studies where the new software was unsuccessful was the movement of data. The research hypothesis we pursued in this thesis is that cost and risk are likely to exist at the points where information is moved to a context other than the one it was originally designed for. We therefore set to develop a method that early and in a lightweight manner identifies the places in a planned new development where information crosses an organisational, human, or technical type of boundary that can introduce high cost and risk to the organisation.

The basis of our identification method is a novel lightweight modelling technique, called data journey model. Data journey models capture the movement of data through complex networks of people, systems and organisations. They abstract away from the fine-grained details of business processes happening in an organisation that require the movement of data. They conceptualise complex interactions of IT systems and their socio-technical environment into journeys of data.

In particular, we answered the research questions (RQ) we set in the introduction chapter (section 1.4), with the following research contributions (RC):

- *RQ 1: What factors contribute to the failure of a new IT development that can impose high cost and risk to the organisation?*
  RC 1: A set of IT failure factors in healthcare.
  Analysing the case studies from the NHS domain, we extracted a set of 32 factors that contributed to the failure of the IT systems recently introduced in a variety of settings in the NHS. A mixture of technical, human and organisational factors were found.

- *RQ 2: Are there any early warning signs of high cost and risk when developing a new IT system?*
  RC 2: A catalogue of data movement anti-patterns.
  We devised a catalogue of data movement anti-patterns providing early warning indicators of problematic movements of data that under certain criteria, can introduce high cost and risk in planned new developments.

- *RQ 3: How can we abstract away from the complexity of large organisational*

IT systems, to quickly and cheaply identify the aforementioned warning signs?
RC 3: Data journey model and notation: a modelling technique that captures data movements in large systems.
We propose the data journey model as an abstraction of large complex IT systems focusing on the journeys data make between collaborating systems of often different organisations. They capture the socio-technical elements of organisations that can contribute as early warning signs of high cost and risk.

- *RQ 4: How can we identify the warning signs of high cost/risk in the abstract model of the new system, early in the decision making process?*
  RC 4.1: Socio-technical boundaries: warning indicators that identify socio-technical points of cost and risk in new developments.
  Data movement boundaries are lightweight socio-technical indicators of problematic movements that can potentially impose high cost and risk to the organisation and cause the failure of the planned new IT system.

  RC 4.2: A set of five easy-to-apply boundaries.
  We found five boundary types, so far, that are easy and early to acquire and apply. The boundaries identify points of cost/risk imposed by a variety of factors be organisational, human, and technical level in nature.

  RC 4.3: A lightweight method that identifies socio-technical boundaries in data journey models.
  We devised a new method that applies socio-technical boundaries onto a data journey model and identifies the places in the journey of data with potential high cost and risk. The method is lightweight and low-cost. It can be completed in a couple of hours and relies only on easy-to-acquire and apply information.

- *RQ 5: Can the warning signs accurately and cheaply identify points of high cost and risk in a planned new system?*
  RC 5: Application in radiology department.
  The application of the model and method in a real-world case study in a nearby foundation trust. The retrospective evaluation showed that the model was able to quickly identify points of high cost and risk that the hospital staff found and replaced, but also identified other optimisation points that domain experts agreed could be promising.

- *RQ 6: Can the warning signs identify points of high cost and risk of new systems,*

across domains?
RC 6: Application in three Clinical Genomics settings.
The evaluation of the model and method in three case studies in the clinical genomics area showed the stability of our method in identifying most of the significant high cost and risk in this new and more complex domain.

- *RQ 7: Do the warning signs identify all the significant points of high cost and risk, or do we need additional signs when applied in other domains?*
  RC 7.1: Application in the full Clinical Genomics patient pathway.
  Evaluating the model and method in the complete patient pathway showed that although the method identified points of high cost and risk, some other significant sources were not identified. This suggests that other domain-specific organisational requirements might drive the need for additional boundaries.

  RC 7.2: A low-cost method to identify boundaries in other domains.
  We propose an up-front and low-cost approach to identify socio-technical boundaries in domains with specific requirements.

9.3 Significance and Impact of Contributions

From the literature review conducted and presented in chapter 3 (page 42) we found that although semantic interoperability and information sharing between information systems and organisations is vital for the effective and efficient patient care, they are still far from fully functional [47].

The invention and broad use of the HL7 and FiHR standards in the NHS and internationally, gave some structure on the technical side of data interoperability and sharing [19, 17]. Such standards provide structured and consistent exchange of electronic information between information systems across agencies. However, critical challenges in semantic interoperability still remain [131, 160].

Vernadat investigates interoperability in enterprise systems with focus on services and processes [177]. The author investigates interoperability at technical, semantic, and organisational levels and identifies the necessity of using standard ontology to reach semantic interoperability. He concludes that interoperable enterprise systems should place customers fully at the centre of an interoperability architecture. Chen *et al.* agrees and highlights that interoperability is at least as much about people, organisation, processes, and strategies as it is about technology [46].

In the data journey model, although data are at the centre of the model, people and organisational structures have equal prominence. The data journey model provides a holistic approach to modelling movements of data and capturing interoperability issues and barriers.

The results of the evaluation of our model and method in five case studies in the healthcare domain showed that even a simple, lightweight model can identify places of significant high cost and risk in new developments early in the decision making process without investing immense resources by domain experts.

Additionally, although we have some initial evidences that our model can identify substantial points of high cost and risk in existing systems in the healthcare domain, we can not know for sure that it can be used in potential applications. Further evaluation is needed to check whether our model and method can identify places in a planned new data journey that may cause high cost and risk to the organisation that can lead to the failure of the new functionality to realise the expected benefits.

Another domain where data journeys can potentially produce some value is during the requirements gathering phase of a planned new IT development. According to McConnell the most difficult part of requirements gathering is not to document what the end users want, but to figure out what they need and whether it can be successfully provided within the cost and schedule parameters available to the development team [130]. An important characteristic of an efficient requirements gathering phase, as highlighted by Young, is that best practice evolves the real requirements via a joint user and developer effort [185]. Data journeys involve the end users into the process of modelling potential journeys of data, as they do not require technical modelling expertise. They also provide an early identification of potential high cost and risk places to be further considered by the requirements elicitation experts.

Furthermore, our evaluation showed some preliminary results on the time and effort needed to create a data journey model. This new lightweight approach of data journey modelling contributes to the new trend of agile approaches and lightweight modelling, which argues that:

> *"Simplicity – the art of maximising the amount of work not done – is essential." (Agile Manifesto Principles* [1]*)*

Proposing the data journey model has some impact on the Software Engineering modelling community in the following four dimensions:

[1] http://agilemanifesto.org/principles.html

- The lightweight property of the model proposes a movement from heavy-weight, detailed modelling towards low-cost, and quick models facilitating an agile perspective.
- The low-cost property of the model makes it suitable for early stage decision making.
- The simplicity of the model's notation and method to identify points of cost enables non-technical personnel to participate in the modelling processes.
- The socio-technical analysis of cost and risk brings the element of people and organisational structures into the cost estimation processes.

Devising the data journey model and method, also proposes a move towards a more agile way of estimating points of cost/risk. Although existing approaches are very powerful and sometimes highly accurate, they are aimed at assisting managers and stakeholders throughout the decision making process, often after some development has been initiated. They rely on heavy-weight, detailed models and require substantial time and resources, that are often not existing.

Applications of our model and method in several cases showed that instances of data journey models can be developed in a couple of hours. Also, our method that identifies points of cost and risk in a data journey model relies only on low-cost information that are either already available to domain experts or are easy-to-acquire. The lightweight and low-cost properties of our modelling approach makes it suitable for early stage decision making as it is quick and lightweight to create and apply.

Additionally, evidences of developing data journey models with clinical consultants in the NHS indicate that our model may not require extensive technical background, or modelling expertise for domain experts to create new models. Also, data journey models created by technical personnel can be understood by non-experts facilitating the agile principle of:

> *"Business people and developers must work together (daily throughout the project)." (Agile Manifesto Principles)*

This property can empower domain experts to develop models of their organisation in-house without the need for external technical personnel, and to keep the costs of the decision making process low.

Finally, the data journey model combines both social and technical factors that can affect the movement of data, identifying socio-technical points of high cost and risk in a new development. Our model brings the elements of people and organisational structures into cost estimation problems, two parameters that are often neglected by current software cost estimation techniques.

9.4 Strengths and Limitations

One of the key strengths of this thesis is that we developed a method based on information we retrieved from real case studies written by domain experts working in the NHS. Literature and governmental bodies (i.e. NICE guidelines) provide typical scenarios of the processes happening in NHS FTs and the information systems and infrastructure that FTs have. However, the case studies provided rich data on what is actually happening in numerous NHS FTs across the UK. They describe existing systems and current underlying infrastructures in the NHS. They provide the processes that clinical and clerical personnel take to accomplish tasks in healthcare, and reasons why new systems have failed to realise the expected benefits. In such cases workaround processes have been devised by staff to save time and ensure an uninterrupted patient care. Such information is hard to find elsewhere. Also, the case studies cover a variety of application areas, and the authors come from a range of professions covering numerous points of patient care.

Another key strength is the application of the produced model and method into real world studies in the NHS, other than those used in analysis. By doing so, we evaluated the feasibility and accuracy of our model in a very complex setting in which is filled not only with technical barriers, but primarily organisational and social barriers. A setting that suffers from significant budget cuts and is in need of major re-engineering and optimisation exercises. Healthcare is a complex and adaptive eco-system meaning that its performance and behaviour changes over time. Jeffrey argues that healthcare lies in the most complex eco-systems that cannot be completely understood by simply knowing about the individual components [34]. Data journey models provide a tool to healthcare stakeholders to capture the bigger picture of the organisation they are working at. It helps them conceptualise the complexities of interactions between the teams of people and network of systems that have to work together in order to care for patients with often diverse needs and complicated interventions.

Limitations of this thesis are mainly derived from restrictions we had in time, and resources. We are a team of one researcher and two supervisors working for a limited period of time.

One of the inherent limitations of using case studies as the primary data for this project is that although they describe existing infrastructures and processes in a variety of FTs, they reflect the authors' personal perspectives of reality and points of view. Also, the contents of the case studies are confidential restricting us with publishing identifiable details.

Also, developing the model by following an agile approach that required extensive feedback from the users, might have ensured the continuous delivery of working product, but could also introduced the risk of developing a model that is well-defined for this particular set of end users. Our evaluation of the model in a different setting, involved end users of a different role (managerial clinical consultants) than those involved in designing and developing the model. However, further evaluation with other sets of stakeholders is needed to clarify this.

Finally, working with domain experts to evaluate our model and method, created some threats to validity points:

- There might be barriers in the organisation that people we worked with might not have been aware of.

- It could be that we have missed some stakeholders that have some key knowledge. How do we know that we have the right set of stakeholders involved in our studies? How to identify barriers that the people we worked with were not aware of?

- We might have incorporated in our evaluation an incorrect view of how the settings function. The way that the NHS system functions with a plethora of legacy systems; what people think is happening and what really is happening can be different things. So we might have got an incorrect view of how the system functions.

- Although we worked with clinicians to apply our modelling exercise to several studies, we have not investigated other potential stakeholders that might be more suitable in taking early-stage decisions for new developments. What are the right people to work with? How do we get the right knowledge and information out of them? These are questions that might not have a definite answer. Extra time and evaluation are needed in answering the above questions.

- A level of training is needed for non-experts to use the data journey model. The approach we followed in training potential stakeholders, involved face-to-face demonstration. However, this approach needs facilitators, people to demonstrate how the method works, for the modelling to work. We have not tested the approach of creating a new model without the need for a facilitator to tell the stakeholders what a data journey model is and how to use it.

- Healthcare practitioners and managerial staff are usually very busy, especially in the healthcare sector. Although we are providing them with a lightweight model, some level of training is needed in the beginning. However, we have not experimented with different levels of expertise and technical backgrounds to identify the depth of training needed for the different groups of people/stakeholders that might need to use the model.
- The fact that the domain experts found the model easy to apply and use, maybe is significantly based on the fact that we provided a training session introducing the model and how to use it.

9.5 Future Directions

Evaluating our model and method in 5 real-world case studies gave some evidence that our modelling approach can identify points of significant cost and risk in planned new developments. However, there are still some aspects that need further investigation. We describe each of them in the following sections.

9.5.1 Further Evaluation

All five case studies used to evaluate our model are in the healthcare domain. Further evaluation is needed in the following aspects:

- In other domains:
  Although the NHS is a large, and very complex organisation, it has some properties that distinguish it from other enterprises. It is primarily funded by the government where change is evidently slower and governance is at its highest. These two properties may have influence the formulation of the boundaries of our model. However, to fully test this, we have to further evaluate our model's boundaries on other organisations apart from the NHS, and ideally not governmental institutions.

- In a planned new development:
  Given the resources available, we could only evaluate our model in a retrospective way (that is, by applying the model on the system in place before major re-engineering work was done and comparing it with the current system in place), and in existing developments. To fully evaluate the accuracy of the model in identifying high cost and risk and its capability to assist the decision making process, we have to work with a team that currently has to take a go/no go decision.
  We first have to apply our model and identify places of cost/risk. The staff of the organisation have to make a decision without the help of our model. After the new development has been fully developed, we can compare the decision taken by the organisation's personnel and the findings of our model, to evaluate its accuracy and capability in assisting early stage decisions.
  Finally, we need an evaluation case study where the decision making team uses our model.

- In other size teams:
  We have already evaluated the scalability of the model in the bigger, more complex area of the Clinical Genomics patient pathway. However, we have, so far, worked with teams of maximum two stakeholders at a time. To test the scalability of the model in terms of the team size, we also have to apply it in a study of a bigger team size with a variety of stakeholder roles.

9.5.2 Model Refinement

Evaluating the completeness of our model we found that domain-specific requirements of an organisation might drive the need for further boundaries. Several research questions arise from this: how to distinguish whether a particular organisation has domain-specific requirements? How to identify the new set of domain-specific boundaries in this new domain?

The above two questions have been answered in chapter 5 (section 7.6) and an up-front, low-cost method for identifying new boundaries in other domains is proposed. However, the method is not yet evaluated. Evaluation is needed in a domain with requirements other than those we have already seen.

Additionally, another research question arises from this part of the evaluation: is there a refined set of boundaries that are applicable in a larger range of domains?

Answering this question will refine the set of boundaries of our method and further evaluation will be required.

Finally, reporting mechanisms to present the model's findings to stakeholders have not been fully evaluated. Refinement and evaluation is needed to ensure the best way to report findings is available to managers and stakeholders.

9.5.3 Model Expansion

Our model, so far, identifies a set of high cost/risk points, so what happens next? Once the model identifies a possible point of high cost or risk, does that mean that it should not be implemented, or be carefully monitored during implementation and running of the new development? In this section we suggest further actions to expand our modelling method to analyse, control, and monitor cost/risk.

Plans to expand our model to assist managers and stakeholders throughout the implementation, running, and maintaining phases of the new IT system lifecycle must cover the following points:

- **Cost/Risk Analysis and Prioritisation:**

Our model so far identifies places in the journey of data where cost and risk are possible to appear. In an attempt to control and monitor possible cost and risk of future developments, we first have to identify the types of cost and/or risk that can exist. Cost and risk need to be carefully, but also low-cost, analysed and then prioritised from the most to least dangerous.

Existing approaches can be found in the risk analysis literature. Risk analysis assesses the loss probability and loss magnitude for each identified risk point [27]. Typical techniques include: performance models, cost models, network analysis, statistical decision analysis, and quality-factor analysis (e.g. reliability, availability, and security). A couple of suggestions for the lightweight quantification of cost and risk, not to jeopardise the lightweight property of our model, are the scale-of-10 estimation and t-shirt sizing techniques, proposed by Boehm and Cohn respectively [27, 56].

Risk prioritisation produces a ranked ordering of the risk items identified and analysed. Typical techniques include risk-exposure analysis, risk-reduction leverage analysis (particularly involving cost-benefit analysis), and Delphi or group-consensus techniques [27, 93, 122]. However, a low-cost prioritisation of the identified cost and risk can be used utilising a traffic light system (Red – Amber – Green). A red warning sign can indicate the points with the highest cost and risk that need corrective actions, while amber can indicate points of cost/risk that can be implemented, but need monitoring, and finally green indicates points of minimal cost/risk that can be accepted (but mitigated).

- **Cost/Risk Control and Monitoring:**

Having identified potential places of red – amber – green types of cost/risk, we can then monitor those places to control the cost and risk that appear during the implementation, running and maintenance phases of the new IT system.

Performance measurement plans and key performance indicators can be used to create a scheme in which organisations compare actual with expected, but also optimal performance [102, 127, 138, 145]. Having established such a scheme we can monitor the types of actual cost/risk appearing in each identified place. This will give vital insight on refining our model, but also in controlling any cost/risk that may arise by providing early warnings whenever a monitored cost/risk increases.

Cost and risk resolution approaches can then be used to produce a situation in which the risk items are eliminated or otherwise resolved (for example, risk avoidance via relaxation of requirements). Typical techniques include prototypes, simulations, benchmarks, mission analyses, key-personnel agreements, design-to-cost approaches, and incremental development [27].

9.6 Conclusion

In this chapter we discussed how we answered the research questions set in the beginning of this thesis, critically reflected on the research work conducted, and compared our contributions with the context of the overall body of work already conducted in the field. Finally, we discussed strengths and limitations of the methodological approach as well as threats to validity of our evaluation and provided a clear direction for future areas of investigation.

Chapter 10

Conclusion

Often new organisational requirements, or technological advances require new IT systems to be developed to either expand or complement already existing functionality. Managers and stakeholders of the organisations have to make quick up-front decisions on the feasibility of such systems. Making such decisions is hard. Cost and risk of developing new systems are regularly underestimated, causing the failure of the system and often detrimental consequences to the organisation and people affected. Specifically in the health care domain, a significant amount of money was wasted in implementing IT developments that partly or completely failed to realise the expected benefits. This causes vital resources to be wasted that could otherwise be invested in saving patients' lives.

The analysis of 18 case studies from the NHS domain showed that although movement of data is vital, it can be affected by several socio-technical factors that can impose high costs on the development of new applications. Given that these costs are often underestimated, we need a way to quickly identify and predict barriers of data movement, ideally before initiating any development.

In this thesis we proposed a new low-cost method that uses easy-to-acquire socio-technical information to identify places of high cost and risk when an existing dataset moves to a new development. The method is based on a lightweight model, called data journey model, which conceptualises the journey of a set of data from their original location in an information infrastructure to the new development, through a complex network of systems, people, and organisations.

We evaluated the effectiveness of our method and the accuracy of our findings by applying our method and model in five real world studies in the NHS domain. We worked with clinicians from several Foundation Trusts across the UK to retrospectively model the movements of data in the settings of Radiology, and Clinical Genomics. We modelled the journey of data before and after a major information infrastructure redesign, and compared the two models to find if our identified places of high cost and risk in the old models have been overcome in the new.

Finally, the results of the evaluation showed that the data journey modelling method can accurately identify points of significant cost and risk in new developments. It proved to be a cost-effective method that can be completed in less than a couple of hours including training, making it suitable to be used in early stage decision making process. The application of the method in several domains showed the stability of the method's boundaries across domains. Last but not least, the simplicity of our modelling technique can empower domain experts with no particular modelling expertise to quickly identify opportunities for cost savings in new developments, as well as existing ones.

Appendix A

NHS Radiology Case Study

A.1 NHS Case Study Business Processes

This section gives the business processes of a typical GP and hospital in the NHS when a GP patient might have a fracture and needs an X-ray scan to be taken at the local radiology department.

1. A GP fills in a request card to initiate the process of requesting a radiograph. The request card describes the type of X-ray needed and the patient's details. The request card is sent by post to the radiology department at the clerical reception area.

2. At the radiology department, a member of clerical staff receives the request card and creates an appointment for the patient in the radiology system. A letter containing the time and date of the appointment is created and sent to the patient through the post.

3. Before the patient arrives at radiology, the packet with the patient's previous X-rays and reports is transferred from the Filmstore area to the radiology clinical area by clerical staff using a trolley. If the patient has no previous X-ray scans, a member of clerical staff will create a packet at reception and takes it to the clinical area. A label with the patient's identification details will be attached to the packet.

4. On the day of the appointment the patient arrives at reception. Clerical staff will guide him to the clinical area. At the clinical area, a new X-ray is produced by a

radiographer. The new X-ray is placed inside the patient's packet. The packet is then put into a pigeon hole by the clinical staff to be transferred to reception.

5. The packet is then transferred to the reception area by a member of clinical staff. The packet is then placed in a pigeon hole by clerical staff, from where a radiologist collects it. The radiologist takes the packet to his/her office, examines the X-ray scan and dictates a report onto a cassette.

6. The radiologist gives the cassette and the packet to the secretary who transcribes the report into the radiology computer system. The report is printed and given to the radiologist to verify. If changes have to be made, the secretary amends the report in the system and prints it for verification.

7. A print out of the final report is placed in the packet by the secretary. The packet is then placed on a trolley to be sent back to the Filmstore by clerical staff. The secretary prints another copy of the report and puts it into an envelope to be sent to the GP. The porter collects the envelopes and transfers them to the porters area into a pigeon hole based on the GP address. The courier collects the envelopes from the pigeon hole and transfers them to the GP reception. The GP secretary gives the reports and the patient's folder to the GP. Sometimes, the GP secretary scans the printed report and inputs it into the GP system. The scanned report is linked with the patient's record. The GP accesses the scanned report.

A.2 Creating the Data Journey Model

This section describes the process of designing the data journey model of the NHS case study. The model represents the journey of data needed by a GP to decide on an action plan when a patient may have a fracture, based on the business processes given in A.1. Below steps follow the bottom-up approach described in chapter section 6.4.

Step 1: Identify data entities of interest

The first step after understanding the process and identifying the scope of the movement is to identify the data entities of interest. These are the data we want to move to the new development and their transformations. They can usually be derived from the scope of our journey. The data entities from the NHS case study are the data that a GP needs to decide on an action plan. These are the patient's identification details and the radiograph findings (referred to as report).

However, in order to create a report, a radiograph image (X-ray) is needed. But, what initiates the process of taking an X-ray? Data, once created, can be transformed, annotated, and updated before they are used by a consumer. In order to track the flow of moving data we need to trace previous forms of that data to find their origins. For example, a GP has to request an X-ray to be taken by filling in a request card and sending it to the radiology department of the foundation trust. The request card will then cause an appointment to be made for the patient to attend the radiology, and the X-ray is taken.

Step 2: Identify the data containers in which data entities are stored.
Once we have identified the data of interest, we have to find the containers from which those data originate, are moved into and are finally made use of. Containers are stable, non-transferable places in which data can be stored. Containers can be electronic databases or physical locations, such as desks, filing cabinets or even pigeon holes. Data containers we identified from the NHS case study:

- GP's desk
- GP reception desk
- Radiology clerical reception desk
- Radiology information system's database
- Patient letter box
- Filmstore storing area
- Radiology clinical desk
- Radiology clerical reception pigeon hole
- Radiologist's desk
- Secretary's desk
- Porter area pigeon holes
- GP system database

Steps 3 and 4: Identify the routes and the media by which data are transferred. After we find the containers of the journey, we identify the routes and the medium by which data are transferred from a source container to a target container. The medium is the means by which data are moved and can be in electronic or physical form, such as a sheet of paper, a request card, a folder, a label, etc. The routes, medium and the data entities moved we identified in the case study are:

- The request card is transferred from the GP's desk to the reception desk, and lastly to the radiology clerical reception desk. It is in physical form and contains the following data entities: the patient's NHS ID[1], patient demographics, type of X-ray request.

- The radiology department patient packet is moved from the filmstore to the clinical desk, to the clerical reception desk, pigeon hole, radiologist's desk, secretary's desk and finally back to the filmstore area. It has a physical form and contains the data entities of: the unit ID, patient demographics, previous X-ray images, previous reports, new X-ray, new report.

- The cassette is transferred from the radiologist's desk to the secretary's desk and contains the data entities: patient identification details (various specified depending on the preferences and habits of the radiologists) such as, NHS or unit ID, name, surname, date of birth, the report. (A single cassette usually contains multiple dictations reporting on numerous patients.)

- The details captured in the cassette are moved into the radiology's system database which contains: the patient's demographic information, address, telephone, GP details, next of kin, etc.

- The report is moved from the radiology system database to the secretary's desk and it contains: the NHS and unit ID, patient name, surname, date of birth, radiograph findings.

- The envelope moves from the secretary's desk to the GP reception, and then to GP system database. It contains the patient's details and the report.

- The report is moved from the GP reception desk to the GP system database.

[1] Each patient has a unique NHS ID. The NHS ID is given to the patients when they are born or become eligible for NHS care. When a patient attends hospital, they get a hospital ID, called unit ID. The Unit ID is unique per patient per hospital. Hospitals use the unit patient ID, but GPs usually use the NHS ID.

- The GP patient folder is moved from the GP filing cabinet to the reception desk, and finally to the GP's desk. It contains all the details of the patient since first registered with the GP and the report.

Step 5: Identify the actors interacting with containers.
The fourth step in constructing a data journey model is to identify the actors who interact with the previously identified containers to create, use or transform data entities stored in them. Actors can be people or systems and interact with the data stored in a container. They do not interact with data while they are moving between containers. The actors we identified in the case study are the following:

- GP
- GP secretary
- Patient
- Radiology secretary
- Radiology clerical staff
- Radiology clinical staff
- Radiologist
- Radiographer

Step 6: Draw the data journey diagram
The final step is to diagrammatically represent the data journey model using the notation given in figure 6.1 on page 90. The result of the representation is illustrated in figure 8.1 on page 121.

A.3 Other Data Journey Boundary Diagrams

Once we have created the data journey model of the data entities of interest, we can overlay on it the boundaries; socio-technical information to help us identify places of the journey of high costs and risks.

Figure A.1 gives the organisational boundaries, and figure A.2 the media boundary. Figure A.3 shows the actors interacting with the containers to create, consume or transform data in order to produce some value. All figures note journey legs that crossed a boundary with a red warning sign indicating a likely place of high cost.

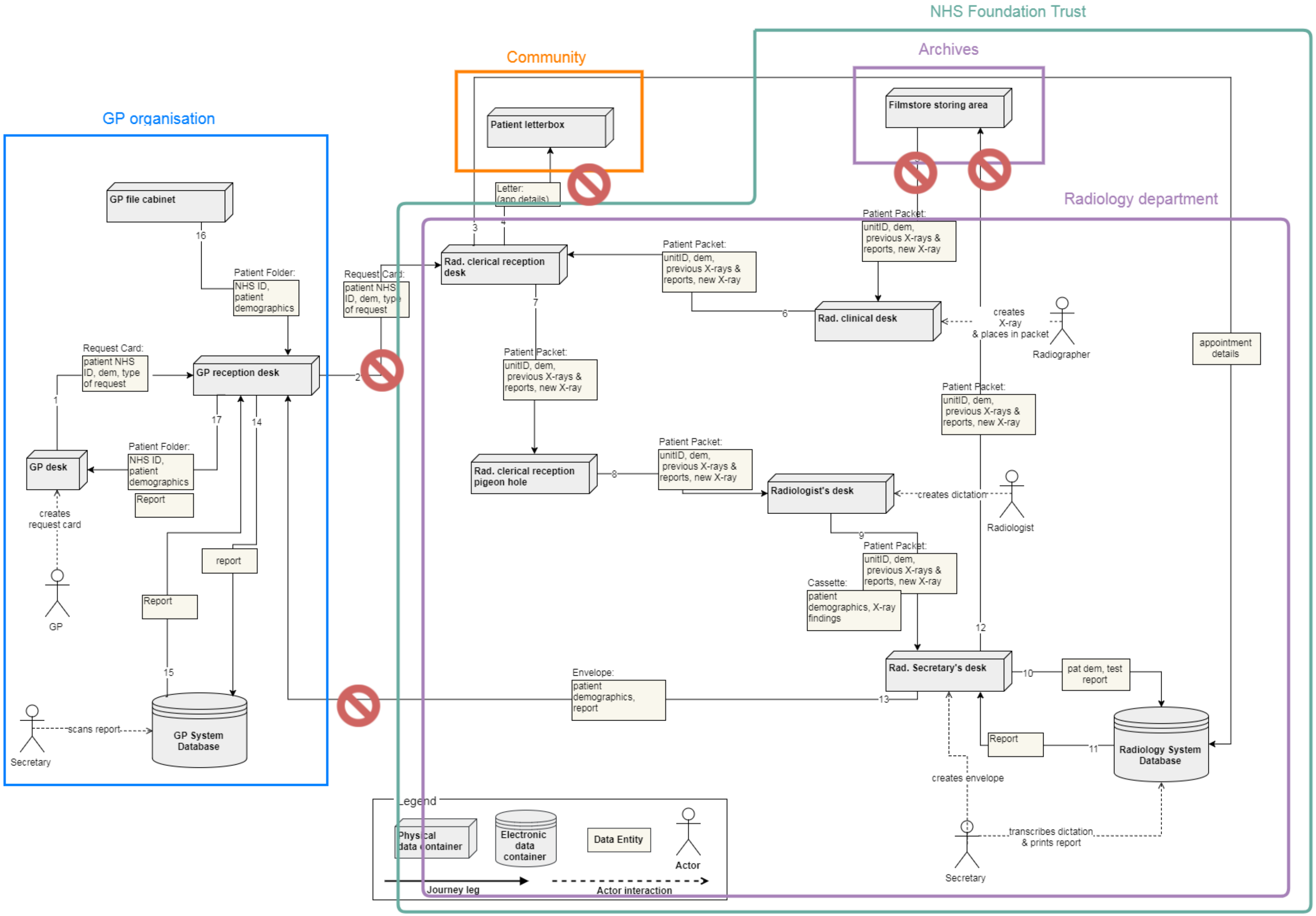


Figure A.1: Data journey model of the old system with overlaid organisational boundaries.

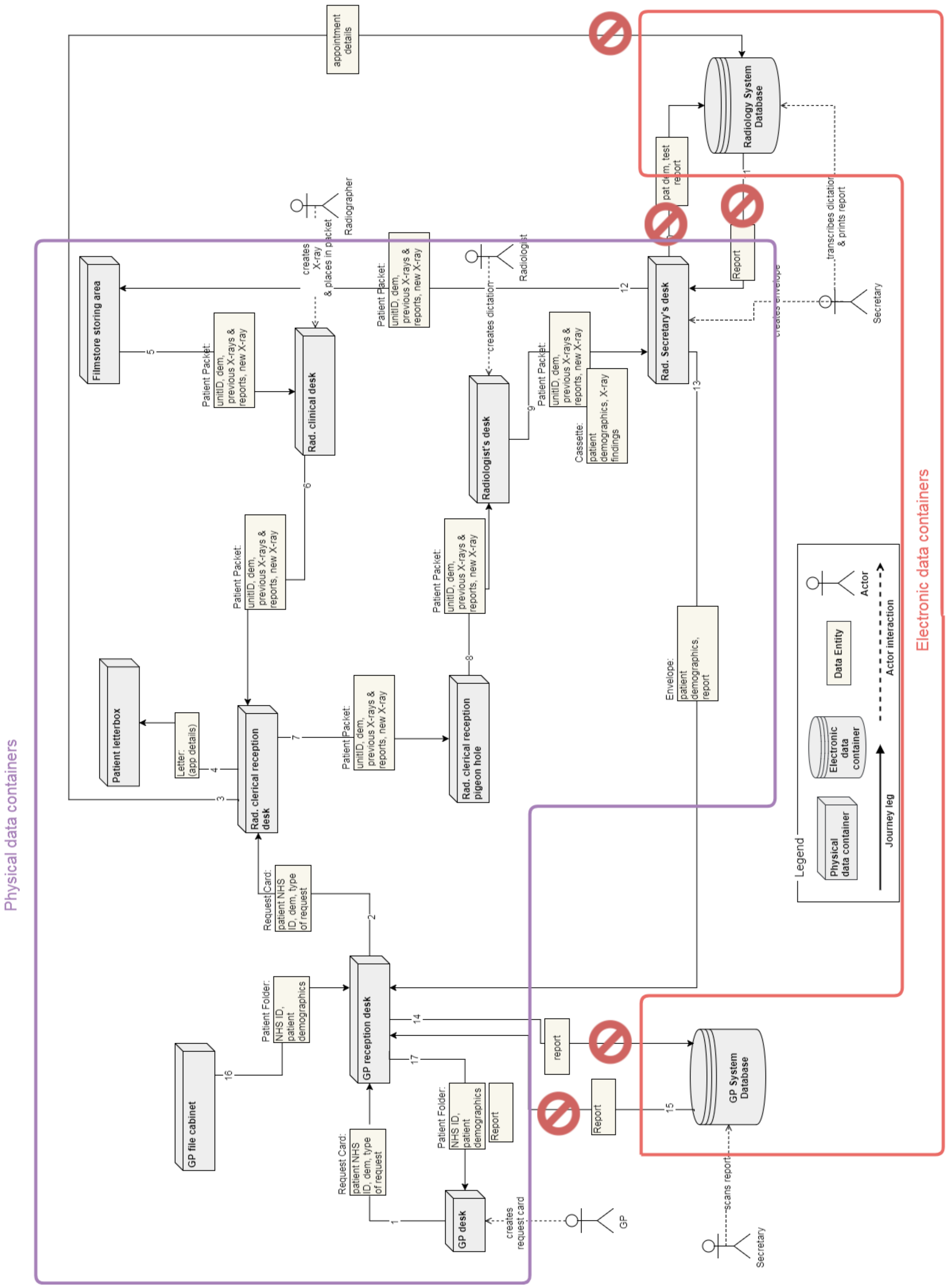

Physical data containers
Electronic data containers
appointment details
Radiology System Database
1
pat dem, test report
Radiographer
creates X-ray & places in packet
Filmstore storing area
5
Patient Packet: unitID, dem, previous X-rays & reports, new X-ray
Rad. clinical desk
Radiologist
creates dictation
Patient Packet: unitID, dem, previous X-rays & reports, new X-ray
12
Rad. Secretary's desk
Report
creates envelope
Secretary
transcribes dictation & prints report
9
Patient Packet: unitID, dem, previous X-rays & reports, new X-ray
Radiologist's desk
Cassette: patient demographics, X-ray findings
13
6
Patient Packet: unitID, dem, previous X-rays & reports, new X-ray
Patient Packet: unitID, dem, previous X-rays & reports, new X-ray
Patient letterbox
Letter: (app details)
4
3
Rad. clerical reception desk
7
Patient Packet: unitID, dem, previous X-rays & reports, new X-ray
8
Rad. clerical reception pigeon hole
Envelope: patient demographics, report
Request Card: patient NHS ID, dem, type of request
2
Patient Folder: NHS ID, patient demographics
GP reception desk
16
GP file cabinet
14
17
report
GP System Database
15
Report
Patient Folder: NHS ID, patient demographics
Report
Request Card: patient NHS ID, dem, type of request
1
GP desk
creates request card
GP
scans report
Secretary
Legend
Physical data container
Electronic data container
Data Entity
Actor
Journey leg
Actor interaction

Figure A.2: Data journey model of the old system with change of media boundary.

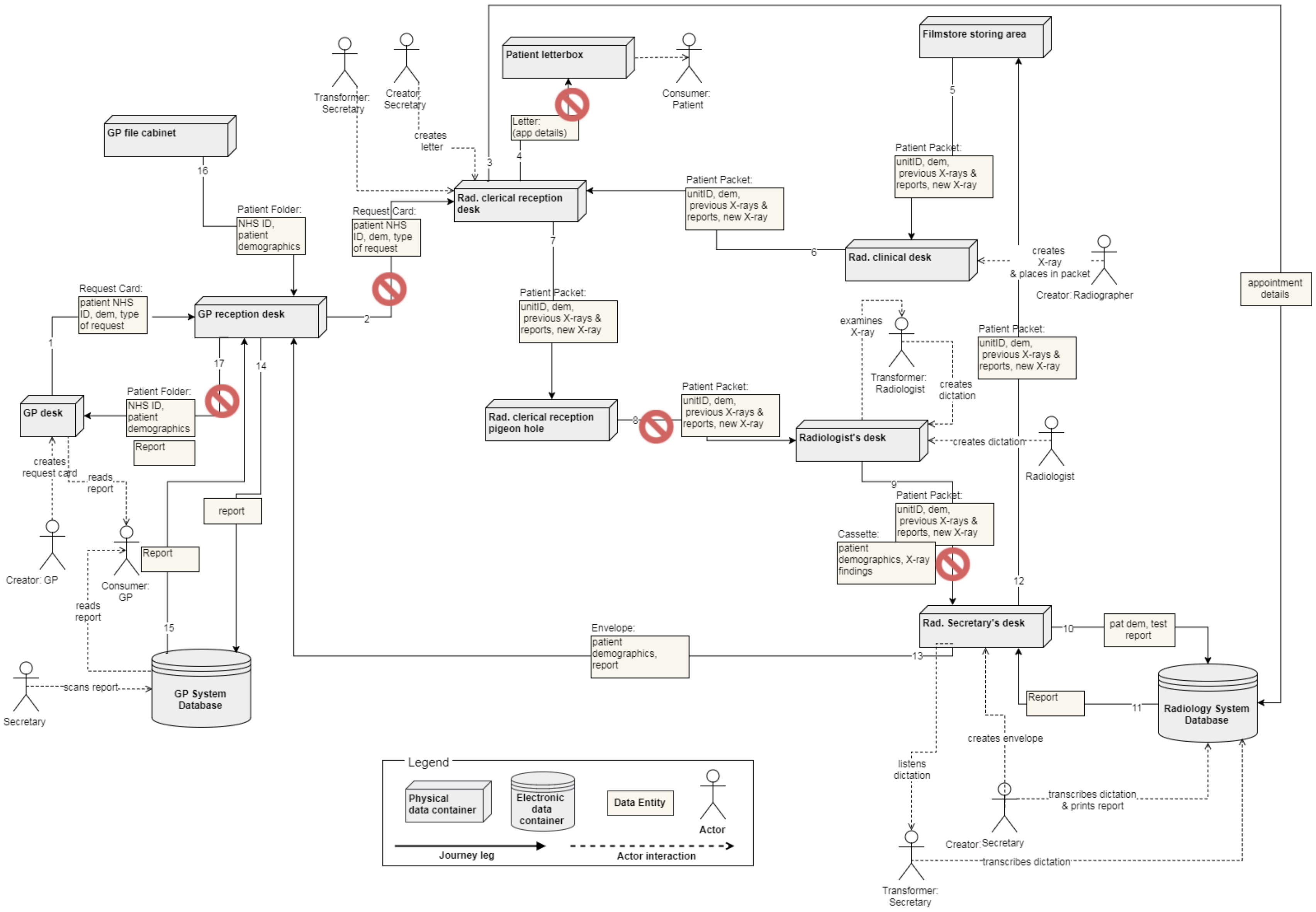


Figure A.3: Data journey model of the old system with actors' role boundary.

Appendix B

NHS Clinical Genomics Case Studies

B.1 Clinical Genomics Patient Pathway Business Processes

The patient pathway can be initiated by different protocols, based on the patients needs, or research aims. There are two main pipelines; a research oriented pipeline and a clinical oriented pipeline. Each pipeline must follow the respected governance framework and guidelines. Audit information is captured along the pipeline based on the respected governance framework. In the studies covered in this thesis we discuss the latter. Figure B.1 shows a typical example of the business processes explained below, as used for the data journey models presented in the NHS studies in section 8.3.

A patient presents an anomaly and the clinician suspects they have a particular rare recessive genetic disease. The patient is referred for genome sequencing to a genetic counsellor. The genetic counsellor discusses with the patient the possibility of having a rare disease and the process of sequencing the patients DNA to identify the variant responsible for the disease. The counsellor must make sure that the patient understands the risks of any procedures and gets informed consent. The counsellor gathers patients inheritance information and the patients phenotype. The patient will provide a blood sample, which will be sent to the genome technician.

The genome technician inputs the patient sample into a Next Generation Sequencing (NGS) machine that extracts the patients DNA sequence into fragments. A set of short overlapping fragments from the genome is produced and given to the bioinformatician. Data come from the sequencing machine in the form of FASTQ files and each file contains information about the sequence of the short fragment that was sequenced and some quality data (error probability). A big sequencing run can generate millions of FASTQ files and require significant disk space to store.

The first step of the bioinformaticians role is to assemble the set of fragments into a complete genome. A very challenging and computationally intense process, primarily because of the big amount and size of data given (3 billion base pairs, around 2GB of data), and the inaccuracy of the sequencing process (errors are common, and hence expected). This is stored in the form of a big sequence alignment - a SAM file (Sequence Alignment/Map format) which contains information about the alignment and the quality of the alignment. These files can get very large 10 to 100GB. Numerous tools are used to assemble the genome, such as Bowtie (aligns short DNA reads to the human genome), BWA (Burrows-Wheeler Aligner, a tool used for mapping low-divergent sequences against the human genome), Samtools, (converts BAM files to SAM).

There are numerous techniques to sequence human genomes. Each can produce a different set of fragments called gene panel, exome, or whole genome.

The second step of the bioinformaticians role is to find the variants; the bioinformatician is looking for differences between the patients genome and a typical prototype human reference. Several tools help the bioinformatician throughout this process. The variants are stored in Variant Call Format (VCF) files. VCF files are compact representations of the differences between the patient genome and the reference genome.

The third step is to triage (prioritise) the variants. The bioinformatician looks for changes in the genotype (the DNA sequence) that could be responsible for the phenotype (set of observable traits) in the patient to filter and prioritise the variants with the help of specialised tools and scripts. The fourth step is to add metadata to the filtered list of variants of the patient by searching disease/condition specific databases to see if the variant changes have been reported by other clinical scientists.

The final step of the bioinformaticians role is to report to the clinical geneticist. The bioinformatician has taken an entire genomes worth of data and filtered it down to a small excel sheets worth of candidate variants for each of which is annotated with supporting evidence. The excel sheets are given to the clinical geneticist to decide whether anyone of the candidate variants can provide an explanation of the patients phenotype. Further information on the patient pathway can be found at:
https://www.genomicsengland.co.uk.

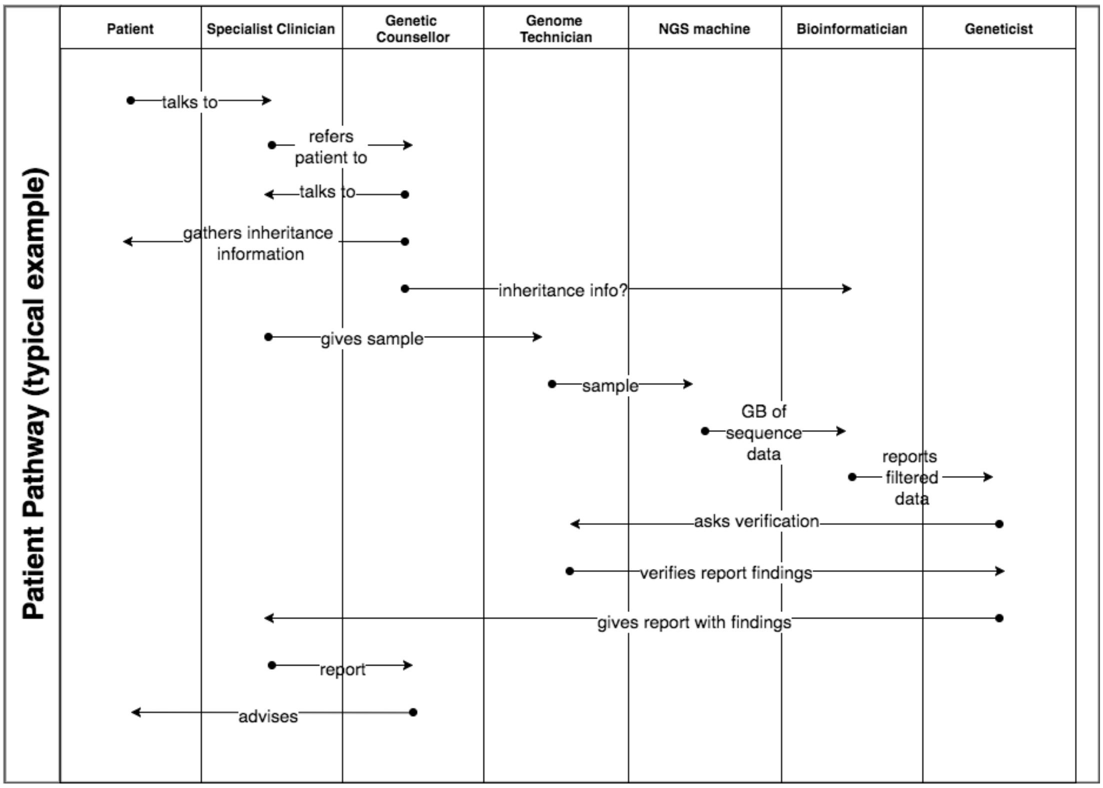


Figure B.1: A typical example of the processes happening in the Clinical Genomics patient pathway we produced.

B.2 Data Journey Models of Studies B and C

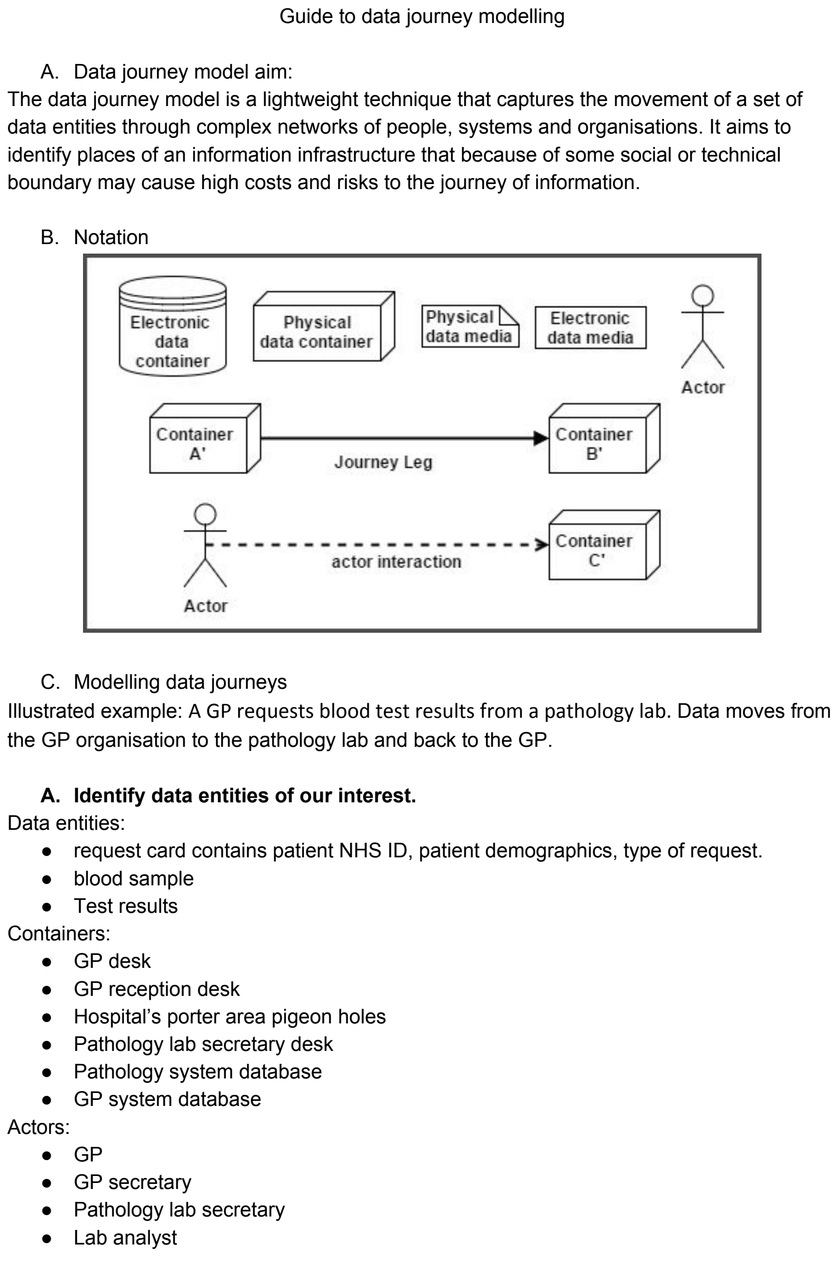
This section presents pictures of the data journey models created as part of the NHS evaluation studies described in section 8.3.

The data journey models given in figures B.5, and B.7, were created by NHS Clinical Consultant Managers coming from various Clinical Genomics departments across the UK. The NHS staff had no previous experience in designing data journey models. They were introduced to the modelling approach on the day, prior to creating the models.

The two groups of Clinical Consultant Managers were first introduced to the data journey modelling technique and were given a working example as a guide. Figures B.2 – B.4 show the pathology lab working example given to them prior to the data journey modelling exercise.

Then the Clinical Consultant Managers created the data journey models of their respected trusts and then overlaid the organisational, containers media, actors role boundaries on the model. The overlaid boundaries are shown in figures B.6 and B.8.

We used a physical approach to model the journeys using paper, coloured markers, post-it notes and transparency films (to overlaid the boundaries). This agile approach abstracted away from the details of learning and using an online drawing tool, while it provided an informal–safe environment were mistakes are encouraged as no one can perfectly know beforehand all the data movements happening in their departments.



Guide to data journey modelling

A. Data journey model aim:

The data journey model is a lightweight technique that captures the movement of a set of data entities through complex networks of people, systems and organisations. It aims to identify places of an information infrastructure that because of some social or technical boundary may cause high costs and risks to the journey of information.

B. Notation

C. Modelling data journeys

Illustrated example: A GP requests blood test results from a pathology lab. Data moves from the GP organisation to the pathology lab and back to the GP.

A. Identify data entities of our interest.

Data entities:

- request card contains patient NHS ID, patient demographics, type of request.
- blood sample
- Test results

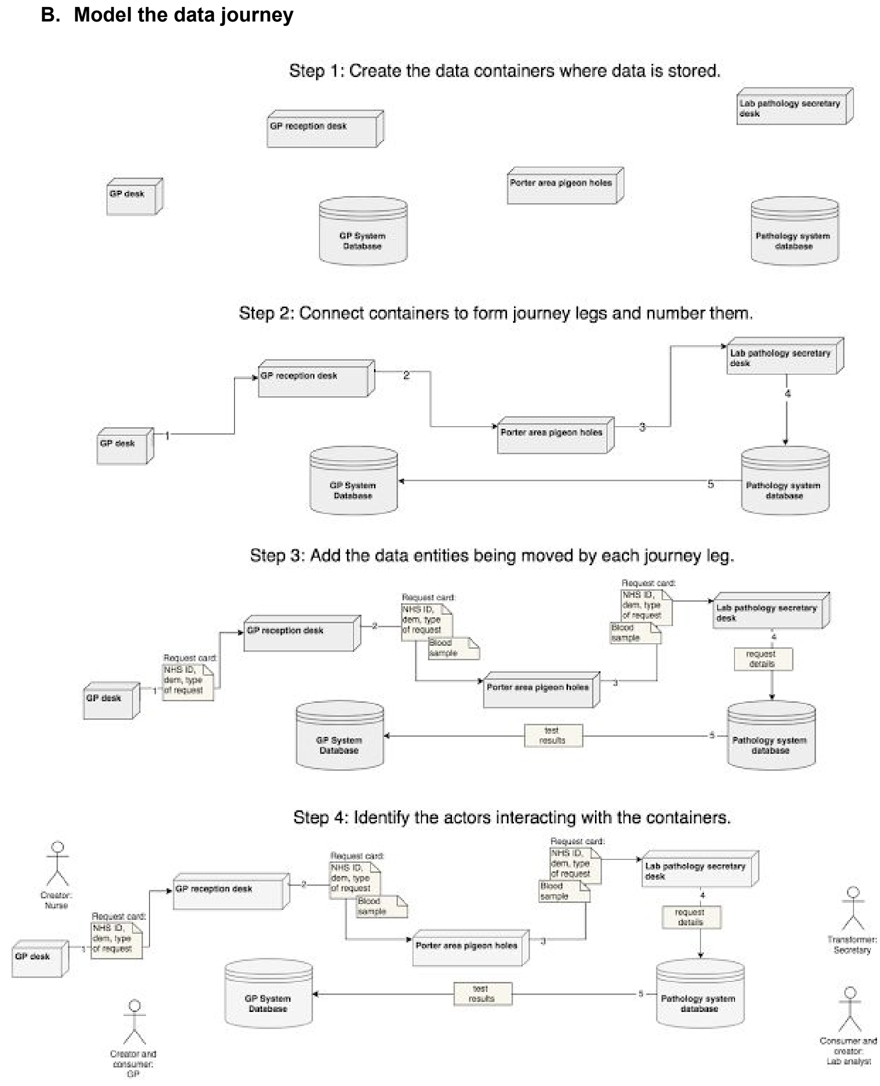
Containers:

- GP desk
- GP reception desk
- Hospital’s porter area pigeon holes
- Pathology lab secretary desk
- Pathology system database
- GP system database

Actors:

- GP
- GP secretary
- Pathology lab secretary
- Lab analyst

Figure B.2: Guide to data journey modelling given to Clinical Consultants as part of the training session (page 1).



B. Model the data journey

Step 1: Create the data containers where data is stored.

Step 2: Connect containers to form journey legs and number them.

Step 3: Add the data entities being moved by each journey leg.

Step 4: Identify the actors interacting with the containers.

Figure B.3: Guide to data journey modelling given to Clinical Consultants as part of the training session (page 2).

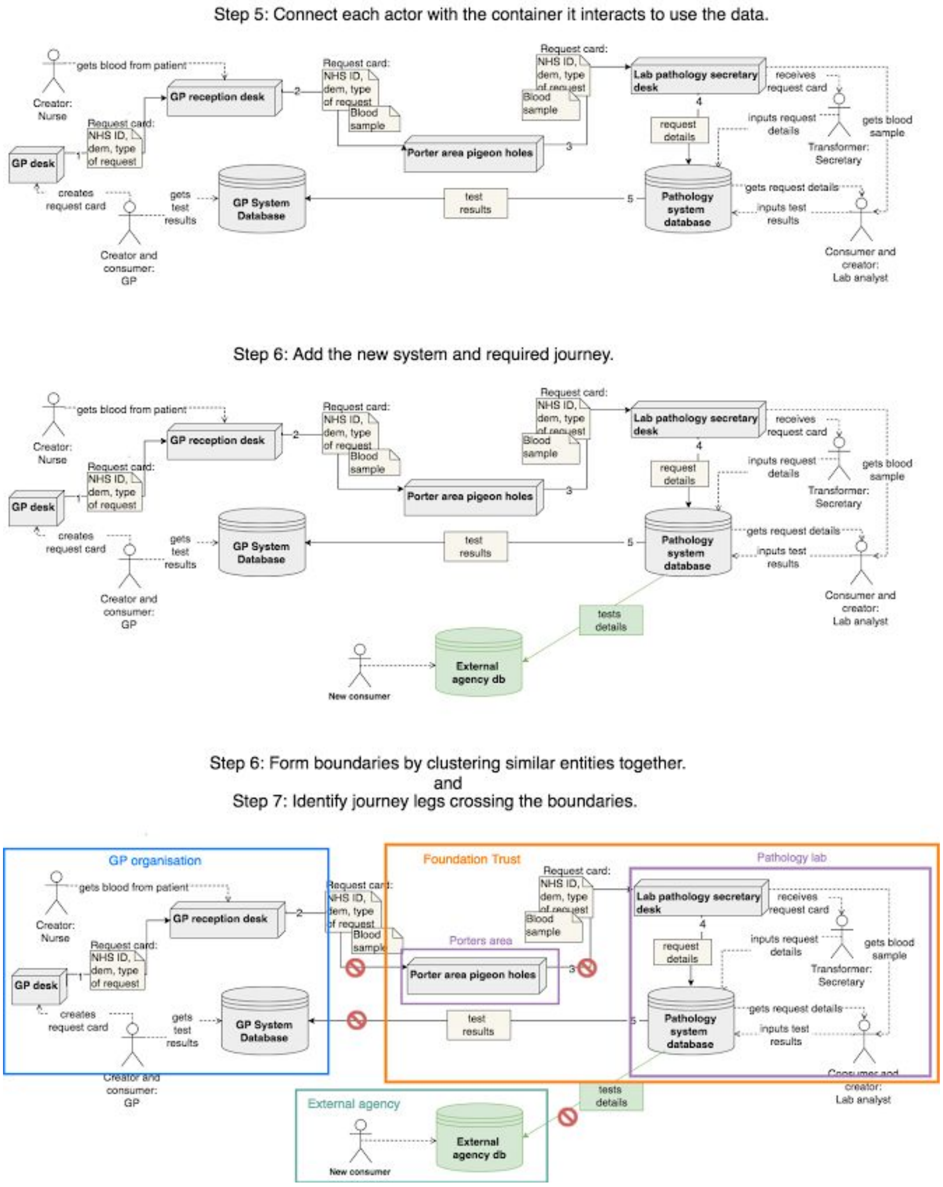


Figure B.4: Guide to data journey modelling given to Clinical Consultants as part of the training session (page 3).

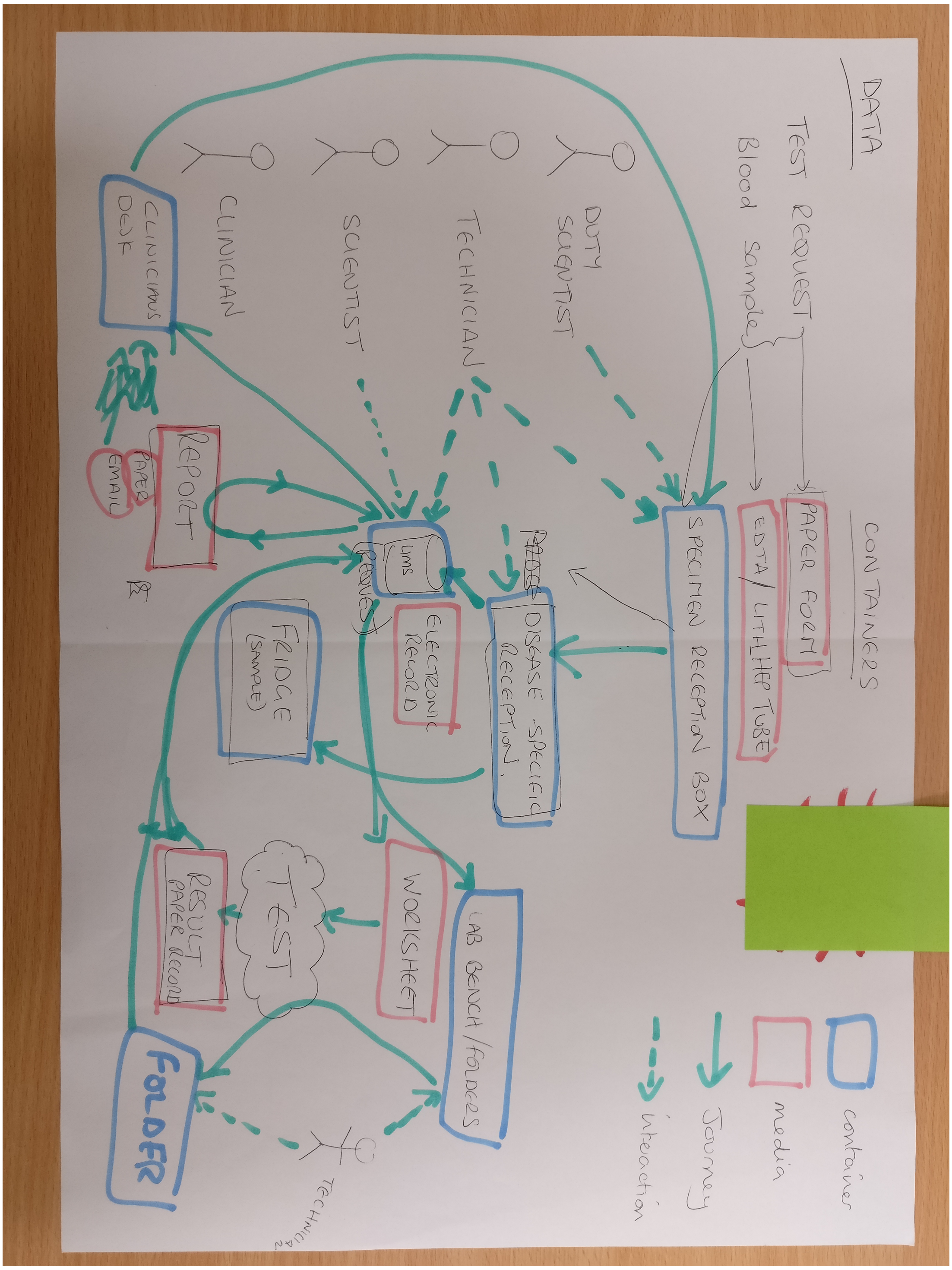


Figure B.5: Data journey model of Clinical Genomics Study B as designed by Clinical Consultant Managers in a nearby foundation trust.

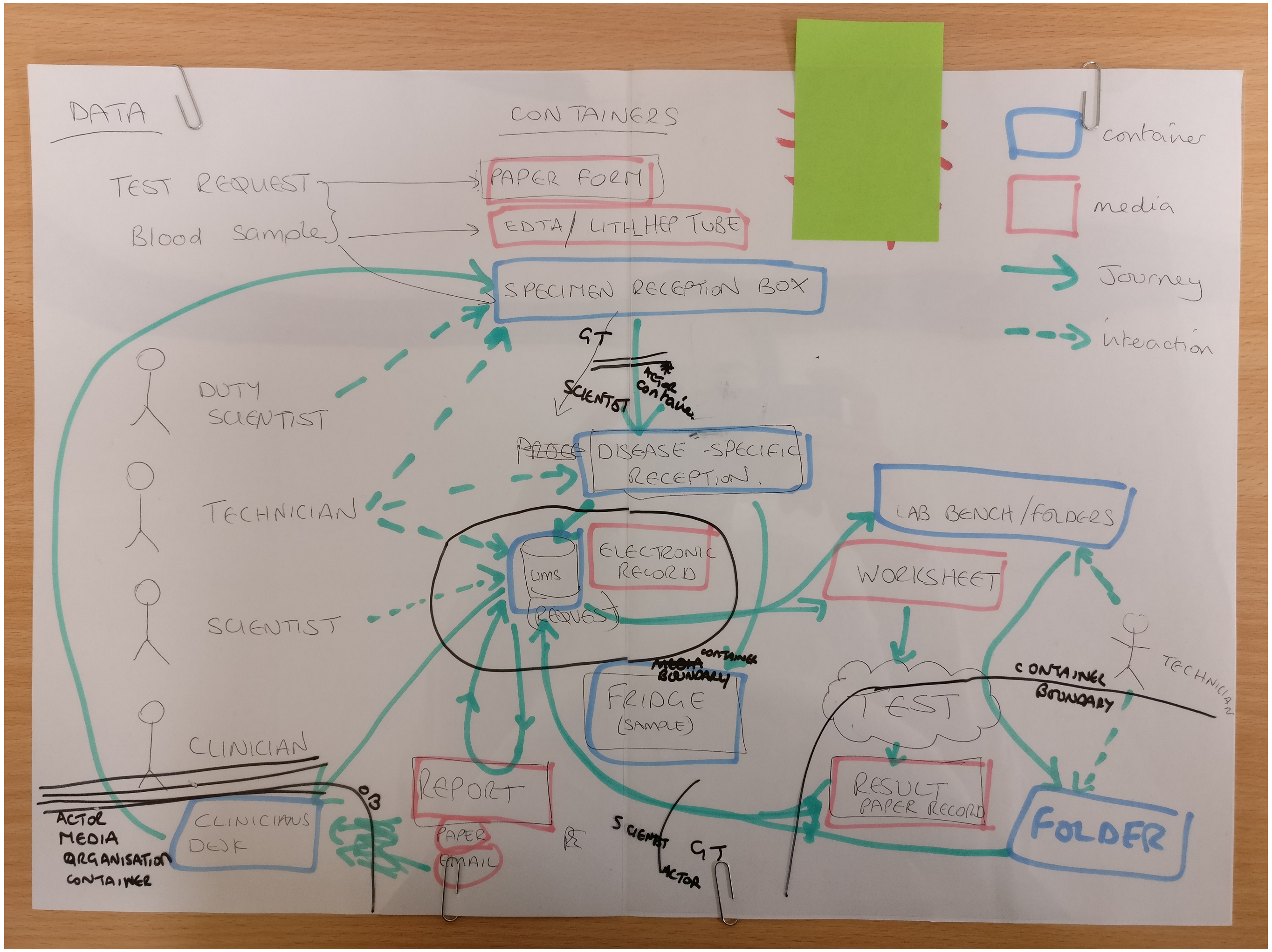


Figure B.6: Data journey model overlaying the organisational, media, and actors boundaries of Clinical Genomics Study B.

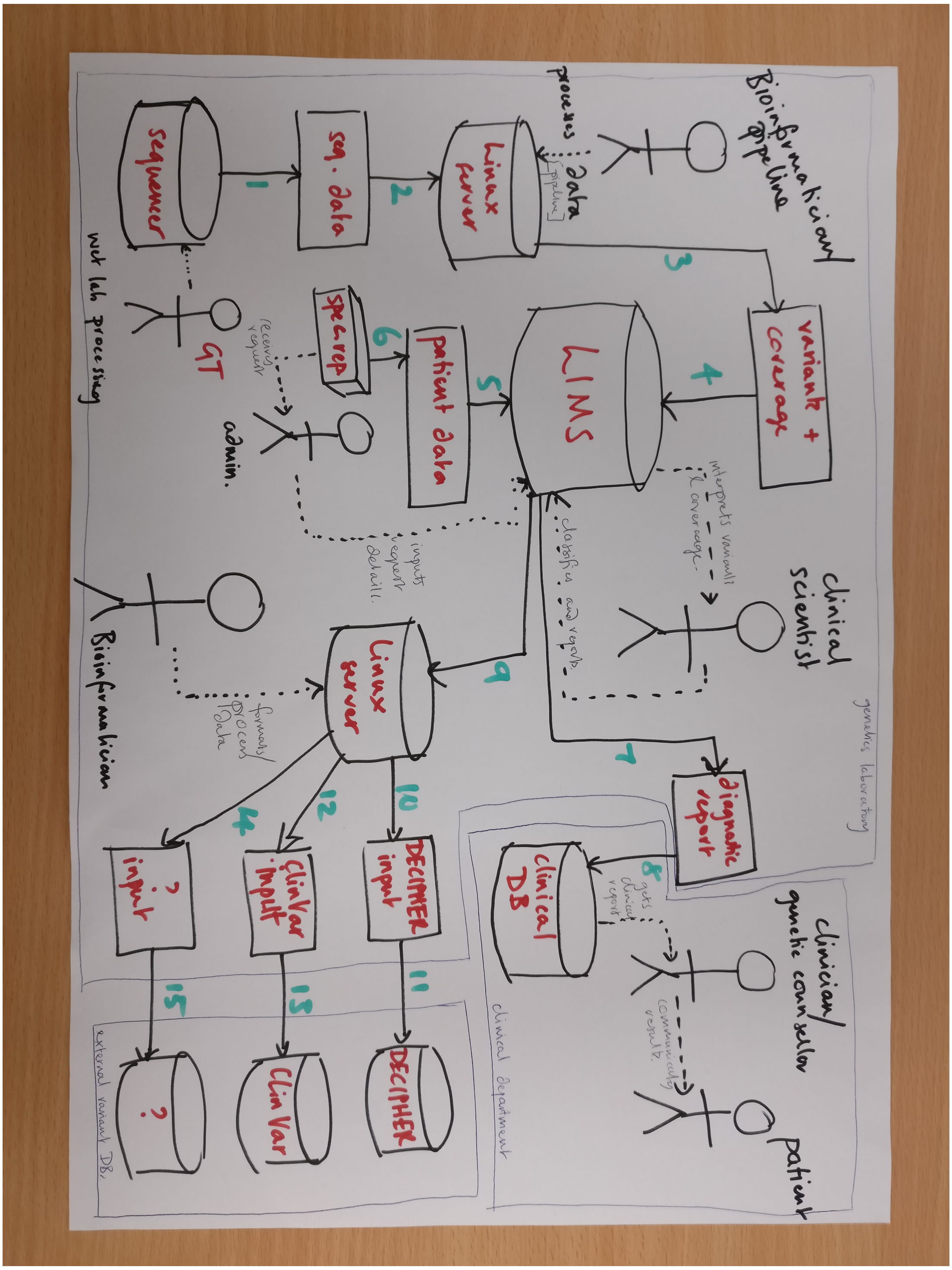


Figure B.7: Data journey model of Clinical Genomics Study C as designed by Clinical Consultant Managers in a nearby foundation trust.

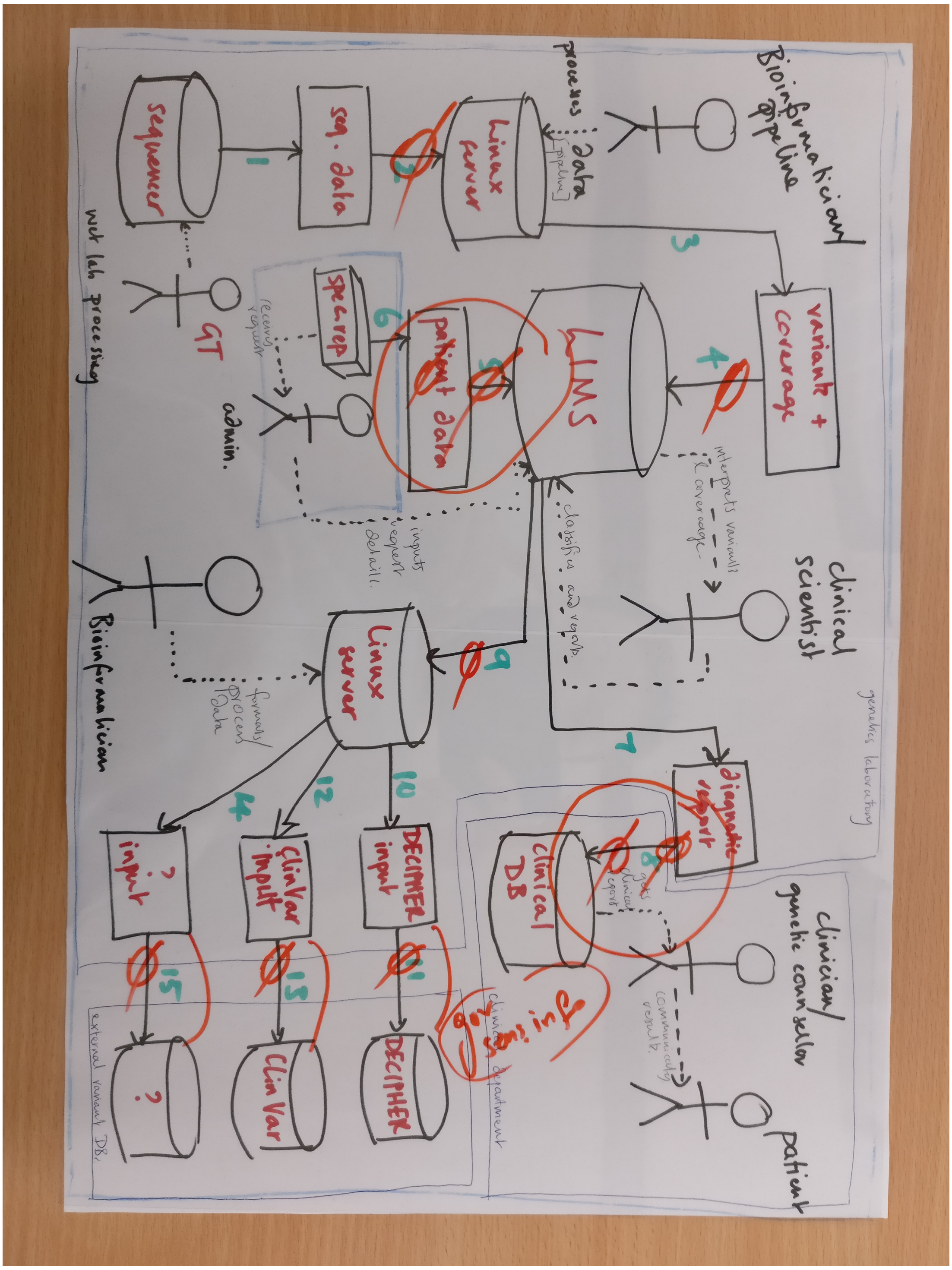


Figure B.8: Data journey model and boundaries of Clinical Genomics Study C.

Bibliography

[1] R. L. Ackoff. Towards a system of systems concepts. *Management science*, 17(11):661–671, 1971.

[2] R. S. Aguilar-Saven. Business process modelling: Review and framework. *International Journal of production economics*, 90(2):129–149, 2004.

[3] A. J. Albrecht and J. E. Gaffney. Software function, source lines of code, and development effort prediction: a software science validation. *IEEE transactions on software engineering*, (6):639–648, 1983.

[4] A. Alliance. Agile manifesto. *Online at http://www. agilemanifesto. org*, 6(1), 2001.

[5] S. Alter. Work system theory: overview of core concepts, extensions, and challenges for the future. *Journal of the Association for Information Systems*, 14(2):72, 2013.

[6] S. W. Ambler. *Process patterns: building large-scale systems using object technology*. Cambridge University Press, 1998.

[7] F. Artigas, D. Elefante, and A. Marti. Geographic information sharing: A regional approach in northern new jersey, usa. *Information Polity*, 14(1, 2):127–139, 2009.

[8] J. S. Ash and D. W. Bates. Factors and forces affecting ehr system adoption: report of a 2004 acmi discussion. *Journal of the American Medical Informatics Association*, 12(1):8–12, 2005.

[9] M. Banbura, D. Giannone, M. Modugno, and L. Reichlin. Now-casting and the real-time data flow. 2013.

[10] M. Banbura, D. Giannone, and L. Reichlin. Nowcasting. Working Papers ECARES ECARES 2010-021, ULB – Universite Libre de Bruxelles, 2010.

[11] S. Barrett and B. Konsynski. Inter-organization information sharing systems. *MIS Quarterly*, pages 93–105, 1982.

[12] J. Barwise and J. Seligman. *Information flow: the logic of distributed systems*, volume 44. Cambridge University Press, 1997.

[13] C. Batini, S. Ceri, and S. Navathe. *Entity Relationship Approach.* Elsevier Science Publishers BV (North Holland), 1989.

[14] C. Batini, E. Nardelli, and R. Tamassia. A layout algorithm for data flow diagrams. *Software Engineering, IEEE Transactions on*, (4):538–546, 1986.

[15] M. Y. Becker. Information governance in nhs's npfit: A case for policy specification. *International Journal of Medical Informatics*, 76(5):432–437, 2007.

[16] R. Becker, S. G. Eick, A. R. Wilks, et al. Visualizing network data. *Visualization and Computer Graphics, IEEE Transactions on*, 1(1):16–28, 1995.

[17] D. Bender and K. Sartipi. Hl7 fhir: An agile and restful approach to healthcare information exchange. In *Computer-Based Medical Systems (CBMS), 2013 IEEE 26th International Symposium on*, pages 326–331. IEEE, 2013.

[18] K. Bennett. Legacy systems: Coping with success. *IEEE software*, 12(1):19–23, 1995.

[19] T. Benson. Principles of health interoperability. *HL7 and SNOMED*, 2010.

[20] M. Berg. Implementing information systems in health care organizations: myths and challenges. *International journal of medical informatics*, 64(2):143–156, 2001.

[21] M. Birgersson, G. Hansson, and U. Franke. Data integration using machine learning. In *Enterprise Distributed Object Computing Workshop (EDOCW), 2016 IEEE 20th International*, pages 1–10. IEEE, 2016.

[22] J. Bisbal, D. Lawless, B. Wu, and J. Grimson. Legacy information systems: Issues and directions. *IEEE software*, 16(5):103–111, 1999.

[23] L. J. Black, P. R. Carlile, and N. P. Repenning. A dynamic theory of expertise and occupational boundaries in new technology implementation: Building on barley's study of ct scanning. *Administrative Science Quarterly*, 49(4):572–607, 2004.

[24] A. Bock, M. Kaczmarek, S. Overbeek, and M. Heß. A comparative analysis of selected enterprise modeling approaches. In *PoEM*, pages 148–163. Springer, 2014.

[25] B. Boehm. Software risk management. *ESEC'89*, pages 1–19, 1989.

[26] B. Boehm, C. Abts, and S. Chulani. Software development cost estimation approaches - a survey. *Annals of software engineering*, 10(1-4):177–205, 2000.

[27] B. W. Boehm. Software risk management: principles and practices. *IEEE software*, 8(1):32–41, 1991.

[28] B. W. Boehm et al. *Software engineering economics*, volume 197. Prentice-hall Englewood Cliffs (NJ), 1981.

[29] A. Boroumand, S. Ghose, Y. Kim, R. Ausavarungnirun, E. Shiu, R. Thakur, D. Kim, A. Kuusela, A. Knies, P. Ranganathan, et al. Google workloads for consumer devices: mitigating data movement bottlenecks. In *Proceedings of the Twenty-Third International Conference on Architectural Support for Programming Languages and Operating Systems*, pages 316–331. ACM, 2018.

[30] L. Bossavit. Project management, the movie. *Cutter IT Journal*, 16(12):18–23, 2003.

[31] M.-M. Bouamrane, A. Rector, and M. Hurrell. Using ontologies for an intelligent patient modelling, adaptation and management system. In *On the Move to Meaningful Internet Systems: OTM 2008*, pages 1458–1470. Springer, 2008.

[32] I. Bouty. Interpersonal and interaction influences on informal resource exchanges between r&d researchers across organizational boundaries. *Academy of Management Journal*, 43(1):50–65, 2000.

[33] V. Boyko, N. Rudnichenko, S. Kramskoy, Y. Hrechukha, and N. Shibaeva. Concept implementation of decision support software for the risk management of complex technical system. In *Advances in Intelligent Systems and Computing*, pages 255–269. Springer, 2017.

[34] J. Braithwaite. Changing how we think about healthcare improvement. *BMJ*, 361, 2018.

[35] V. Braun and V. Clarke. Using thematic analysis in psychology. *Qualitative research in psychology*, 3(2):77–101, 2006.

[36] K. Breitfelder and D. Messina. Ieee 100: the authoritative dictionary of ieee standards terms. *Standards Information Network IEEE Press. v879*, 2000.

[37] L. C. Briand, K. El Emam, D. Surmann, I. Wieczorek, and K. D. Maxwell. An assessment and comparison of common software cost estimation modeling techniques. In *Software Engineering, 1999. Proceedings of the 1999 International Conference on*, pages 313–323. IEEE, 1999.

[38] D. Budgen. *Software design*. Pearson Education, 2003.

[39] P. R. Carlile. Transferring, translating, and transforming: An integrative framework for managing knowledge across boundaries. *Organization science*, 15(5):555–568, 2004.

[40] M. J. Carr, S. L. Konda, I. Monarch, F. C. Ulrich, and C. F. Walker. Taxonomy-based risk identification. Technical report, CARNEGIE-MELLON UNIV PITTSBURGH PA SOFTWARE ENGINEERING INST, 1993.

[41] K. K. Chang, P. J. Nair, D. Lee, S. Ghose, M. K. Qureshi, and O. Mutlu. Low-cost inter-linked subarrays (lisa): Enabling fast inter-subarray data movement in dram. In *2016 IEEE International Symposium on High Performance Computer Architecture (HPCA)*, pages 568–580. IEEE, 2016.

[42] S. K. Chang. *Handbook of software engineering and knowledge engineering*, volume 1. World Scientific, 2001.

[43] Y. Charalabidis, S. Pantelopoulos, and Y. Koussos. Enabling interoperability of transactional enterprise applications. In *Workshop on Interoperability of Enterprise Systems, 18th European Conference on Object-Oriented Programming (ECOOP), Oslo*, pages 14–18, 2004.

[44] R. N. Charette. *Software engineering risk analysis and management*. Intertext Publications New York, 1989.

[45] R. N. Charette. Why software fails [software failure]. *IEEE Spectrum*, 42(9):42–49, Sept 2005.

[46] D. Chen, N. Daclin, et al. Framework for enterprise interoperability. In *Proc. of IFAC Workshop EI2N*, pages 77–88. Bordeaux, 2006.

[47] D. Chen and G. Doumeingts. European initiatives to develop interoperability of enterprise applicationsbasic concepts, framework and roadmap. *Annual reviews in control*, 27(2):153–162, 2003.

[48] D. Chen, G. Doumeingts, and F. Vernadat. Architectures for enterprise integration and interoperability: Past, present and future. *Computers in industry*, 59(7):647–659, 2008.

[49] P. P.-S. Chen. The entity-relationship modeltoward a unified view of data. *ACM Transactions on Database Systems (TODS)*, 1(1):9–36, 1976.

[50] Y. L. Chen et al. Data flow diagram. In *Modeling and Analysis of Enterprise and Information Systems*, pages 85–97. Springer, 2009.

[51] Z. Chen, A. Gangopadhyay, S. H. Holden, G. Karabatis, and M. P. McGuire. Semantic integration of government data for water quality management. *Government Information Quarterly*, 24(4):716–735, 2007.

[52] S. R. Chidamber and C. F. Kemerer. A metrics suite for object oriented design. *IEEE Transactions on software engineering*, 20(6):476–493, 1994.

[53] C. G. Cobb. *The project manager's guide to mastering Agile: Principles and practices for an adaptive approach*. John Wiley & Sons, 2015.

[54] A. Cockburn. *Agile software development*, volume 177. Addison-Wesley Boston, 2002.

[55] W. W. Cohen and J. Richman. Learning to match and cluster large high-dimensional data sets for data integration. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 475–480. ACM, 2002.

[56] M. Cohn. *Agile estimating and planning*. Pearson Education, 2005.

[57] P. Craig, P. Dieppe, S. Macintyre, S. Michie, I. Nazareth, and M. Petticrew. Developing and evaluating complex interventions: the new medical research council guidance. *Bmj*, 337:a1655, 2008.

[58] W. L. Currie and M. W. Guah. Conflicting institutional logics: a national programme for it in the organisational field of healthcare. *Journal of Information Technology*, 22(3):235–247, 2007.

[59] W. M. David. Bioinformatics: sequence and genome analysis. *Journal of Bioinformatics*, 28, 2001.

[60] J. L. Dietz. Towards a discipline of organisation engineering. *European Journal of Operational Research*, 128(2):351–363, 2001.

[61] J. L. Dietz. *Enterprise ontology: theory and methodology*. Springer Science & Business Media, 2006.

[62] M. Dixon-Woods, S. Agarwal, D. Jones, B. Young, and A. Sutton. Synthesising qualitative and quantitative evidence: a review of possible methods. *Journal of health services research & policy*, 10(1):45–53, 2005.

[63] A. Doan, P. Domingos, and A. Y. Halevy. Reconciling schemas of disparate data sources: A machine-learning approach. In *ACM Sigmod Record*, volume 30, pages 509–520. ACM, 2001.

[64] W. S. Donelson. Project-planning and control. *Datamation*, 22(6):73, 1976.

[65] I. Eleftheriou, S. Embury, and A. Brass. Data journey modelling: Predicting risk for IT developments. In *Proceedings of the 9th IFIP WG 8.1. Working Conference on the Practice of Enterprise Modeling, PoEM*, pages 72–86. Springer, 2016.

[66] I. Eleftheriou, S. Embury, and A. Brass. Light touch identification of cost/risk in complex socio-technical systems. In *Proceedings of the 10th IFIP WG 8.1. Working Conference on the Practice of Enterprise Modeling, PoEM*. Springer, 2017.

[67] I. Eleftheriou, S. Embury, R. Moden, P. Dobinson, and A. Brass. Data journeys: Identifying social and technical barriers to data movement in large, complex organisations. Technical report, 2016. Submitted Manuscript. Access link: www.datajourney.org/publications/tech_rep_data_journey.pdf.

[68] I. Eleftheriou, S. M. Embury, R. Moden, P. Dobinson, and A. Brass. Data journeys: Identifying social and technical barriers to data movement in large, complex organisations. *Journal of biomedical informatics*, 2017.

[69] J. A. Espinosa, J. N. Cummings, J. M. Wilson, and B. M. Pearce. Team boundary issues across multiple global firms. *Journal of Management Information Systems*, 19(4):157–190, 2003.

[70] J. L. Eveleens and C. Verhoef. The rise and fall of the chaos report figures. *IEEE software*, 27(1):30–36, 2010.

[71] O. K. Ferstl and E. J. Sinz. Som modeling of business systems. In *Handbook on Architectures of Information Systems*, pages 339–358. Springer, 1998.

[72] S. Flowers. *Software Failure: Management Failure: Amazing Stories and Cautionary Tales*. John Wiley & Sons, Inc., New York, NY, USA, 1996.

[73] S. Ford. Challenges to implementing npfit: Nothing counts except what is in front of the clinician to use. *BMJ: British Medical Journal*, 331(7515):516, 2005.

[74] U. Frank. *Multiperspektivische Unternehmensmodellierung: Theoretischer Hintergrund und Entwurf einer objektorientierten Entwicklungsumgebung*. Oldenbourg, 1994.

[75] U. Frank. Multi-perspective enterprise modeling (memo) conceptual framework and modeling languages. In *System Sciences, 2002. HICSS. Proceedings of the 35th Annual Hawaii International Conference on*, pages 1258–1267. IEEE, 2002.

[76] U. Frank. Multi-perspective enterprise modeling: foundational concepts, prospects and future research challenges. *Software & Systems Modeling*, 13(3):941–962, 2014.

[77] B. Gillham. *Case study research methods*. Bloomsbury Publishing, 2000.

[78] M. Gokhale, B. Holmes, and K. Iobst. Processing in memory: The terasys massively parallel pim array. *Computer*, 28(4):23–31, 1995.

[79] L. Goodson and M. Vassar. An overview of ethnography in healthcare and medical education research. *Journal of educational evaluation for health professions*, 8, 2011.

[80] T. Greenhalgh, H. W. Potts, G. Wong, P. Bark, and D. Swinglehurst. Tensions and paradoxes in electronic patient record research: A systematic literature review using the meta-narrative method. *Milbank Quarterly*, 87(4):729–788, 2009.

[81] D. J. Greenwood and M. Levin. *Introduction to action research: Social research for social change*. SAGE publications, 2006.

[82] G. Guest, K. M. MacQueen, and E. E. Namey. *Applied thematic analysis*. sage, 2011.

[83] A. Y. Halevy, N. Ashish, D. Bitton, M. Carey, D. Draper, J. Pollock, A. Rosenthal, and V. Sikka. Enterprise information integration: successes, challenges and controversies. In *Proceedings of the 2005 ACM SIGMOD international conference on Management of data*, pages 778–787. ACM, 2005.

[84] J. Hall and S. Koukoulas. Semantic interoperability for e-business in the isp service domain. In *ICE-B*, pages 390–396, 2008.

[85] M. H. Halstead. *Elements of software science*, volume 7. Elsevier New York, 1977.

[86] V. Haren. Vh publishing. archimate 2. 0 specification. the open group, 2012.

[87] R. Heeks. Health information systems: Failure, success and improvisation. *International journal of medical informatics*, 75(2):125–137, 2006.

[88] B. Henderson-Sellers. *Object-oriented metrics: measures of complexity*. Prentice-Hall, Inc., 1995.

[89] J. Hendy, B. C. Reeves, N. Fulop, A. Hutchings, and C. Masseria. Challenges to implementing the national programme for information technology (npfit): a qualitative study. *Bmj*, 331(7512):331–336, 2005.

[90] R. P. Higuera and Y. Y. Haimes. Software risk management. Technical report, CARNEGIE-MELLON UNIV PITTSBURGH PA SOFTWARE ENGINEERING INST, 1996.

[91] R. Hillestad, J. Bigelow, A. Bower, F. Girosi, R. Meili, R. Scoville, and R. Taylor. Can electronic medical record systems transform health care? potential health benefits, savings, and costs. *Health Affairs*, 24(5):1103–1117, 2005.

[92] E. Hossain, M. A. Babar, H.-y. Paik, and J. Verner. Risk identification and mitigation processes for using scrum in global software development: A conceptual framework. In *Software Engineering Conference, 2009. APSEC'09. Asia-Pacific*, pages 457–464. IEEE, 2009.

[93] C.-C. Hsu and B. A. Sandford. The delphi technique: making sense of consensus. *Practical assessment, research & evaluation*, 12(10):1–8, 2007.

[94] M. Jamshidi. *System of systems engineering: innovations for the twenty-first century*, volume 58. John Wiley & Sons, 2011.

[95] M. Jarke, M. A. Jeusfeld, C. Quix, and P. Vassiliadis. Architecture and quality in data warehouses: An extended repository approach. *Information Systems*, 24(3):229–253, 1999.

[96] P. Jayaratna and K. Sartipi. Hl7 v3 message extraction using semantic web techniques. *International Journal of Knowledge Engineering and Data Mining*, 2(1):89–115, 2012.

[97] C. Jones. Applied software measurement: Assuring. *Productivity and Quality*, 1997.

[98] M. Jørgensen and S. Grimstad. Software development effort estimation – demystifying and improving expert estimation. In *Simula Research Laboratory*, pages 381–403. Springer, 2010.

[99] M. Jørgensen and K. Moløkken-Østvold. How large are software cost overruns? a review of the 1994 chaos report. *Information and Software Technology*, 48(4):297–301, 2006.

[100] M. Jorgensen and M. Shepperd. A systematic review of software development cost estimation studies. *IEEE Transactions on software engineering*, 33(1), 2007.

[101] A. Josey. *TOGAF® Version 9.1 A Pocket Guide*. Van Haren, 2011.

[102] R. S. Kaplan and D. P. Norton. Transforming the balanced scorecard from performance measurement to strategic management: Part i. *Accounting horizons*, 15(1):87–104, 2001.

[103] C. Keating, R. Rogers, R. Unal, D. Dryer, A. Sousa-Poza, R. Safford, W. Peterson, and G. Rabadi. System of systems engineering. *Engineering Management Journal*, 15(3):36–45, 2003.

[104] S. W. Keckler, W. J. Dally, B. Khailany, M. Garland, and D. Glasco. Gpus and the future of parallel computing. *IEEE Micro*, (5):7–17, 2011.

[105] C. F. Kemerer. An empirical validation of software cost estimation models. *Communications of the ACM*, 30(5):416–429, 1987.

[106] S.-a. Knight and J. M. Burn. Developing a framework for assessing information quality on the world wide web. *Informing Science: International Journal of an Emerging Transdiscipline*, 8(5):159–172, 2005.

[107] A. Koenig. Patterns and antipatterns. *Journal of Object-Oriented Programming*, 8(1):46–48, 1995.

[108] K. Kosanke. Iso standards for interoperability: a comparison. In *Interoperability of enterprise software and applications*, pages 55–64. Springer, 2006.

[109] E. Koshy, V. Koshy, and H. Waterman. *Action research in healthcare*. Sage, 2010.

[110] V. Koshy. *Action research for improving practice: A practical guide*. Sage, 2005.

[111] D. J. Kupfer, D. A. Regier, and E. A. Kuhl. On the road to dsm-v and icd-11. *European Archives of Psychiatry and Clinical Neuroscience*, 258(5):2–6, 2008.

[112] F. Lampathaki, S. Koussouris, C. Agostinho, R. Jardim-Goncalves, Y. Charalabidis, and J. Psarras. Infusing scientific foundations into enterprise interoperability. *Computers in Industry*, 63(8):858–866, 2012.

[113] A. L. Lederer and J. Prasad. Perceptual congruence and information systems cost estimating. In *Proceedings of the 1995 ACM SIGCPR conference on Supporting teams, groups, and learning inside and outside the IS function reinventing IS*, pages 50–59. ACM, 1995.

[114] H. Leung and Z. Fan. Software cost estimation. *Handbook of Software Engineering, Hong Kong Polytechnic University*, pages 1–14, 2002.

[115] H. K. Leung. Quality metrics for intranet applications. *Information & Management*, 38(3):137–152, 2001.

[116] G. A. Lewis and L. Wrage. Model problems in technologies for interoperability: Web services. Technical report, CARNEGIE-MELLON UNIV PITTSBURGH PA SOFTWARE ENGINEERING INST, 2006.

[117] Y. Li and A. Ngom. Data integration in machine learning. In *Bioinformatics and Biomedicine (BIBM), 2015 IEEE International Conference on*, pages 1665–1671. IEEE, 2015.

[118] A. Liberati, D. G. Altman, J. Tetzlaff, C. Mulrow, P. C. Gøtzsche, J. P. Ioannidis, M. Clarke, P. J. Devereaux, J. Kleijnen, and D. Moher. The prisma statement for reporting systematic reviews and meta-analyses of studies that evaluate health care interventions: explanation and elaboration. *PLoS medicine*, 6(7):e1000100, 2009.

[119] G. H. Loh. 3d-stacked memory architectures for multi-core processors. In *ACM SIGARCH computer architecture news*, volume 36, pages 453–464. IEEE Computer Society, 2008.

[120] D. Loshin. *Enterprise knowledge management: The data quality approach*. Morgan Kaufmann, 2001.

[121] J. Lumpkin. Report on uniform data standards for patient medical record information, 2002.

[122] K. R. MacCrimmon and C. A. Ryavec. An analytical study of the pert assumptions. *Operations Research*, 12(1):16–37, 1964.

[123] D. M. MacKay. Towards an information-flow model of human behaviour. *British Journal of Psychology*, 47(1):30–43, 1956.

[124] R. C. Mahaney and A. L. Lederer. Runaway information systems projects and escalating commitment. In *Proceedings of the 1999 ACM SIGCPR conference on computer personnel research*, pages 291–296. ACM, 1999.

[125] M. W. Maier. Architecting principles for systems-of-systems. In *INCOSE International Symposium*, volume 6, pages 565–573. Wiley Online Library, 1996.

[126] A. L. Mark. Modernising healthcare–is the npfit for purpose? *Journal of Information Technology*, 22(3):248–256, 2007.

[127] B. Marr. *Key Performance Indicators (KPI): The 75 measures every manager needs to know*. Pearson UK, 2012.

[128] J. P. Marshall. The social (dis)organisation of software: Failure and disorder in information society. *The Australian Journal of Anthropology*, 25(2):190–206, 2014.

[129] N. Mays, C. Pope, and J. Popay. Systematically reviewing qualitative and quantitative evidence to inform management and policy-making in the health field. *Journal of health services research & policy*, 10(1_suppl):6–20, 2005.

[130] S. McConnell. *Software project survival guide*. Pearson Education, 1998.

[131] C. N. Mead et al. Data interchange standards in healthcare it-computable semantic interoperability: Now possible but still difficult. do we really need a better mousetrap? *Journal of Healthcare Information Management*, 20(1):71, 2006.

[132] E. Mendes. Effort and risk prediction for healthcare software projects delivered on the web. In *Practitioner's Knowledge Representation*. Springer, 2014. p. 107–122.

[133] J. Meyer. Using qualitative methods in health related action research. *Bmj*, 320(7228):178–181, 2000.

[134] A. Minkiewicz. Measuring object oriented software with predictive object points. *PRICE Systems, LLC*, 1997.

[135] C. Moraga, M. Moraga, C. Calero, and A. Caro. Square-aligned data quality model for web portals. In *Quality Software, 2009. QSIC'09. 9th International Conference on*, pages 117–122. IEEE, 2009.

[136] C. O. M. P. A. (MPA). Major projects authority programme assessment review of the national programme for it, 2011.

[137] A. C. Myers and B. Liskov. *A decentralized model for information flow control*, volume 31. ACM, 1997.

[138] A. Neely, M. Gregory, and K. Platts. Performance measurement system design: a literature review and research agenda. *International journal of operations & production management*, 15(4):80–116, 1995.

[139] M. Novakouski and G. A. Lewis. Interoperability in the e-government context. Technical report, CARNEGIE-MELLON UNIV PITTSBURGH PA SOFTWARE ENGINEERING INST, 2012.

[140] T. Pardo, A. M. Cresswell, S. S. Dawes, et al. Modeling the social & technical processes of interorganizational information integration. In *Proceedings of the 37th Annual Hawaii International Conference on System Sciences*, page 8. IEEE, 2004.

[141] T. A. Pardo, J. R. Gil-Garcia, and G. B. Burke. Sustainable cross-boundary information sharing. *Digital Government*, pages 421–438, 2008.

[142] R. Park. The central equations of the price software cost model. In *4th COCOMO Users Group Meeting*, 1988.

[143] M. Parker, V. Moleshe, R. De la Harpe, and G. Wills. An evaluation of information quality frameworks for the world wide web. 2006.

[144] C. N. Parkinson and R. C. Osborn. *Parkinson's law, and other studies in administration*, volume 24. Houghton Mifflin Boston, 1957.

[145] D. Parmenter. *Key performance indicators: developing, implementing, and using winning KPIs*. John Wiley & Sons, 2015.

[146] P. E. Peppard, T. Young, M. Palta, and J. Skatrud. Prospective study of the association between sleep-disordered breathing and hypertension. *New England Journal of Medicine*, 342(19):1378–1384, 2000.

[147] S. L. Pfleeger, F. Wu, and R. Lewis. *Software cost estimation and sizing methods: issues, and guidelines*, volume 269. Rand Corporation, 2005.

[148] C. L. Pritchard, P.-R. PMP, et al. *Risk management: concepts and guidance*. CRC Press, 2014.

[149] L. H. Putnam. A general empirical solution to the macro software sizing and estimating problem. *IEEE transactions on Software Engineering*, (4):345–361, 1978.

[150] L. H. Putnam and W. Myers. *Measures for excellence: reliable software on time, within budget*. Prentice Hall Professional Technical Reference, 1991.

[151] B. Randell. A computer scientist's reactions to npfit. *Journal of Information Technology*, 22(3):222–234, 2007.

[152] T. C. Redman. Data quality management past, present, and future: Towards a management system for data. In *Handbook of Data Quality*, pages 15–40. Springer, 2013.

[153] C. Rolland, S. Nurcan, and G. Grosz. Enterprise knowledge development: the process view. *Information & management*, 36(3):165–184, 1999.

[154] G. Ropohl. Philosophy of socio-technical systems. *Techné: Research in Philosophy and Technology*, 4(3):186–194, 1999.

[155] J. Rosenberg. Some misconceptions about lines of code. In *Software Metrics Symposium, 1997. Proceedings., Fourth International*, pages 137–142. IEEE, 1997.

[156] J. Rumbaugh, I. Jacobson, and G. Booch. *Unified Modeling Language Reference Manual, The*. Pearson Higher Education, 2004.

[157] J. Rushby. Software verification and system assurance. In *Software Engineering and Formal Methods, 2009 Seventh IEEE International Conference on*, pages 3–10. IEEE, 2009.

[158] K. Sandkuhl, M. Wißotzki, and J. Stirna. *Unternehmensmodellierung: Grundlagen, Methode und Praktiken*. Springer-Verlag, 2013.

[159] K. Sartipi and A. Dehmoobad. Cross-domain information and service interoperability. In *Proceedings of the 10th International Conference on Information Integration and Web-based Applications & Services*, pages 25–32. ACM, 2008.

[160] K. Sartipi and M. H. Yarmand. Standard-based data and service interoperability in ehealth systems. In *Software Maintenance, 2008. ICSM 2008. IEEE International Conference on*, pages 187–196. IEEE, 2008.

[161] A.-W. Scheer. *ARISModellierungsmethoden, Metamodelle, Anwendungen*. Springer-Verlag, 2013.

[162] B. Shahzad and A. S. Al-Mudimigh. Risk identification, mitigation and avoidance model for handling software risk. In *Computational Intelligence, Communication Systems and Networks (CICSyN), 2010 Second International Conference on*, pages 191–196. IEEE, 2010.

[163] M. Shepperd and C. Schofield. Estimating software project effort using analogies. *IEEE Transactions on software engineering*, 23(11):736–743, 1997.

[164] Y. L. Simmhan, B. Plale, and D. Gannon. A survey of data provenance in e-science. *ACM Sigmod Record*, 34(3):31–36, 2005.

[165] I. Sommerville, D. Cliff, R. Calinescu, J. Keen, T. Kelly, M. Kwiatkowska, J. Mcdermid, and R. Paige. Large-scale complex it systems. *Communications of the ACM*, 55(7):71–77, 2012.

[166] M. Stonebraker and I. F. Ilyas. Data integration: The current status and the way forward. *IEEE Data Eng. Bull.*, 41(2):3–9, 2018.

[167] K. Strike, K. El Emam, and N. Madhavji. Software cost estimation with incomplete data. *IEEE Transactions on Software Engineering*, 27(10):890–908, 2001.

[168] X. Tang, O. Kislal, M. Kandemir, and M. Karakoy. Data movement aware computation partitioning. In *Proceedings of the 50th Annual IEEE/ACM International Symposium on Microarchitecture*, pages 730–744. ACM, 2017.

[169] I. I. Technology. Software product quality - part 1: Quality model, 2001.

[170] I. I. Technology. Software product quality requirements and evaluation (square) - data quality model, 2008.

[171] A. Trendowicz. *Software Cost Estimation, Benchmarking, and Risk Assessment: The Software Decision-Makers' Guide to Predictable Software Development*. Springer Science & Business Media, 2013.

[172] A. Trendowicz and R. Jeffery. Principles of effort and cost estimation. In *Software project effort estimation*, pages 11–45. Springer, 2014.

[173] E. Trist. The evolution of socio-technical systems. *Occasional paper*, 2, 1981.

[174] R. Van Zeist and P. Hendriks. Specifying software quality with the extended iso model. *Software Quality Journal*, 5(4):273–284, 1996.

[175] J. L. Vann. Resistance to change and the language of public organizations: A look at "clashing grammars" in large-scale information technology projects. *Public Organization Review*, 4(1):47–73, 2004.

[176] H. Veer and A. Wiles. Achieving technical interoperability-the etsi approach, european telecommunications standards institute, 2008.

[177] F. B. Vernadat. Interoperable enterprise systems: Principles, concepts, and methods. *Annual reviews in Control*, 31(1):137–145, 2007.

[178] C. E. Walston and C. P. Felix. A method of programming measurement and estimation. *IBM Systems Journal*, 16(1):54–73, 1977.

[179] R. Y. Wang and D. M. Strong. Beyond accuracy: What data quality means to data consumers. *Journal of management information systems*, pages 5–33, 1996.

[180] S. J. Wang, B. Middleton, L. A. Prosser, C. G. Bardon, C. D. Spurr, P. J. Carchidi, A. F. Kittler, R. C. Goldszer, D. G. Fairchild, A. J. Sussman, et al. A cost-benefit analysis of electronic medical records in primary care. *The American journal of medicine*, 114(5):397–403, 2003.

[181] P. T. Ward. The transformation schema: An extension of the data flow diagram to represent control and timing. *Software Engineering, IEEE Transactions on*, (2):198–210, 1986.

[182] R. Winter. Business engineering-auf dem weg zum unternehmen des informationszeitalters, 2000.

[183] T.-M. Yang, T. Pardo, et al. How is information shared across boundaries? In *Proceedings of the 44th Hawaii International Conference on System Sciences (HICSS)*, pages 1–10. IEEE, 2011.

[184] T.-M. Yang, L. Zheng, and T. Pardo. The boundaries of information sharing and integration: A case study of taiwan e-government. *Government Information Quarterly*, 29:S51–S60, 2012.

[185] R. R. Young. Recommended requirements gathering practices. *CROSSTALK The Journal of Defense Software Engineering*, 15(4):9–12, 2002.

[186] E. S. Yu. Social modeling and i*. In *Conceptual Modeling: Foundations and Applications*, pages 99–121. Springer, 2009.

[187] M. M. Yusof, J. Kuljis, A. Papazafeiropoulou, and L. K. Stergioulas. An evaluation framework for health information systems: human, organization and technology-fit factors (hot-fit). *International journal of medical informatics*, 77(6):386–398, 2008.

[188] S. O. Zandieh, K. Yoon-Flannery, G. J. Kuperman, D. J. Langsam, D. Hyman, and R. Kaushal. Challenges to ehr implementation in electronic-versus paper-based office practices. *Journal of General Internal Medicine*, 23(6):755–761, 2008.

[189] L. Zheng, T.-M. Yang, T. Pardo, and Y. Jiang. Understanding the “boundary” in information sharing and integration. In *Proceedings of the 42nd Hawaii International Conference on System Sciences (HICSS’09)*, pages 1–10. IEEE, 2009.