



Delft University of Technology

Designing context-aware architectures for business-to-government information sharing

van Engelenburg, Sélinde

DOI

[10.4233/uuid:d25fd4fd-02d7-4811-b675-615badbb3c05](https://doi.org/10.4233/uuid:d25fd4fd-02d7-4811-b675-615badbb3c05)

Publication date

2019

Document Version

Final published version

Citation (APA)

van Engelenburg, S. (2019). *Designing context-aware architectures for business-to-government information sharing*. [Dissertation (TU Delft), Delft University of Technology]. <https://doi.org/10.4233/uuid:d25fd4fd-02d7-4811-b675-615badbb3c05>

Important note

To cite this publication, please use the final published version (if applicable).
Please check the document version above.

Copyright

Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

Takedown policy

Please contact us and provide details if you believe this document breaches copyrights.
We will remove access to the work immediately and investigate your claim.

Designing context-aware architectures for business- to-government information sharing

Sélinde van Engelenburg



**Designing context-aware architectures for business-to-government information
sharing**

Dissertation

for the purpose of obtaining the degree of doctor
at Delft University of Technology
by the authority of the Rector Magnificus prof.dr.ir. T.H.J.J. van der Hagen
chair of the Board for Doctorates
to be defended publicly on
Wednesday 11 September 2019 at 10.00 o'clock

by

Selinde Helena VAN ENGELENBURG
Master of Science in Artificial Intelligence, Utrecht University, the Netherlands
born in Purmerend, the Netherlands

This dissertation has been approved by the promotor.

Composition of the doctoral committee:

Rector Magnificus,	chairperson
Prof.dr.ir. M.F.W.H.A. Janssen	Delft University of Technology, promotor
Dr.ing. A.J. Klievink	Delft University of Technology, copromotor

Independent members:

Prof.dr. Y.H. Tan	Delft University of Technology
Prof.dr. F.G.M. Smeele	Erasmus University Rotterdam
Prof.dr. M.A. Wimmer	University of Koblenz-Landau
Assoc.prof.dr. B.B. Shishkov	University of Library Studies and Information Technologies Bulgaria Institute of Mathematics and Informatics - BAS Bulgaria
Dr. M.E. Warnier	Delft University of Technology

Keywords: information sharing architecture, e-government, international container shipping, blockchain, design method, context, context-aware system, business-to-government information sharing

This work was funded by the Netherlands Organisation for Scientific Research (NWO) under grant number 438-13-601.

ISBN: 978 94 6380 462 2

Printed by ProefschriftMaken || www.proefschriftmaken.nl

Copyright © 2019 by S.H. van Engelenburg. All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electrical, mechanical, photocopying, recording or otherwise, without the prior permission of the copyright owner.

Table of Contents

Summary	1
Samenvatting.....	9
 PART I: INTRODUCTION AND OVERARCHING RESEARCH APPROACH	18
 1 Introduction	19
1.1 Information sharing in complex environments.....	21
1.2 Definitions.....	25
1.3 Problem statement, objective and research questions	32
1.4 Relationships between the artefacts and parts of this dissertation.....	34
 2 Research approach and methodology.....	37
2.1 Research philosophy	37
2.2 Design science research approach.....	48
2.3 Research methodology	51
 PART II: A METHOD FOR DESIGNING CONTEXT-AWARE SYSTEMS	57
 3 The need for a method for designing context-aware systems	58
3.1 Designing context-aware systems for complex environments	58
3.2 Objectives for the method	59
3.3 Related work on designing context-aware systems	60
 4 Design process for the method for designing context-aware systems	66
4.1 Activity 1 and 2: Problem identification, motivation and objectives of a solution	66
4.2 Activity 3: Design and development	67
4.3 Activity 4: Demonstration.....	68
4.4 Activity 5: Evaluation	68
4.5 Activity 6: Communication.....	71
 5 A definition of context.....	72
5.1 Definitions of context in literature	72
5.2 Formalisation.....	75
5.3 Environment elements, situations and the focus of a context.....	79
5.4 Context relationships, context elements and context.....	83
 6 A method for designing context-aware systems.....	87
6.1 Step 1: Getting insight into context.....	87
6.2 Step 2: Determining the components needed to sense and adapt to context	96
6.3 Step 3: Determining the rules for reasoning with context information	102

PART III: A CONTEXT-AWARE ARCHITECTURE FOR B2G INFORMATION SHARING IN THE CONTAINER-SHIPPING DOMAIN 106

7	B2G information sharing in the international container-shipping domain..	107
7.1	Possible benefits and risks of B2G information sharing in container shipping ...	107
7.2	The complex environment of B2G information sharing in container shipping ...	115
7.3	Related initiatives to support B2G information sharing	123
7.4	The need for a context-aware B2G information sharing architecture	126
7.5	Goals and foci for the context-aware B2G information sharing architecture	127
8	Design process for the context-aware architecture.....	130
8.1	Previous designs of the architecture	130
8.2	Activity 1: Problem identification and motivation	132
8.3	Activity 2: Define the objective of a solution.....	133
8.4	Activity 3: Design and development	137
8.5	Activity 4: Demonstration	139
8.6	Activity 5: Evaluation.....	139
8.7	Activity 6: Communication	142
9	The context of B2G information sharing in the container-shipping domain	144
9.1	Focus 1: The willingness of businesses to participate in the information flow ...	144
9.2	Focus 2: The lawfulness of the information flow provided by the architecture ..	164
9.3	Validity and reliability.....	182
10	Sensors, adaptors and context rules for the context-aware architecture.....	185
10.1	Adaptors	185
10.2	Sensors.....	192
10.3	Context rules.....	201
11	The basic components for context-awareness	205
11.1	Architectural pattern and basic components	205
11.2	Meeting the goals for the architecture	207
11.3	Blockchain technology and its typical characteristics	208
11.4	Decision component: information flow planner	216
11.5	A blockchain-based context information repository	218
12	A context-aware architecture for B2G information sharing.....	222
12.1	The overall context-aware architecture for B2G information sharing.....	222
12.2	Demonstration of the architecture	223

PART IV: EVALUATION, CONCLUSION AND FUTURE RESEARCH 228

13	Evaluation of the architecture	229
13.1	Interviews and workshops	229
13.2	The lawfulness of information sharing using the architecture	230

13.3 The willingness of businesses to use the architecture	231
13.4 The usefulness of the architecture to customs for compliance monitoring	241
13.5 Discussion and Input for the next design cycle	243
14 Evaluation of the method.....	245
14.1 Case study design	245
14.2 Results of the evaluation of the method	248
14.3 Discussion and Input for the next design cycle	255
15 Conclusions	258
15.1 Answering the research questions and solving the research problem	258
15.2 Scientific contribution	260
15.3 Research limitations	263
15.4 Recommendations for future research on context-aware systems.....	266
15.5 Reflections on technology hype in the field of ICT	267
REFERENCES	268
APPENDICES.....	285
Appendix A: Search query and overview of literature on context-aware systems	285
Appendix B: Overview of context elements and objects.....	292
ACKNOWLEDGEMENTS	298
CURRICULUM VITAE	300

Summary

New developments in Information and Communication Technology (ICT), such as big data, the Internet of Things (IoT), and blockchain technology provide opportunities for businesses and government organisations to benefit from business-to-government (B2G) information sharing. For example, big data analytics might provide government organisations with knowledge on how to assess risks using the information they receive from businesses. However, B2G information sharing can entail risks as well. Sensitive data could fall in the wrong hands and by that the competitor of the business might obtain this data. In addition, B2G information sharing could be unlawful.

These new technologies can make the environment in which information sharing takes place more complex. Complex multi-actor environments are characterised by consisting of a high variety of elements with different properties, including many different actors. For instance, in a complex environment, there might be various IoT sensors involved in information sharing, that generate a variety of data and that are governed by a variety of parties.

In such complex environments, likely more situations occur that require information sharing via different information flows to reduce the risks of sharing for businesses and to ensure that information sharing is lawful. For example, if a higher variety of data needs to be shared, then there are higher chances that there are different types of data that need to be shared in a different way. If data is competitively sensitive to a business, for instance, then it might need to be ensured that the information is shared in a flow which their competitor cannot access. Similarly, when a higher variety of parties are involved, then there are higher chances that these include different types of parties that experience different benefits and risks of information sharing and that have different requirements for that reason (e.g., different types of businesses and government organisations).

To support B2G information sharing in different situations in complex multi-actor environments, the flow of information needs to be adapted to the situation. What a flow of information looks like depends on the systems that the information goes through and their connections. The software architecture determines what flow of information an information system provides for. This means that we need the architecture of the system itself to adapt to the context and provide for the appropriate information flow in different situations. We thus need a context-aware architecture. The problem we address in this research is that *there is a lack of knowledge on what the design of context-aware architectures that support business-to-government information sharing in complex environments should look like*. Our corresponding objective is to *create a design for context-aware architectures that support business-to-government information sharing in complex environments*.

There are two views on design: design as a process and design as an artefact resulting from that process. The quality of the design artefact depends on the quality of the design process. Furthermore, the quality of the design artefact can provide information on the quality of the design process. This means that both views on design are important for reaching our objective.

A context-aware architecture requires sensors to sense the context. In addition, a context-aware architecture requires adaptors to adapt the information flow to the context. And finally, context rules are needed as well to prescribe what adaptations the adaptors need to make based on the context information obtained by the sensors.

To design context-aware architectures, it needs to be determined what sensors, adaptors and context rules for adapting to context should be included in the design. For that, insight is required into context at design time. However, in complex environments, a high number of elements could potentially belong to the context that is relevant to take into account. This means that two things are important: a systematic investigation of context and a way to decide quickly on what is relevant for the design of the architecture. The information gathered in this way should be used to model the context, and then the sensors, adaptors and rules can be derived from this model.

There is a gap in knowledge on how to model and systematically investigate the context in complex environments. The first research question we need to answer is thus: *“What should the design process of context-aware architectures supporting business-to-government information sharing in complex environments look like?”* To answer this question, we developed a design process, i.e., a method, for designing context-aware systems in complex multi-actor environments. This method is used in the design process for the context-aware architecture that we also developed as part of this research. However, it can also be part of the design process of other context-aware architectures and systems in complex environments.

In addition to the sensors, adaptors and rules, a context-aware B2G information sharing architecture also has some other components. More specifically, these are the components required to store context information, make decisions on what flow of information is appropriate for sharing, and to share information according to that flow. These components need to be connected to each other and with the adaptors and sensors to provide for an overall context-aware B2G information sharing architecture. To obtain our objective, the second research question we need to answer is *“What should a context-aware architecture that supports business-to-government information sharing in a complex environment look like?”*

The method and the context-aware B2G information sharing architecture are two different artefacts and thus are designed in their own design processes. The method is used for designing the architecture. Furthermore, the evaluation of the architecture provides information on the quality of the method. For both design processes, we relied on the activities for performing design science research specified by Peffers et al. (2007), viz. 1) problem identification and motivation, 2) define the objectives for a solution, 3) design and development, 4) demonstration, 5) evaluation, and 6) communication.

A method for designing context-aware systems

To identify the problem and the motivation for solving it, we performed an analysis of the problems having to do with determining what belongs to the relevant context in complex environments. We concluded that these environments contain many elements that are possibly relevant and for which this needs to be determined. This risks the effectiveness (i.e., not reaching design goals) and the efficiency (i.e., spending a lot of

effort on deciding if elements are relevant) of the design process. To reduce these risks, the method needs to meet the following objectives: 1) supporting the designer in systematically investigating and modelling the relevant context for their system and 2) supporting the designer in deriving the sensors, adaptors and context rules their system requires from their model of context.

Design of the method

To develop the method, first, we required a definition of context that can be used to make easy decisions on what belongs to the relevant context and to model the relevant context. In a review of the literature, we could not find a definition of context that is suitable as a basis for the method. A new, pragmatic definition of context was thus needed.

The next step was to develop a new definition of context. The problem of how to deal with real-world complex multi-actor environments when designing context-aware systems requires a practical point of view. Therefore, the definition of context that we present in this research relies on a pragmatic paradigm. According to our new definition, something belongs to context if it can affect whether the designer of the system reaches their goal.

We based our definition of context on definitions of several other notions. *Environment elements* are relationships between different objects in the environment of a system (e.g., what businesses are competitors). A *situation* is a state of the world in which certain elements in the environment are true and certain relationships exist (e.g., business *A* is a competitor of business *B*). The *focus* of a context is an environment element that needs to be true for the designer to reach their goal (e.g., the flow of information is lawful). A *context relationship* is a relationship between a focus and a set of environment elements. This relationship is such that in every situation where these environment elements have a certain value, the focus has the same value as well (e.g., if business *A* and business *B* are competitors and *B* can access competitively sensitive information from *A*, then the flow of information is unlawful). A *context element of a focus* is an environment element that is one of the environment elements that has a context relationship with a focus (e.g., what businesses are competitors). Finally, the *context of a focus* consists of its context elements.

We formalised the definition of context using the syntax of the logic-programming paradigm. This enforces a high level of specificity of the definitions. Furthermore, the syntax provides a language for expressing information on context by the designer at design time, as well as in the context-aware system at runtime.

We developed the method for designing context-aware systems in complex environments using this definition of context. The method consists of three steps, viz. 1) getting insight into context, 2) determining the components needed to sense context and adapt to context, and 3) determining the rules for reasoning with context information.

Step 1 is divided into three sub-steps. In step 1.1, a designer derives the foci for their system from their design goal. In step 1.2, they gather data on what situations restrict the value of a focus. If they find data suggesting that something restricts the value of a focus, they have found a context relationship. In step 1.3, the designer analyses the gathered data and derives context elements from the context relationships identified.

Step 2 is divided into two sub-steps. In step 2.1, the designer establishes for each context relationship what context element could be manipulated by the context-aware system in order to adapt and ensure that the value of the focus is in accordance with their design goal. Then, they determine what adaptor the context-aware system needs as a component to perform this manipulation. In step 2.2, the designer determines for the remaining context elements how they can be sensed and what sensors need to be a component in the context-aware system to do so.

In step 3, a context rule is derived from each context relationship. The body of these rules expresses a situation or a set of context elements, that can be sensed by the sensors of the context-aware system. If this situation is detected by the sensors, then an adaptor in the system should perform an action to make the context element in the head of the rule true.

Demonstration and evaluation of the method

The method was demonstrated by using it to develop a context-aware B2G information sharing architecture in a complex environment, namely international container shipping. To evaluate the feasibility of the method we performed a case study with cases in which the method was used. The first case is that of the B2G information sharing architecture we developed as part of this research. In this case, the author of this dissertation used the method herself. For this case, we evaluated the effectiveness of the method by validating the context model we built using the method. For the validation, we performed expert interviews.

For the second case, a designer not involved in this research used the method to develop a decentralised data marketplace. This designer provided a reflection on the effectiveness and efficiency of the design process using the method. In the third case, the method was used by another designer not involved in this research to develop a context-aware urban transport system. We interviewed this designer to determine their experience with applying the method and the effect of the method on the efficiency and effectiveness of their design process in particular.

Based on the results of the case study, we can conclude that the method is useful to designers due to its systematic and structured steps. The method helped to provide for an effective and rigorous design process. In addition, it enhanced efficiency in the cases, as with the method, decisions to take into account elements are made explicitly and early in the process. However, it takes some effort to apply the method. Hence, it should only be used when designing context-aware systems in highly complex environments.

A context-aware architecture for B2G information sharing in the container-shipping domain

The international container-shipping domain provides a typical instance of a complex environment and the related issues with information sharing. By developing a context-aware architecture for B2G information sharing in this domain, we thus solve a typical instance of the overall research problem of this research. This contributes to answering our second research question.

To specify further the design problem, we studied the ample literature that is available on information sharing in the international container-shipping domain. In the international container-shipping domain, customs is tasked with monitoring the flow of goods and businesses' compliance with laws and regulations. Customs needs high-quality information from businesses to use for risk assessment and to target high-risk containers for inspection. The law requires businesses to share some information with customs. However, businesses have more information available that would be useful for customs as well. We expect that if such information is shared with customs, the risk that businesses that are not compliant will be caught will increase and that therefore their compliance will improve. This, in turn, might have a positive impact on society.

Businesses gather high-quality information to base their own operations on. Customs could reuse this information to perform risk assessment. However, as they do not want to obligate businesses to share additional information with them, businesses need to share this additional information voluntarily. Furthermore, such information sharing should be lawful. This might not always be the case. For example, information sharing might be unlawful when it is possible for competitors to have access to each other's data.

International container shipping provides a complex environment for information sharing. This thus means that in different situations, different information flows are required to ensure that businesses are willing to participate in information sharing and to ensure that information sharing is lawful. Supporting B2G information sharing in this domain thus requires a context-aware architecture.

The objectives of the context-aware architecture

To define the objectives of a solution, we applied step 1 of the new method. From the problem specification, we derived two main goals for the architecture, viz. 1) to provide for information flows in which businesses supplying information are willing to participate and 2) to provide for information flows that are lawful. In accordance with the method, we derived two corresponding foci of context for the architecture, viz. 1) the willingness of businesses supplying information to participate in the information flow provided by the architecture, and 2) the lawfulness of the information flow provided by the architecture.

To gather information on the willingness of businesses to participate in information sharing, we performed a case study. The units of analysis were information flows between businesses that customs can piggyback on and information flows from businesses to customs. The cases were information flows with information sharing architectures implemented in the Cassandra project and in the CORE project. We gathered data on what impacted businesses' willingness to participate in these information flows based on a study of the documentation in these projects and on interviews with researchers and staff working at a business involved in the Cassandra project. In total, we derived twelve context relationships. Each of these context relationships describes a situation that affects whether businesses are willing to participate in an information flow and that thus affects whether the design goal is reached.

To gather information on the lawfulness of information flows, we performed interviews with juridical experts with a background in competition law, customs law and Intellectual Property (IP) law. Based on these interviews, we identified eight context relationships. Each of these context relationships describes a situation that affects the lawfulness of an information flow.

Design of the context-aware architecture

To design the context-aware architecture, we used the new method to derive the required sensors, adaptors and the context rules from the context relationships. Secondly, we determined what the components for storing context information and making decisions on how to adapt should look like. For designing this part of the architecture, we relied on the insight we gained in previous design cycles in the research about the willingness of businesses to participate in information sharing and its lawfulness. We then combined all components into the overall context-aware architecture.

In the context-aware architecture we designed, context information can be provided by businesses, independent third parties and customs. The businesses provide context information on their relationships and agreements with other businesses. Furthermore, they provide information on the properties of data elements that are shared via the architecture. In addition, they provide information on what data they consider sensitive and on who they are willing to share their data with and under which circumstances. The latter is expressed using business rules.

The independent third parties consist of two groups. The first group consists of identity managers. These parties provide context information on the identities of parties involved in information sharing. The second group consists of trusted parties. They provide context information that is difficult to obtain by businesses or that is safer to let an independent party provide. Customs provides context information on the obligations businesses have to share information with them.

The context information provided by these parties is stored on a blockchain. In addition, context rules and a history of access to data by different parties are stored on the blockchain as well. This blockchain is distributed in a network that only consists of certified parties that are allowed to use the architecture and participate in information sharing. The consensus mechanism relies on parties checking whether data was added to the blockchain by a party that is certified. Furthermore, the data stored on the blockchain is encrypted.

The decision component of the architecture obtains the context information from the blockchain, as well as the context rules and the access history. Based on this, it decides on what is the appropriate flow of information according to which to share a set of data element. This flow of information consists of a sequence of systems, where the information should be sent from one system to the next. The flow can include systems of businesses and government organisations, but also systems that provide certain functionality, such as data pipelines. The decision component, in our case, is called an information flow planner. The information flow planner stores the information flow that it decides is appropriate on the blockchain.

The information is shared according to the information flow that is proposed by the planner using information routers. Each system that is connected to the architecture has an information router and is a node in the blockchain network. To share information, the router obtains the information flow from the blockchain. It looks up what information router is next in the information flow. Then, it sends the information to the next router in the proposed flow and adds access history to the blockchain stating that the system of the next information router now has access to the information. The next router sends the information further in the same manner.

The architecture, in addition, contains three adaptors, namely an encryption component, a component that makes information flows thin, and a component that lets users only view data instead of storing it. This allows the architecture to adapt further the flow of information by sending the information via one of these adaptors to encrypt it, make it thin, or make it viewable without the need to store it.

Demonstration of the context-aware architecture

Before evaluating the architecture, we demonstrated how it could be used in several scenarios. The first scenario shows how the architecture can be used to support the sharing of the Entry Summary Declaration (ENS). In the second scenario, the architecture is used to share an update to the ENS. In the third scenario, an invoice is shared with customs using the context-aware architecture. The second and third scenarios were generated based on an interview with a policy advisor at customs in which we established what information sharing could be useful for them for risk assessment.

Evaluation of the context-aware architecture

We evaluated the architecture as well as the context rules. The aim of the evaluation was to determine whether the architecture, in fact, provides for information flows in which businesses are willing to participate and that are lawful. We validated the context rules based on an interview with a juridical expert and an interview with a researcher with expertise in information sharing in international container shipping. To evaluate the overall architecture, we gathered data from three sources: 1) workshops at Maersk Line, 2) interviews with an expert in formal law at Dutch customs and an expert from academia in trade law, and 3) an interview with an expert in IT and governance.

The results from the evaluation indicate that the context-aware B2G information sharing architecture is useful to businesses and customs, as it provides for more timely information sharing and direct access control. Furthermore, the architecture adapts to the context to ensure that businesses are willing to share in various situations and to ensure that sharing is lawful. In addition, we established that there are no obvious juridical issues with the way in which the architecture provides for different flows of information (e.g., with storing context information or using a decision component). The results also suggest that an important next step is investigating how the architecture can be implemented in practice at a large scale and how to arrange governance of the architecture.

Conclusions and implications of this research

This research has three main scientific contributions: 1) a definition of context, 2) a method for designing context-aware systems, and 3) a context-aware architecture for B2G information sharing for international container shipping. The new method for designing context-aware systems provides steps to systematically investigate context and derive a design from that. This can contribute to the efficiency and effectiveness of the design process in complex environments. The method relies on a pragmatic definition of context that is highly specific and provides a conceptual framework that can be used at design time, as well as during runtime. Furthermore, the B2G information sharing architecture for international container shipping provides more insight into what context-aware information sharing architectures can look like. To develop context-aware B2G information sharing architectures for other domains, the basic architecture can be used as a reference architecture, and the new method can be used to determine the sensors, adaptors and context rules that are appropriate in the domain.

By supporting B2G information sharing, this research aims to contribute to society as well. Government organisations, such as customs, can use additional high-quality information from businesses to improve their risk assessment. This can directly have an impact on safety and security. In addition, we expect that it will improve businesses' compliance, as their risk to be caught if they are not compliant will increase. This might have an additional positive effect on safety and security.

Samenvatting

Nieuwe ontwikkelingen in het veld van informatie- en communicatietechnologie (ICT), zoals big data, Internet of Things (IoT) en blockchain technologie zorgen voor nieuwe mogelijkheden voor bedrijven en overheden om voordelen te behalen uit het delen van informatie. Het delen van informatie tussen bedrijven en de overheid wordt ook wel business-to-government (B2G) delen van informatie genoemd. Analyse van big data kan overheden bijvoorbeeld informatie verschaffen over hoe zij risico's kunnen inschatten op basis van informatie die zij ontvangen van bedrijven. Echter, het B2G delen van informatie brengt ook risico's met zich mee. Gevoelige informatie zou in de verkeerde handen kunnen vallen en de concurrent zou deze informatie kunnen verkrijgen. Daarnaast kan het B2G delen van informatie onwettig zijn.

Deze nieuwe technologieën kunnen de omgeving waarin informatie gedeeld wordt complexer maken. Complexe omgevingen met meerdere actoren worden gekenmerkt door een grote verscheidenheid aan elementen met verschillende eigenschappen, waaronder veel verschillende actoren. In een complexe omgeving kan bijvoorbeeld data worden gedeeld van verschillende IoT sensoren die verschillende soorten informatie genereren en die beheerd worden door verschillende partijen.

In zulke complexe omgevingen is de kans groter dat zich vaker situaties voordoen die vereisen dat informatie via verschillende informatiestromen wordt gedeeld om de risico's van het delen voor bedrijven te verkleinen en ervoor te zorgen dat het delen van informatie wettig is. Als er een grotere variatie van informatie gedeeld moet worden, bijvoorbeeld, dan is de kans groter dat er verschillende soorten informatie tussen zitten die op verschillende manieren gedeeld moeten worden. Als informatie bijvoorbeeld concurrentiegevoelig is voor een bedrijf, dan moet er mogelijk voor worden gezorgd dat de informatie gedeeld wordt in een informatiestroom waarin hun concurrent geen toegang heeft tot de informatie. Op dezelfde manier, wanneer er een grotere verscheidenheid aan partijen betrokken zijn, dan is de kans groter dat hier partijen tussen zitten die andere voordelen en risico's ervaren van het delen van informatie en daarom andere eisen hebben (bijv. verschillende soorten bedrijven en overheidsorganisaties).

Om het B2G delen van informatie te ondersteunen in verschillende situaties in complexe multi-actor omgevingen, moet de informatiestroom aangepast kunnen worden aan de situatie. Hoe een informatiestroom eruitziet hangt af van de systemen waar de informatie doorheen gaat en hun verbindingen. De software-architectuur bepaalt voor wat voor informatiestroom een systeem zorgt. Dit betekent dat de architectuur van het systeem zich aan moet passen aan de context om te zorgen voor een juiste informatiestroom in verschillende situaties. We hebben dus een context-aware architectuur nodig. Het probleem waar wij ons op richten in dit onderzoek, is dat er een *gebrek aan kennis is over hoe het ontwerp van context-aware architecturen eruit zou moeten zien die het delen van informatie tussen bedrijven en de overheid ondersteunen in complexe omgevingen*. Ons overeenkomstige doel, is *om een ontwerp te creëren voor context-aware architecturen die het delen van informatie tussen bedrijven en de overheid in complexe omgevingen ondersteunen*.

Er zijn twee opvattingen over 'ontwerp': ontwerp als proces en ontwerp als artefact als gevolg van dat proces. De kwaliteit van het ontwerp-artefact hangt af van de kwaliteit van het ontwerpproces. De kwaliteit van het ontwerp-artefact kan daarnaast informatie verschaffen over de kwaliteit van het ontwerpproces. Dit betekent dat beide opvattingen op ontwerpen van belang zijn om ons doel te bereiken.

In een context-aware architectuur zijn sensoren nodig om de context waar te nemen. Daarnaast zijn er adaptoren nodig zodat de informatiestroom aangepast kan worden aan de context. Ten slotte zijn er contextregels nodig om voor te schrijven welke aanpassing er gedaan moet worden door de adaptoren op basis van de contextinformatie die verkregen is door de sensoren.

Om context-aware architecturen te ontwerpen is inzicht in context tijdens het ontwerpproces nodig om te bepalen welke sensoren, adaptoren en contextregels er in het ontwerp moeten zitten. Echter, in complexe omgevingen zou mogelijk een groot aantal elementen bij de context kunnen horen en relevant kunnen zijn om mee te nemen in het ontwerp. Dit betekent dat twee dingen belangrijk zijn: een systematisch onderzoek naar context en een manier om snel te bepalen wat relevant is voor het ontwerp van de architectuur. De informatie die op deze manier verzameld wordt, moet gebruikt worden om de context te modelleren en om dan de sensoren, adaptoren en contextregels af te leiden van dit model.

Er is te weinig kennis over het modelleren en systematisch onderzoeken van context in complexe omgevingen. De eerste onderzoeksvraag die we moeten beantwoorden is daarom: *“Hoe moet het ontwerpproces eruitzien van context-aware architecturen die het delen van informatie tussen bedrijven en de overheid ondersteunen in complexe omgevingen?”* Om deze vraag te beantwoorden hebben we een ontwerpproces (m.a.w. een methode) ontwikkeld voor het ontwerpen van context-aware systemen in complexe omgevingen. Deze methode is onderdeel van het ontwerpproces voor de context-aware architectuur die we ook hebben ontwikkeld als onderdeel van dit onderzoek. Echter, het kan daarnaast ook onderdeel van zijn van het ontwerpproces voor andere context-aware architecturen en systemen in complexe omgevingen.

Naast sensoren, adaptoren en regels heeft een architectuur voor het B2G delen van informatie ook andere componenten nodig. Meer specifiek zijn dit de componenten die nodig zijn om context informatie op te slaan, om te beslissen welke informatiestroom geschikt is in een situatie en om informatie te delen via die informatiestroom. Deze componenten moeten in verbinding staan met elkaar en met de adaptoren en sensoren om een volledige architectuur voor het B2G delen van informatie te vormen. De tweede onderzoeksvraag die we moeten beantwoorden is daarom: *“Hoe moet een context-aware architectuur eruitzien die het B2G delen van informatie ondersteunt in een complexe omgeving?”*

De methode en de context-aware architectuur voor het B2G delen van informatie zijn twee verschillende artefacten en worden dus ontworpen in hun eigen ontwerpproces. De methode wordt gebruikt om de architectuur te ontwerpen. Daarnaast geeft de evaluatie van de architectuur informatie over de kwaliteit van de methode. Voor beide ontwerpprocessen vertrouwden we op door Peffers e.a. (2007) gespecificeerde activiteiten voor het uitvoeren van design science onderzoek, namelijk 1) identificeren

van het probleem en motivatie, 2) definiëren van de doelstellingen voor een oplossing, 3) ontwerp en ontwikkeling, 4) demonstratie, 5) evaluatie, en 6) communicatie.

Een methode om context-aware systemen te ontwerpen

Om het probleem en de motivatie om het op te lossen te identificeren, hebben we een analyse gemaakt van de problemen die te maken hebben met het bepalen wat tot de relevante context behoort in complexe omgevingen. We hebben hieruit geconcludeerd dat deze omgevingen veel elementen bevatten die mogelijk relevant zijn en waarvoor dit vastgesteld moet worden. Dit is een risico voor de effectiviteit (m.a.w. het niet behalen van doelen) en de efficiëntie (m.a.w. veel inspanning besteden aan het beslissen of elementen relevant zijn) van het ontwerpproces. Om deze risico's te verkleinen, moet de methode voldoen aan het volgende: 1) het ondersteunen van de ontwerper in het systematisch onderzoeken en modelleren van de relevante context voor hun systeem en 2) het ondersteunen van de ontwerper in het afleiden uit het model van context van de sensoren, adaptoren en regels die hun systeem nodig heeft.

Ontwerp van de methode

Om de methode te ontwikkelen, hadden we een definitie van context nodig die gebruikt kan worden om gemakkelijk te beslissen over wat bij de relevante context hoort en om de relevante context te modelleren. Wij konden in de literatuur geen definitie van context vinden die geschikt is als basis voor de methode. Daarom was er een nieuwe, pragmatische definitie van context nodig.

De volgende stap was daarom om een nieuwe definitie van context te ontwikkelen. Het probleem van hoe om te gaan met complexe multi-actor omgevingen in de echte wereld bij het ontwerpen van context-aware systemen vraagt om een praktisch perspectief. Daarom is de definitie van context die wij presenteren in dit onderzoek gebaseerd op een pragmatische filosofie. Volgens onze nieuwe definitie behoort iets tot context wanneer het kan beïnvloeden of de ontwerper zijn of haar doel behaalt.

Wij hebben onze definitie van context gebaseerd op definities van verschillende andere begrippen. *Omgevingselementen* zijn relaties tussen verschillende objecten in de omgeving van een systeem (bijv. welke bedrijven concurrenten zijn). Een *situatie* is een staat van de wereld waarin bepaalde elementen in de omgeving waar zijn en bepaalde relaties bestaan (bijv. bedrijf A is een concurrent van bedrijf B). De *focus* van een context is een omgevingselement dat waar moet zijn om te zorgen dat ontwerpers hun doel bereiken (bijv. de informatiestroom is wettig). Een *contextrelatie* is een relatie tussen een focus en een verzameling omgevingselementen. Deze relatie is zo dat in elke situatie waarin deze omgevingselementen een bepaalde waarde hebben, de focus ook altijd dezelfde waarde heeft (bijv. wanneer bedrijf A en bedrijf B concurrenten zijn en B toegang heeft tot concurrentiegevoelige informatie van A, dan is de informatiestroom onwettig). Een *contextelement van een focus* is een omgevingselement dat een van de omgevingselementen is dat een contextrelatie heeft met de focus (bijv. welke bedrijven concurrenten zijn). Ten slotte bestaat de *context van een focus* uit zijn contextelementen.

De definitie van context hebben wij geformaliseerd door gebruik te maken van de syntax van logisch programmeren. Dit dwingt af dat de definities heel specifiek zijn.

Daarnaast kan het gebruikt worden door de ontwerper om informatie over context uit te drukken tijdens het ontwerpproces, maar ook in het context-aware systeem wanneer deze in werking is.

We hebben deze nieuwe definitie van context gebruikt om de methode voor het ontwerpen van context-aware systemen in complexe omgevingen te ontwikkelen. De methode bestaat uit drie stappen, namelijk 1) inzicht krijgen in context, 2) de componenten bepalen die nodig zijn om context waar te nemen en aan te passen aan context en 3) het vaststellen van de contextregels om met contextinformatie te redeneren.

Stap 1 is verdeeld in drie substappen. In stap 1.1 leiden ontwerpers de focussen voor het systeem af van hun ontwerpdoel. In stap 1.2 verzamelen ze data over welke situaties de focus beperken. Als ze data vinden die suggereert dat iets de waarde van de focus beperkt, dan hebben ze een contextrelatie gevonden. In stap 1.3 analyseren ontwerpers de verzamelde data en leiden zij contextelementen af uit de geïdentificeerde contextrelaties.

Stap 2 is verdeeld in twee substappen. In stap 2.1 bepalen de ontwerpers voor iedere contextrelatie welke contextelementen gemanipuleerd kunnen worden door het context-aware systeem om zich aan te passen en ervoor te zorgen dat de waarde van de focus overeenkomt met het ontwerpdoel. Daarna bepalen ze welke adaptoren het context-aware systeem nodig heeft als een component om deze manipulatie uit te voeren. In stap 2.2 bepalen de ontwerpers voor de overige contextelementen hoe deze waargenomen kunnen worden en welke sensoren hiervoor een component moeten zijn van het context-aware systeem.

In stap 3 worden contextregels afgeleid uit iedere contextrelatie. De body van deze regels drukt een situatie, oftewel een verzameling contextelementen, uit die waargenomen kunnen worden door de sensoren van het context-aware systeem. Als de situatie door de sensoren wordt gedetecteerd, dan moet een adaptor in het systeem een actie uitvoeren om het contextelement in de kop van de regel waar te maken.

Demonstratie en evaluatie van de methode

Het gebruik van de methode werd gedemonstreerd door het te gebruiken om een context-aware architectuur te ontwikkelen voor het B2G delen van informatie in een complexe omgeving, namelijk die van het internationaal containervervoer. Om de praktische uitvoerbaarheid van de methode te evalueren hebben we een case study uitgevoerd met casussen waarin de methode werd gebruikt. De eerste casus is die van de context-aware architectuur voor het B2G delen van informatie. In deze casus gebruikte de auteur van dit proefschrift de methode zelf. Voor deze casus hebben we de effectiviteit van de methode geëvalueerd door het contextmodel te valideren dat we gebouwd hebben met de methode. Voor deze validatie hebben we interviews met experts afgenomen.

Voor de tweede casus heeft een ontwerper die niet betrokken is bij dit onderzoek de methode gebruikt om een gedecentraliseerde datamarktplaats te ontwikkelen. De ontwerper heeft een reflectie op de effectiviteit en de efficiëntie van het ontwerpproces met de methode geschreven die we gebruikt hebben voor de evaluatie. In de derde casus werd de methode gebruikt door weer een andere ontwerper die niet betrokken was bij dit onderzoek om een context-aware systeem te ontwerpen voor stedelijk transport. We

hebben deze ontwerper geïnterviewd om zijn ervaring te bepalen met het toepassen van de methode en het effect van de methode op de efficiëntie en effectiviteit van zijn ontwerpproces in het bijzonder.

Op basis van de evaluatie kunnen we concluderen dat de methode nuttig is voor ontwerpers vanwege de systematische en gestructureerde stappen. De methode hielp om te zorgen voor een effectief en strikt ontwerpproces. Daarnaast verbeterde de efficiëntie in de casussen, omdat met de methode beslissingen om elementen mee te nemen in het ontwerp expliciet en vroeg in het ontwerpproces gemaakt worden. Echter, het kost wel wat inspanning om de methode toe te passen. Daarom moet de methode alleen gebruikt worden voor het ontwerpen van context-aware systemen in complexe omgevingen.

Een context-aware architectuur voor het B2G delen van informatie in internationaal containervervoer

Internationaal containervervoer is een typisch geval van een complexe omgeving en de bijbehorende problemen met het delen van informatie. Door een context-aware architectuur te ontwikkelen voor het B2G delen van informatie in dit domein, lossen we een typisch voorbeeld van het algemene onderzoeksprobleem van dit onderzoek op. Dit draagt bij aan het beantwoorden van onze tweede onderzoeksvraag.

Om het ontwerpprobleem verder te specificeren, hebben we de uitgebreide literatuur bestudeerd die beschikbaar is over het delen van informatie in het domein van internationaal containervervoer. In het domein van internationaal containervervoer is de douane belast met het controleren van de goederenstroom en de naleving door bedrijven van wet- en regelgeving. De douane heeft informatie van hoge kwaliteit nodig van bedrijven voor hun risicobeoordeling en om risicovolle containers voor inspectie te selecteren. De wet vereist dat bedrijven bepaalde informatie delen met de douane. Bedrijven hebben echter meer informatie beschikbaar die ook nuttig zou kunnen zijn voor de douane. We verwachten dat als dergelijke informatie gedeeld wordt met de douane, de kans zal toenemen dat bedrijven die zich niet aan wet- en regelgeving houden zullen worden gepakt. We verwachten dat dit ervoor zal zorgen dat zij zich beter aan wet- en regelgeving houden. Dit kan op zijn beurt een positief effect hebben op de samenleving.

Bedrijven verzamelen informatie van hoge kwaliteit om hun eigen activiteiten op te baseren. Deze informatie zou ook door de douane gebruikt kunnen worden, en wel om een risicobeoordeling uit te voeren. Omdat ze bedrijven echter niet willen verplichten aanvullende informatie met hen te delen, moeten bedrijven deze aanvullende informatie vrijwillig delen. Bovendien moet dergelijke informatie-uitwisseling wettig zijn. Dit is niet altijd het geval, bijvoorbeeld wanneer concurrenten toegang hebben tot elkaars gegevens.

Internationaal containervervoer biedt dus een complexe omgeving voor het delen van informatie. Dit betekent dat in verschillende situaties verschillende informatiestromen nodig zijn om ervoor te zorgen dat bedrijven bereid zijn deel te nemen aan het delen van informatie en om ervoor te zorgen dat het delen van informatie wettig is. Het ondersteunen van het B2G delen van informatie in dit domein vereist een context-aware architectuur.

De doelstellingen van de context-aware architectuur

Om de doelstellingen van een oplossing te definiëren, hebben we stap 1 van de nieuwe methode toegepast. Uit de probleemspecificatie hebben we twee hoofddoelen voor de architectuur afgeleid, namelijk 1) zorgen voor informatiestromen waaraan bedrijven die informatie verstrekken deel willen nemen, en 2) zorgen voor informatiestromen die wettig zijn. In overeenstemming met de methode hebben we twee corresponderende focussen van context afgeleid voor de architectuur, namelijk 1) de bereidheid van bedrijven die informatie verstrekken om deel te nemen aan de informatiestroom die wordt geboden door de architectuur, en 2) de rechtmatigheid van de informatiestroom die wordt geboden door de architectuur.

Om informatie te verzamelen over de bereidheid van bedrijven om deel te nemen aan het delen van informatie hebben we een case study uitgevoerd. De analyse-eenheden bestond uit informatiestromen tussen bedrijven die ook door de douane gebruikt kunnen worden en van bedrijven naar de douane. De casussen waren informatiestromen met daarin architecturen voor informatie-uitwisseling die werden geïmplementeerd in het Cassandra-project en in het CORE-project. We hebben data verzameld over wat impact had op de bereidheid van bedrijven om deel te nemen aan deze informatiestromen op basis van een studie van de documentatie in deze projecten en interviews met onderzoekers en personeel van bedrijven die betrokken zijn bij het Cassandra-project. In totaal hebben we twaalf contextrelaties afgeleid. Elk van deze contextrelaties beschrijft een situatie die van invloed is op de vraag of bedrijven deel willen nemen aan een informatiestroom, en die dus van invloed is op het bereiken van het ontwerpdoel.

Om informatie te verzamelen over de wettigheid van informatiestromen, hebben we juridische experts geïnterviewd met een achtergrond in mededingingsrecht, douanerecht en intellectueel eigendomsrecht. Op basis van deze interviews hebben we acht contextrelaties geïdentificeerd. Elk van deze contextrelaties beschrijft een situatie die van invloed is op de wettigheid van een informatiestroom.

Ontwerp van de context-aware architectuur

Om de architectuur te ontwerpen hebben we de nieuwe methode gebruikt om uit de contextrelaties de benodigde sensoren, adaptoren en de contextregels af te leiden. Daarnaast hebben we bepaald hoe de componenten voor het opslaan van contextinformatie en het nemen van beslissingen over hoe de architectuur zich moet aanpassen aan de context eruit moeten zien. Voor het ontwerpen van dit deel van de architectuur hebben we ons gebaseerd op inzicht in de bereidheid van bedrijven om deel te nemen aan het delen van informatie en de wettigheid hiervan. Dit inzicht we hebben verkregen in eerdere ontwerp cycli in het onderzoek. Vervolgens hebben we alle componenten gecombineerd tot een algehele context-aware architectuur.

In de context-aware architectuur die we hebben ontworpen kan contextinformatie worden verstrekt door bedrijven, onafhankelijke derde partijen en de douane. De bedrijven geven contextinformatie over hun relaties en contracten met andere bedrijven. Bovendien bieden ze informatie over de eigenschappen van data-elementen die via de architectuur worden gedeeld. Daarnaast geven ze informatie over welke gegevens

ze als gevoelig beschouwen en over met wie ze hun gegevens willen delen en onder welke omstandigheden. Dit laatste wordt uitgedrukt met behulp van business-regels.

De onafhankelijke derde partijen bestaan uit twee groepen. De eerste groep bestaat uit identiteitsbeheerders. Deze partijen geven contextinformatie over de identiteit van partijen die betrokken zijn bij het delen van informatie. De tweede groep zijn vertrouwde partijen. Deze bieden contextinformatie die moeilijk te verkrijgen is door bedrijven of waarvoor het veiliger is als een onafhankelijke partij die toevoegt. De douane verstrekt contextinformatie over de verplichtingen die bedrijven hebben om informatie met hen te delen.

De contextinformatie die geleverd wordt door de partijen wordt opgeslagen in een blockchain. Daarnaast worden contextregels en een geschiedenis van toegang tot data door verschillende partijen ook opgeslagen in de blockchain. Deze blockchain wordt gedistribueerd in een netwerk dat alleen bestaat uit gecertificeerde partijen: partijen die de architectuur mogen gebruiken en deelnemen aan het delen van informatie. Het consensusmechanisme is afhankelijk van partijen die controleren of gegevens aan de blockchain zijn toegevoegd door een dergelijke gecertificeerde partij. Daarnaast zijn gegevens die zijn opgeslagen op de blockchain versleuteld.

De beslissingscomponent van de architectuur verkrijgt de contextinformatie uit de blockchain, evenals de contextregels en de toegangsgeschiedenis. Op basis hiervan beslist het wat de juiste informatiestroom is om een verzameling data-elementen te delen. Deze stroom van informatie definieert een reeks systemen waarin de informatie van het ene naar het volgende systeem moet worden gestuurd. De stroom kan systemen van bedrijven en overheidsorganisaties bevatten, maar ook systemen die bepaalde functionaliteit bieden, zoals datapijplijnen. De beslissingscomponent wordt in ons geval dus een informatiestroomplanner genoemd. De informatiestroomplanner slaat de informatiestroom waarvan het beslist dat deze geschikt is op in de blockchain.

De informatie wordt gedeeld volgens de informatiestroom die wordt voorgesteld door de planner door informatierouters te gebruiken. Elk systeem dat is verbonden met de architectuur heeft een informatiestroomplanner en is een knooppunt in het blockchainnetwerk. Om informatie te delen, verkrijgt de router de informatiestroom van de blockchain. Het zoekt op welke informatierouter de volgende is in de informatiestroom. Vervolgens wordt de informatie naar de volgende router in de voorgestelde stroom verzonden en wordt aan de blockchain toegangsgeschiedenis toegevoegd waarin staat dat het systeem van de volgende informatierouter nu toegang heeft tot de informatie. De volgende router stuurt de informatie verder op dezelfde manier.

De architectuur bevat bovendien drie adaptoren, namelijk een encryptiecomponent, een component die de informatiestromen ‘thin’ (dun) maakt en een component waarmee gebruikers alleen informatie kunnen bekijken in plaats van deze op te slaan. Dit zorgt ervoor dat de architectuur de informatiestroom verder aan kan passen door het via een van deze adaptoren te verzenden om het te versleutelen, ‘thin’ te maken of zichtbaar te maken zonder de noodzaak om de informatie op te slaan.

Demonstratie van de context-aware architectuur

Voordat we de architectuur evalueerden, hebben we gedemonstreerd hoe deze in verschillende scenario's kon worden gebruikt. Het eerste scenario laat zien hoe de architectuur kan worden gebruikt om het delen van de summier aangifte bij binnenbrengen (oftewel 'Entry Summary Declaration') te ondersteunen. In het tweede scenario wordt de architectuur gebruikt om een update van de summier aangifte bij binnenbrengen te delen. In het derde scenario wordt een factuur gedeeld met behulp van de context-aware architectuur. Het tweede en derde scenario zijn gegenereerd op basis van een interview met een beleidsadviseur bij de douane waarin we hebben vastgesteld welke informatie voor hen nuttig zou kunnen zijn voor risicobeoordeling.

Evaluatie van de context-aware architectuur

Het doel van de evaluatie was om te bepalen of de architectuur zorgt voor informatiestromen waaraan bedrijven deel willen nemen en die wettig zijn. We hebben de contextregels gevalideerd op basis van een interview met een juridische expert en een interview met een onderzoeker met expertise in het delen van informatie in internationaal containervervoer. Om de gehele architectuur te evalueren verzamelden we gegevens uit drie bronnen: 1) workshops bij Maersk Line, 2) interviews met een expert in formeel recht bij de Nederlandse douane en een expert uit de academische wereld in handelsrecht, en 3) een interview met een expert in IT en bestuur.

Uit de resultaten van de evaluatie blijkt dat de context-aware architectuur voor het B2G delen van informatie nuttig is voor bedrijven en de douane, omdat deze zorgt voor een snellere uitwisseling van informatie en controle op toegang tot informatie. Bovendien past de architectuur zich aan de context aan om ervoor te zorgen dat bedrijven in verschillende situaties willen delen en om ervoor te zorgen dat delen wettig is. Daarnaast hebben we vastgesteld dat er geen duidelijke juridische problemen zijn met de manier waarop de architectuur zorgt voor verschillende informatiestromen (bijv. door contextinformatie op te slaan of een beslissingscomponent te gebruiken). De resultaten suggereren ook dat een belangrijke volgende stap is om te onderzoeken hoe de architectuur op grote schaal in de praktijk kan worden geïmplementeerd en hoe het beheer van de architectuur kan worden geregeld.

Conclusies en implicaties van dit onderzoek

Dit werk heeft drie belangrijke wetenschappelijke bijdragen: 1) een definitie van context, 2) een methode voor het ontwerpen van context-aware systemen, en 3) een context-aware architectuur voor het B2G delen van informatie in internationaal containervervoer. De nieuwe methode voor het ontwerpen van context-aware systemen biedt stappen om de context systematisch te onderzoeken en daaruit een ontwerp af te leiden. Dit kan bijdragen aan de efficiëntie en effectiviteit van het ontwerpproces in complexe omgevingen. De methode is gebaseerd op een pragmatische definitie van context die zeer specifiek is en een conceptueel framework biedt dat zowel tijdens het ontwerpproces als wanneer het systeem in werking is kan worden gebruikt. Bovendien biedt de architectuur voor het B2G delen van informatie in internationaal containervervoer meer inzicht in hoe context-aware architecturen voor informatiedeling eruit kunnen zien. Om context-aware

architecturen voor het B2G delen van informatie voor andere domeinen te ontwikkelen, kan de basisarchitectuur worden gebruikt als een referentiearchitectuur en de nieuwe methode kan worden gebruikt om te bepalen wat de noodzakelijke sensoren, adaptoren en contextregels zijn in het domein.

Door aanvullend B2G delen van informatie te ondersteunen, wil dit onderzoek ook een bijdrage leveren aan de samenleving. Overheidsorganisaties, zoals de douane, kunnen aanvullende informatie van hoge kwaliteit van bedrijven gebruiken om hun risicobeoordeling te verbeteren. Dit kan direct van invloed zijn op de veiligheid. Daarnaast verwachten we dat bedrijven zich beter aan wet- en regelgeving zullen houden, omdat hun risico om gepakt te worden als ze dit niet doen zal toenemen. Dit kan een extra positief effect hebben op de veiligheid.

PART I: INTRODUCTION AND OVERARCHING RESEARCH APPROACH

1 Introduction

Information and information sharing seems to be more important in society than ever before. Supported by Information and Communication Technologies (ICT), such as big data, the Internet of Things (IoT), and blockchain technology, more information has become available that can be shared faster and easier. Not more than a quick look around is necessary to see the benefits that these developments have had on our daily lives. I am typing this on a computer that checks my spelling while I type. It guesses what word I mean when I make a mistake, based on an analysis of data on common mistakes. Next to my laptop is my phone. The phone makes it possible for me to keep an eye on any messages that I may receive and to keep in touch with friends and family. It also stores our published papers, so I can respond fast to full-text requests, even when I leave my office and travel home.

Unfortunately, the same developments have a darker side as well. There are regular news reports about security breaches in which the personal data of a large number of people are compromised. Misinformation is spread via social media and cyberattacks interfere with the processes of organisations. Society is struggling with this dark side of information sharing and there is a tension between the benefits and the risks of information sharing.

In this dissertation, we focus on business-to-government (B2G) information sharing, where such tensions play a role as well. There are some clear benefits and risks of information sharing in this domain. On the one hand, government organisations need information to perform their duties. For example, a customs organisation might use information from businesses to perform risk assessment (Customs Administration of the Netherlands, 2014). Businesses also might benefit from such information sharing, as they might be rewarded for it by the government (Customs Administration of the Netherlands, 2014).

On the other hand, a high variety of businesses can be involved in information sharing for whom information sharing could pose various risks. For example, businesses might be afraid that if their information is not kept confidential, their competitive position might be harmed (Urciuoli, Hintsa, & Ahokas, 2013). This makes businesses only willing to share certain information if they have technical and legal protection. Furthermore, the benefits and risks of information sharing have been weighted by the legislator, resulting in laws and legislation requiring certain data to be shared or protected in certain ways.

Balancing the benefits and risks of information sharing is highly difficult. What the exact benefits and the risks are depends on what information is shared, and how, with whom and when, *inter alia*. For example, if information is competitively sensitive (e.g., pricing information), businesses might consider it a risk to share it via a system that their competitors use as well. Furthermore, this might not be lawful. However, this might not be a problem when information is already public. To complicate matters further, businesses might become competitors over time, or they might only compete on certain products and not on others.

Due to these complexities, information might need to be shared via different information flows in different situations. For example, in case information is competitively sensitive, the risk of sharing it via a system used by competitors might be

considered too high by a business. However, this may be not the case if the information is not competitively sensitive or if the system offers certain security measures to reduce the risks. In that case, the system can be used anyway.

Environments in which information is shared can be highly complex. Such complex environments are characterised by consisting of a high variety of elements with a high variety of relationships and properties, including multiple and different actors that play a role. In such complex multi-actor environments, for example, a high variety of parties with varying interests might be involved in information sharing. Furthermore, the systems that are involved in the information sharing process might vary and have different properties, such as their level of security. Additionally, the data that is shared might be diverse as well.

In complex multi-actor environments, likely more situations occur in which information needs to be shared in different ways. For example, a higher variety of the information that needs to be shared means higher chances that there are different types of information that need to be shared in a different way (e.g., competitively sensitive information and public information). Similarly, when a higher variety of parties are involved, there are higher chances that these include different types of parties that experience different benefits and risks of information sharing and that have different requirements for that reason (e.g., different types of businesses and government organisations).

The new developments in ICT can make environments in which information is shared more complex. For example, the development of IoT introduces a variety of new systems used by a variety of parties that generate heterogeneous data. As these same technologies generate new benefits of information sharing, there is a need to support information sharing in such environments.

In complex environments, it is unlikely that there will be one, or even a few systems that provide an information flow that is suitable in the different situations in which information sharing can be beneficial. Therefore, in this research, we take a different approach. We investigate the possibility of different information sharing systems existing and being used besides each other. The choice of including systems in the information flow is then based on what is most appropriate in the situation in which information is shared.

This approach requires the sensing of context to assess the situation in which information is shared. In addition, it requires adapting the flow of information, such that the appropriate flow of information is provided in the situation. In other words, we need context-awareness.

The flow of information provided by a system is determined by its components, or subsystems and their connections. The flow of information provided by a system thus depends on its architecture. The architecture of the information sharing system needs to adapt in order to adapt the information flow. Therefore, to support information sharing in a variety of situations, we need a context-aware architecture.

The environment in which B2G information sharing takes place often is complex and involves multiple actors. The problem that we address in this research is that *it is not known what the design of context-aware architectures for supporting B2G information*

sharing in complex environments should look like. Our corresponding objective is to create a design of a context-aware architecture for supporting B2G information sharing in complex environments.

Design is a process as well as an artefact that is created during a design process (Walls, Widmeyer, & El-Sawy, 1992). To meet our objective, we have to address both the design process of the context-aware architecture and the context-aware architecture as the resulting design artefact. Namely, the design process affects whether the resulting context-aware architecture (i.e. the design artefact) meets its requirements. On the other hand, an evaluation of the context-aware architecture is needed to assess and improve the design process.

The result of this research is thus two design artefacts. The first design artefact is a method for designing context-aware architectures in complex environments. This method is applied in the design process of the second artefact, viz. a context-aware architecture supporting B2G information sharing in the container-shipping domain. The evaluation of this architecture contributes to the evaluation of the method. The international container-shipping domain is an environment that is particularly complex and thus requires a context-aware architecture. This makes it a suitable domain to develop and evaluate the context-aware architecture in.

In this chapter, to obtain a better understanding of the problem, we discuss the complexities of sharing information in complex environments in section 1.1. Furthermore, we discuss how information needs to be shared more often in such complex environments due to the new developments in ICT. Next, we provide an overview and definitions of the notions that are important for solving the problem we focus on in this research. In section 1.3, we present the problem statement and the objectives of this research. In section 1.4, we describe the relationships between different parts of the research and we explain the structure of this dissertation.

Parts of this chapter have been published in van Engelenburg, Janssen and Klievink (2017), van Engelenburg, Janssen, Klievink, Tan and Rukanova (2018), and van Engelenburg, Janssen and Klievink (2018).

1.1 Information sharing in complex environments

Developments in Information and Communication Technology (ICT) provide new techniques to collect, store, share, and analyse information. In this research, we take a socio-technical perspective on these technologies and their environments. According to Bostrom and Heinen (1977), an organisation consists of a technical system and a social system that are jointly independent, but correlatively interact. The technical systems that are of concern in this research are the technologies that support information sharing. The social systems we are concerned with are government organisations, businesses and society as a whole.

In this section, we first provide a concise overview of the new developments of technologies in ICT and how they affect the benefits and risks of information sharing. Next, we describe how this leads to information sharing in environments that are more often complex. As described at the beginning of this introduction, such highly complex socio-technical environments involving various actors give rise to the need for supporting

different information flows in different situations. Therefore, such environments require a context-aware architecture to support information sharing. The problem addressed in this research is that it is not known what such an architecture should look like.

1.1.1 The impact of big data, IoT and blockchain technology

There are three important recent developments in ICT, viz. big data analytics, Internet of Things (IoT) and blockchain technology. The interest in big data has grown exponentially since 2011 (Ward & Barker, 2013). However, currently, there is no consensus on its definition (M. Chen, Mao, & Liu, 2014; Ward & Barker, 2013). Data has been characterised by volume, velocity and variety and these properties can be used to characterise big data as well (Laney, 2001; McAfee & Brynjolfsson, 2012; Mohanty, 2015; Ward & Barker, 2013). The volume of big data is huge and expanding (McAfee & Brynjolfsson, 2012; Mohanty, 2015). The velocity of big data is that it is created in real-time or near real-time (McAfee & Brynjolfsson, 2012; Mohanty, 2015). Furthermore, big data has a variety of sources, such as social networks, various sensors and smartphones (McAfee & Brynjolfsson, 2012; Mohanty, 2015). The analysis of big data has the goal to extract useful values and to provide suggestions or decisions (M. Chen et al., 2014).

Big data has a profound impact on the benefits and risks of information sharing. There are numerous practical examples of that. For instance, businesses enhance their performance by using big data analytics to understand and predict consumer behaviour (Herschel & Miori, 2017). Government organisations can use it as part of their risk assessment strategy (Customs Administration of the Netherlands, 2014). On the other hand, privacy issues might arise from sharing and analysing large volumes of data, especially when it is combined with other data (Herschel & Miori, 2017). Furthermore, it leads to new power dynamics, with the parties that have access to big data and that can use it yielding a lot of power (Zwitter, 2014).

Data growth is stimulated by the development of the IoT (M. Chen et al., 2014). The idea of the IoT is that the objects that surround us are in one form or another in a network (Gubbi, Buyya, Marusic, & Palaniswami, 2013). The IoT relies on Radio Frequency Identification (RFID), wireless sensors networks, addressing schemes, data storage and analytics, and visualisation (Gubbi et al., 2013). The main application domains of IoT are transportation and logistics, healthcare, smart environments, and personal and social domains (Atzori, Iera, & Morabito, 2010). The main risk of sharing information generated by the devices in IoT are in the area of privacy and security (Ouaddah, Mousannif, Abou, & Ait, 2017; Roman, Zhou, & Lopez, 2013). Security is a challenge for IoT, as there is a lot of interaction between devices that need to be secured (Roman et al., 2013). Furthermore, privacy is a challenge as a lot of privacy-sensitive data can be generated by the devices (Roman et al., 2013).

Blockchain technology was originally conceptualised by Nakamoto (2008) to store and share transactions of the cryptocurrency Bitcoin in a network of blockchain nodes (see section 11.3.1). Blockchain technology has recently gained a lot of attention in the academic world and outside of that for sharing other data than cryptocurrency transactions. Blockchain is investigated to support different processes in the domains of e-government, supply chain management and business process management, for example

(Batubara, Ubacht, & Janssen, 2018; Korpela, Hallikas, & Dahlberg, 2017; López-Pintado, García-Bañuelos, Dumas, & Weber, 2017; Mendling et al., 2017; Ølnes, Ubacht, & Janssen, 2017; Saveen & Monfared, 2016; Schweizer, Schlatt, Urbach, & Fridgen, 2017; Tian, 2016; van der Aalst, De Masellis, Di Francescomarino, & Ghidini, 2017; van Engelenburg, Janssen, & Klievink, 2017a; Weber, Xu, Governatori, Ponomarev, & Mendling, 2016). Examples are tracing goods throughout their lifecycle (Saveen & Monfared, 2016; Tian, 2016), conflict resolution in supply chains (Weber et al., 2016), crowdlending (Schweizer et al., 2017), business-to-government information sharing (van Engelenburg, Janssen, & Klievink, 2017a) and supply chain integration (Korpela et al., 2017).

The literature mentions various benefits of blockchain technology. Mentioned particularly often is that blockchain technology allows for transparency and traceability and thereby provides trust without requiring an intermediary party (Korpela et al., 2017; Mendling et al., 2017; Saveen & Monfared, 2016; Tian, 2016; van der Aalst et al., 2017). Examples of other benefits ascribed to blockchain technology are robustness by decentralisation, practical immutability of stored data, the anonymity of nodes, and high data integrity (Korpela et al., 2017; Mendling et al., 2017; Saveen & Monfared, 2016; Schweizer et al., 2017; Tian, 2016). The disadvantages mentioned in literature are the difficulty of changing data once it is stored in the blockchain, scalability issues, issues with privacy, access control and confidentiality, the unclear legal status of smart contracts, and wasted resources (Mendling et al., 2017; Schweizer et al., 2017; Tian, 2016; van Engelenburg, Janssen, & Klievink, 2018; Vukolić, 2016; Weber et al., 2016).

The claims of the impact that blockchain technology is going to have on society are grand. Blockchain is said to be a revolutionary technology that will have a disruptive impact on society (see e.g., (Swan, 2015)). From a practical point of view, such an impact is not noticeable yet in day-to-day life. It could be the case that blockchain technology is a foundational technology and that it takes time for its effects to become visible (Iansiti & Lakhani, 2017). However, there are some unknowns surrounding blockchain technology and a thorough look into the potential technical and sociotechnical issues of blockchain is necessary to get a more realistic view on what the technology is and is not useful for (Batubara et al., 2018; van Engelenburg, Janssen, & Klievink, 2018). Time will tell whether we are able to overcome these issues and whether there will be a small or great revolution or no revolution at all. However, what is clear is that due to blockchain technology receiving considerable attention, it will affect the way information sharing is viewed within and outside of the academic community. This, in turn, will affect at least the perceived benefits and risks of information sharing.

To illustrate how these new technologies affect the benefits and risks of information sharing, we turn to the domain of B2G information sharing in international container shipping. In this domain, customs is responsible for monitoring the flow of goods (Bharosa et al., 2013). However, containerisation prohibits a direct view of the goods. Furthermore, the volume of goods is so high that it is not feasible to open each container to see what is inside (Levinson, 2010). Fortunately, customs can use big data analytics to determine what kind of containers are high risk and target them for inspection (Customs Administration of the Netherlands, 2014). For this risk assessment, they can

reuse information that businesses involved in the supply chain of the goods that are shipped share with each other (Customs Administration of the Netherlands, 2014; Hesketh, 2010; Hofman, 2011; Rukanova et al., 2011). Information sharing, in this case, could thus have benefits for customs, as it supports them in performing their duties. In addition, it can benefit society, as it supports customs in better protecting safety and security.

Information sharing can be beneficial from the businesses' point of view as well. For different parties in a supply chain, big data can have potential applications, such as forecasting, inventory management, transportation management and human resources (Waller & Fawcett, 2013). The real-time monitoring of almost every link in the supply chain can be made possible using IoT by assigning products a unique identifier and making their data available through the network (Atzori et al., 2010; Whitmore, Agarwal, & Da Xu, 2015). RFID, for example, is currently already used for tracking products in supply chains (Whitmore et al., 2015). Business-to-business (B2B) information sharing can thus be beneficial to businesses. This can be beneficial to customs as well, as they could reuse this information (Rukanova et al., 2011). Furthermore, businesses could be rewarded by customs for sharing their (IoT) data (Customs Administration of the Netherlands, 2014).

However, the level of openness that businesses need to provide to each other and to customs to benefit from information sharing might pose risks for them. In particular, businesses may view autonomous control over data and sharing arrangements as key to their competitive position (Gawer & Cusumano, 2013; Johnston & Vitale, 1988; Tilson, Lyytinen, & Sørensen, 2010; Tiwana, Konsynski, & Bush, 2010). They want control over the information sharing system, as well as use the information sharing system to control access to the data. Opening up data to others means that businesses have to give up some control and autonomy. They may fear that they will be more vulnerable to misuse of the data or to opportunism by others, and that sensitive information is not kept confidential (Fawcett, Osterhaus, Magnan, Brau, & McCarter, 2007; Hart & Saunders, 1997; Klievink, Janssen, & Tan, 2012; Urciuoli et al., 2013; van Stijn et al., 2011). These risks should be carefully balanced against the benefits to meet the requirements of businesses. Otherwise, businesses might not be willing to share their data and there will be no benefits for them, customs or for society. Furthermore, things like unfair competition or harming privacy can be viewed as a risk to society and be unlawful, and this leads to additional requirements.

1.1.2 Information sharing requirements in complex environments
As discussed in the previous section, the new developments in ICT affect the benefits and risks of information sharing. What requirements a system that supports information sharing should meet depends on these benefits and risks. Functional requirements *"describe the behavioral aspects of a system"* ((Anton, 1997) in (Glinz, 2007, p. 1)). When there are different risks and benefits, the behavioural aspects of the system might need to be different.

The variety of situations that occur that lead to different requirements, depends on the complexity of the environment in which the information is shared. A complex

multi-actor environment is characterised by a variety of elements, with a variety of relationships and properties. These elements could include the information shared and the systems and parties involved in the information sharing process. Examples of properties and relationships are the personal nature of information, the level of security of a system, the owner of a system and the competitive relationship between businesses. In a more complex environment, a higher variety of information is shared, by a higher variety of parties, involving a higher variety of systems, *inter alia*. This increases the chances that within these varieties there are different types of data for which there are different benefits and risks of sharing for different parties involved, and so on. This increases the chances that different situations will occur that pose different information sharing requirements to the multiple parties involved.

Big data, IoT and blockchain technology can lead to a more complex environment. Big data, per definition, includes a variety of information from different sources. Furthermore, this information can be used and combined in new ways. In addition, the development of IoT leads to the generation of new types of information by a variety of devices, owned by a variety of parties. In addition, a fundamental feature of blockchain technology is that data is stored by each node in a distributed network, involving additional parties in the information sharing process.

While on the one hand, the development of new technologies in ICT lead to new benefits and risks of information sharing, on the other hand, these technologies make the environments in which information is shared more complex. In these complex environments, more situations occur in which information-sharing requirements are different. To support information sharing in such environments, we need to be able to deal with these varying requirements. A way to do so is by supporting B2G information sharing with a context-aware architecture.

1.2 Definitions

There are some notions that are central to the research in this dissertation. In this research, we investigate context-aware architectures. Providing a clear definition of this notion and related notions makes more clear what exactly we are investigating. Therefore, based on literature, we define the notions of information, metadata, information sharing, business-to-government information sharing, information sharing architecture, and context-aware architecture for the purposes of this research. Of course, context is central to this study as well. However, we found that the literature does not provide a definition of context that is clear enough for our purposes. As part of our research, we provide and formalise such a definition. An analysis of the literature on context and the definition of context we propose can be found in chapter 5.

1.2.1 A definition of information and metadata

According to the Stanford Encyclopaedia of Philosophy (Adriaans, 2013), ‘information’ in everyday use refers to any amount of data, code or text that is stored, sent, received or manipulated. Scientific literature often describes a close relationship of the concept of information with the concepts of data and knowledge. While attempts to come up with

general definitions have been made, there still seems to be some disagreement on the exact meaning of these concepts and the way in which they are related, depending on the context in which they are interpreted or used (Aamodt & Nygård, 1995; Losee, 1997; Zins, 2007). Researchers have, for instance, defined data as symbols that represent properties of objects and events, as patterns with no meaning, and as the measurement or description of facts or states (Aamodt & Nygård, 1995; Ackoff, 1989; Kettinger & Li, 2010). The same researchers define information respectively as data with meaning, as data that is processed to increase its usefulness, and as the joint function of data and knowledge (Aamodt & Nygård, 1995; Ackoff, 1989; Kettinger & Li, 2010).

On the basis of the definitions above, we are mainly interested in looking at information as something that has meaning. If we do not consider the meaning of data, we cannot straightforwardly talk about its usefulness or sensitivity, for example. Considering this, we will use the definition of Aamodt and Nygård (1995) as the definition of information for this research.

Definition 1 (information):

“Information is data with meaning; it is the output from data interpretation as well as the input to, and output from, the knowledge-based process of decision making.” (Aamodt & Nygård, 1995, p. 197)

The notions of data and information are often treated as synonyms in literature. Since our research is based on literature in which this sometimes occurs, we cannot completely avoid using information and data as synonyms ourselves when discussing the literature. We do not expect this to be a problem, but it might be important for the reader to be aware of this.

Metadata is often considered to just be data about data (Burnett, Ng, & Park, 1999; Jeffery, 2000; Schuurman, Deshpande, & Allen, 2008). This ‘simple’ definition, however, is criticised for being too simple and therefore useless (Dempsey & Heery, 1998; Greenberg, 2003). Greenberg (2005) provides a concise overview of alternative definitions that are more descriptive. In addition, Greenberg (2003) provides a definition of metadata that is more precise. However, this definition is quite broad and difficult to understand.

Metadata is relevant to this research, as it can be used to describe the properties of the information that is shared. Information about these properties can be important, as the way in which information should be shared can depend on them, *inter alia*. To develop a context-aware architecture, this metadata should be obtained and reasoned with. When we take into account the role of metadata in this research, the definition of Haslhofer and Klas (2010) seems to fit best.

Definition 2 (metadata):

Metadata is “the sum total of what one can say about any information object at any level of aggregation, in a machine understandable representation.” (Haslhofer & Klas, 2010, p. 4)

1.2.2 A definition of business-to-government information sharing
The exact meaning of the notion of information sharing is often implicit in the scientific literature. Nevertheless, since this notion plays a central role in this research, we do want an explicit definition for our research. Table 1 below shows some of the existing definitions in the literature. We will not discuss each definition in detail, but we will look at some of their elements and use those to build the definition we need for our research.

Definition of information sharing	Source
<i>“the willingness to make strategic and tactical data available to other members of the supply chain”</i>	(Global Logistics Research Team at Michigan State University, 1995) as cited by (Mentzer et al., 2001, p. 8)
<i>“exchanging or otherwise giving other executive agencies access to program information”</i>	(Dawes, 1996, p. 382)
<i>“the act of providing a helpful answer to a request for information”</i>	(Rafaeli & Raban, 2005, p. 63)
<i>“activities that distribute useful information among multiple entities (people, systems, or organisational units) in an open environment”</i>	(Sun & Yen, 2005, p. 422)

Table 1: Definitions of information sharing in literature

All definitions seem to view information sharing as performing some kind of action, with the exception of the Global Logistics Research Team (1995), who view information sharing as willingness to perform an action. It can be observed that according to these definitions information (or data or answers) are made available, distributed, provided or exchanged. Defining information sharing as making information available seems to impose the least restrictions on which actions are covered by the definition. Ultimately, we want information to be made available in this research. There are different ways to do so. In this research, we will support different ways of sharing. A definition that does not limit the actions that fall under information sharing too much fits with this.

Something that also can be observed is that except for Rafaeli and Raban (2005), each of the definitions mentions the entities that the information is shared with explicitly. The entities mentioned in the definitions of Dawes (1996) and the Global Logistics Research Team (1995) are specific for their research domain, which is different from ours. We will therefore use part of the definition of Sun and Yen (2005) to base our definition on. However, in contrast to Sun and Yen (2005), our definition includes information sharing with one entity as well, since there is no apparent reason to exclude this possibility.

Definition 3 (information sharing):

Information sharing is the making available of information to one or multiple other entities (people, systems or organisational units).

Information can be shared in different ways. In this research, it is not only important to look at whether information sharing happens, but also at how it happens. This determines what parties can have access and how secure information is, for example. How information is shared is determined by the flow according to which it is shared. The notion of information flow is thus central to this research and needs to be defined explicitly as well.

Definition 4 (information flow):

A flow of information according to which information is shared is determined by:

- the sequence of systems that the information goes through starting with the party that makes information available and ending with the party to which the information is made available,
- the way in which the information is transferred from one system to the other, and
- the alterations that are made by the systems to the information.

E-government is often defined in terms of the delivering of information or services by the government by electronic means (Z. Fang, 2002; Yildiz, 2007). Whether the sharing of information in the other direction, i.e. by businesses with the government, is or is not part of E-government seems to depend on the definition of E-government (Bharosa et al., 2013; Z. Fang, 2002; Yildiz, 2007). This might have to do with the fact that B2G information sharing has received less attention than Government-to-Citizen (G2C) information sharing (Bharosa et al., 2013).

The little attention that B2G information sharing has received also means that no explicit definition could be found in the literature. Bharosa et al. (2013, p. 9), however, state the following: “*B2G information exchange often concerns the collection of business (usually financial) information by government agencies*”. We do not want to limit the actions that fall under information sharing too much, as we want to support different ways of information sharing with the context-aware architecture. In contrast to Bharosa et al. (2013), we therefore will not talk about the government organisations collecting information from businesses, but about the businesses making information available to the government organisation. B2G information sharing is thus viewed as a type of information sharing in which the parties involved are limited to businesses and government organisations. Based on our definition of information sharing (definition 3, p. 27), we can formulate the definition of B2G information sharing for the purposes of this research as follows:

Definition 5 (business-to-government information sharing):

Business-to-Government (B2G) information sharing is the making available of information by businesses to one or multiple government organisations.

1.2.3 A definition of information sharing architecture

There does not seem to be a commonly agreed-upon definition of ‘architecture’ (Rood, 1994; Software Engineering Standards Committee of the IEEE Computer Society, 2000; Zachmann, 1987). Instead, the word ‘architecture’ is used with different meanings in different domains and the literature mentions several different types of architectures that sometimes are not easy to tell apart at first sight. This can lead to unclearness and confusion. We want to minimise this risk by first discussing some definitions of ‘architecture’ in the literature.

According to Zachmann (1987), in the case of information systems architectures, the notion is relative to what you are doing and this explains difficulties communicating about architectures. There exists no single architecture, but a whole set of architectural representations (Zachmann, 1987). In concordance, the Software Engineering Standards Committee (2000) mentions architectural views, which are representations of the complete system from the perspective of related stakeholder concerns. Independent of the view or perspective, the definitions of architectures in the literature often contain the same elements. The shared elements seem to be that an architecture is the *organisation* of a *system*, this involves the *components* of the system and their *relationships* or *connections* with each other, and sometimes also their relationships with the environment (Garlan & Shaw, 1992; Monroe, Kompanek, Melton, & Garlan, 1997; Perry & Wolf, 1992; Rood, 1994; Software Engineering Standards Committee of the IEEE Computer Society, 2000). Since the definition of the Software Engineering Standards Committee (2000) includes all elements we found common in the literature, we will use their definition for our research.

Definition 6 (software Architecture):

A software architecture is “*the fundamental organisation of a system embodied in its components, their relationships to each other, and to the environment, and the principles guiding its design and evolution*” (Software Engineering Standards Committee of the IEEE Computer Society, 2000, p. 3).

The system mentioned in definition 6 is an information system in the case of this research. In the scientific literature, there is a high variety of definitions of ‘information system’ (for an overview, see e.g., (Alter, 2008; Carvalho, 2000)). However, the definition of Alter (1996) fits well with our research.

Definition 7 (information system):

An information system is a “*system that uses information technology to capture, transmit, store, retrieve, manipulate, or display information used in one or more business processes*” ((Alter, 1996, p. 2) as cited by (Carvalho, 2000, p. 275)).

In this research, the system for which we design an architecture uses information technology to make information available. This can be done by transmitting, retrieving

or displaying the information, for instance, as in the definition above. Since the architecture we develop is that of an information system, it is an information system architecture. To stress the focus on information sharing, we will refer to the architecture as an information sharing architecture as well.

Definition 8 (information system architecture):

An information system architecture is the architecture of an information system. When the main purpose of the system is to share information, then we can refer to its architecture as an information sharing architecture.

1.2.4 A definition of context-aware architecture

Typically, a definition of ‘context-aware architecture’ would be built upon definitions of context and context awareness. However, the literature does not provide a definition of context and context awareness that is clear enough for our research (see section 5.1). In chapter 5, we therefore provide our own definition. In this section, we provide our definition of context awareness, context-aware system, and context-aware architecture. The notion of context in these definitions is the one we present in definition 20 (p. 86).

Schilit and Theimer (1994) were among the first to introduce the term ‘context-aware’. Hong, Suh and Kim (2009) provide an extensive overview of context-aware systems. Their work shows that context awareness involves acquiring, sensing or being aware of context as well as adapting to it or using it. Correspondingly, according to Dey and Abowd (1999), the definitions fall into two categories, namely using context and adapting to context.

A simple definition of ‘context-aware system’ would be that they sense and adapt to context. However, such a definition does not seem to cover what is normally meant by ‘context-aware system’. It seems too broad, as it would cover almost every system.

As an example, consider an electronic tour guide that provides users with information on sights. Users can type in the name of the sight they want information on and then it is provided by the system. According to the simple definition, the keyboard could be viewed as a sensor and providing the information as an adaptation. However, this does not conform to the intuitive meaning of the notion of context-aware system. Now, consider another electronic tour guide that automatically provides information on a sight when the user is near the sight. Such a tour guide would conform to the intuition of what a context-aware system is. However, what is the difference?

At first sight, the difference seems to be that for the system that is not context-aware, the user more directly chooses what they want to see than in the case of the context-aware tour guide. In the latter case, the information is presented to the user without them asking for it. Context-aware systems thus seem to be more autonomous than systems that are not context-aware.

This idea conforms with the work of Baldauf, Dustdar and Rosenberg (2007, p. 1), who state “*context-aware systems are able to adapt their operations to the current context without explicit user intervention*”. This observation also is in agreement with the work of Finkelstein and Savigni (2001). They make a distinction between fixed goals and

requirements. They state that goals are fixed objectives and requirements are volatile and can be influenced by context (Finkelstein & Savigni, 2001). In the case of context-aware systems, the requirements thus depend on context and are in a way conditional. The additional autonomy that these systems have is automatically adapting their operations to these conditions to meet the requirements and meet the system's goal.

If we look at the example of the tour guide again, providing relevant information to the user seems like a goal for both types of tour guides. For the tour guide that is not context-aware, what information is provided to the user depends on the input of the user. The user thus decides what operations they require the system to perform for the system goal (and their goal) to be met. For the context-aware tour guide, what information is provided to the user depends on the location of the user. The system itself thus decides what operations it is required to perform for the system goal to be met. This is where the difference in autonomy is.

Based on this, we can provide our definition of a context-aware system, where 'context' should be interpreted as defined in definition 20 (p. 86).

Definition 9 (context-aware system):

A context-aware system is a system that senses context, then makes an autonomous decision on what operations it is required to perform to meet the system goal based on the context information, and then adapts their operations to meet the system goal.

It could be argued that such context-aware systems do not make autonomous decisions in the strict sense of the word 'autonomous', but that their designer or programmer takes over some of the decision making from the user and designs or programs the system to perform according to these decisions. A discussion of what autonomy means and whether systems really can be autonomous is very interesting, but outside of the scope of this dissertation. We will just remark that autonomy in this case at least has as a consequence that the designer makes additional decisions and translates these into a design. In chapter 6, we discuss in more detail the components of a context-aware system and how it can be designed.

In the literature, there is no clear definition of the notion of 'context-aware architecture'. In fact, the term does not seem to be used that often. When it is, it usually refers to the architecture of a context-aware system (see e.g. (H. Chen, Finin, & Joshi, 2005; Dey, 2000; Schmohl & Baumgarten, 2008)). However, can an architecture itself adapt to context and be context-aware?

We have defined an architecture to be the fundamental organisation of a system (see definition 6, p.29). An organisation of a system can itself not sense context or make decisions. A system, however, can incorporate sensors and a decision component according to its architecture. Yet, adaptation can be viewed as making a change. In our case changing operations to meet the system goal. The organisation of a system could change and thus an architecture can adapt. A change in the organisation of a system would mean that its components or their relationships change. Physically removing components, for example, would of course not be feasible automatically at runtime. However, the

organisation of the system could be changed at runtime by using or not using components associated with certain operations based on the context. In fact, this is exactly what is needed in this research (see section 1.3).

Of course, such a context-aware architecture necessarily would be the architecture of a context-aware system. However, it is a specific type of context-aware system, which adapts to context by changing its organisation. To prohibit confusion, we will refer to this specific type of architecture with the term ‘context-aware architecture’. When we discuss the architecture of context-aware systems in general, we will consistently use the term ‘architecture of the context-aware system’.

Definition 10 (context-aware architecture):

A context-aware architecture is an architecture of a context-aware system that adapts by changing its organisation to only use the required components to perform the required operations to meet the system goal.

1.3 Problem statement, objective and research questions

As we discussed in section 1.1, recent developments in ICT can cause the environment in which information is shared to become more complex. In complex environments, the benefits and risks of information sharing can be different in different situations. This can lead to different information sharing requirements in these different situations.

The notion ‘flow of information’ refers to the way in which information sharing is arranged (see definition 4, p. 28). The flow of information determines what systems the information goes through and who has access. From this perspective, whether information-sharing requirements are met is determined by the information flow. To support information sharing in complex environments, information needs to be shared via different information flows in different situations.

The architecture of a B2G information sharing system determines what components and subsystems are included in the system and how they are connected. What information flows are supported by a system thus depends on its architecture. According to definition 10 (p. 32), a context-aware architecture adapts by changing the organisation of the system. As a result, a context-aware information sharing architecture can adapt the flow of information it provides and it can provide the appropriate flow of information for different situations. The appropriate flow of information, in this case, is one in which the information sharing requirements in the situation are met.

B2G information sharing often takes place in highly complex environments. To support B2G information sharing in such complex environments, we require context-aware B2G information sharing architectures to provide for the appropriate information flow in different situations. In this dissertation, we address the problem that there is no knowledge of what the design of such context-aware architectures should look like. In accordance, we have formulated the problem statement for this research below.

Problem statement:

There is a lack of knowledge on what the design of context-aware architectures that support business-to-government information sharing in complex environments should look like.

The approach to solving the research problem is by providing the design of such context-aware architectures. This is thus our objective.

Objective:

Create a design for context-aware architectures that support business-to-government information sharing in complex environments.

The word ‘design’ can refer to a design process consisting of a sequence of activities, and a design artefact resulting from that process (Hevner, March, Park, & Ram, 2004; Walls et al., 1992). For the former meaning of ‘design’, the world is viewed as acted upon, for the latter meaning the world is sensed (Hevner et al., 2004). The design process is used to produce the design artefact (Hevner et al., 2004). Therefore, the quality of the design artefact will depend on the design process used for creating it. The design artefact is then evaluated to provide information about the problem and improve the design process as well as the design artefact (Hevner et al., 2004). To reach our objective, we should not investigate the design artefact without investigating the design process.

In correspondence with this line of reasoning, we reach our objective when we provide a design process for context-aware B2G information sharing architectures, as well as a context-aware B2G information sharing architecture that is designed using this process. To provide a design process and a design artefact, knowledge is needed on what they should look like.

There is a gap in knowledge of what the design process should look like. The literature describes several design processes that can be used to design artefacts (see section 2.2). However, in this case, we are concerned with the design of context-aware architectures in complex environments. The components that are needed to be incorporated in the architecture to sense context, adapt to context and the context rules according to which the architecture should adapt, depend on what belongs to the relevant context. In complex environments, this is not always obvious as there are many different elements that could be part of the relevant context.

Investigating context systematically to get insight into what belongs to the relevant context as well as a way to translate this insight into a design, therefore is needed to design context-aware architectures in complex environments. Without this, the design process might become inefficient (i.e., spending a lot of effort on deciding if elements are relevant) or ineffective (i.e., not reaching design goals) (see section 3.1). However, there is a gap in knowledge on how to obtain the necessary insight into context and to derive the design from that (see section 3.3). This means that we cannot simply apply existing methods and design a context-aware architecture. Instead, we need to generate new knowledge on what the design process should look like. We therefore need to answer research question 1 below.

Research question 1:

What should the design process of context-aware architectures supporting business-to-government information sharing in complex environments look like?

The answer to this question is a design process. A design process is a sequence of activities that are used to create a design artefact (Hevner et al., 2004; Walls et al., 1992). A method is a set of steps that are used to perform a task (March & Smith, 1995). A design process can thus consist of applying different methods in sequence. The gap of knowledge is on how to get insight into context and on how to derive a design from that. This gap of knowledge does not only exist for the context-aware architectures in a complex environment that we want to develop but more broadly, for context-aware systems in complex environments in general (see chapter 3). To answer research question 1, therefore we develop a method for performing these tasks in the design process for context-aware systems in complex environments. Such a method is a design artefact as well and requires its own design process (March & Smith, 1995).

There is also a gap in knowledge on what the overall design of the context-aware architecture should look like as an artefact (see section 7.4). This overall design should also include the components that are necessary for any context-aware B2G information sharing architecture, such as a context information repository and a decision component, and their connections. In addition, we thus need to answer research question 2.

Research question 2:

What should a context-aware architecture that supports business-to-government information sharing in a complex environment look like?

The answer to research question 2 is a design of a context-aware architecture that supports business-to-government information sharing. For the complex environment, we have chosen the domain of international container shipping. This domain is particularly complex and thus requires a context-aware architecture to support B2G information sharing (see section 7.2). In the design process of this architecture, the new method is applied. The evaluation of the architecture contributes to the evaluation of the method.

1.4 Relationships between the artefacts and parts of this dissertation

In this research, we design a context-aware architecture and a method. The context-aware architecture and the method are different types of design artefacts that are developed for a different purpose. Their design processes will need to be different in some way to accommodate this. Yet, the method is applied in the design process of the architecture and the evaluation of the architecture contributes to evaluating the method. Developing two different, but related artefacts within the same research requires clarity on the decisions that are made about how the research is conducted at different levels. This is

necessary to ensure the consistency of the line of reasoning on which the research is based.

To provide the appropriate level of clarity, we define the different notions that refer to how research is conducted at different levels (e.g., research approach, method). We also describe the relationships between these notions. Then, we make clear at what levels decisions are made for the research as a whole, and at what levels decisions are made for investigating the specific artefacts. We will then present an overview of how the high-level structure of this dissertation follows this line of reasoning.

1.4.1 Overarching approach versus specific methods

The notions ‘research approach’, ‘research methodology’, ‘research method’, and ‘research design’ refer to how research is conducted at different levels. The notions are often used interchangeably or with different meanings. We want to ensure that we base our research on a clear and consistent line of reasoning, in spite of developing different types of artefacts within the same overall research. To ensure this, we need these notions and their relationships to be defined clearly, at least for within the scope of this research.

What is a research approach is often explained by separating it into qualitative, qualitative or mixed methods research approaches (Creswell, 2014; Kothari, 2004). In the domain of information systems, design science is mentioned as a research approach as well (Järvinen, 2007; Peffers et al., 2007). The research approach is thus the type of research.

Quantitative, qualitative, mixed methods and design science approaches have various subtypes. A research design can be viewed as the choice for a subtype of study within a research approach (Creswell, 2014). Kothari (2004, p. 31) defines a research design as “*the conceptual structure within which research is conducted; it constitutes the blueprint for the collection, measurement and analysis of data.*” The research design should include an outline of what the researcher will do and can be viewed as an action plan (Kothari, 2004; Yin, 1994).

The research methodology is the strategy for undertaking research and systematically solving the research problem (Howell, 2012; Kothari, 2004). It consists of the process, methods, and tools used to conduct the research (Nunamaker, Chen, & Purdin, 1991). This notion is very similar to that of ‘research design’ and the work mentioned does not describe what the difference is.

A method is a set of steps that are used to perform a task (March & Smith, 1995). Research methods are the methods or means that researchers use in their research operations (Howell, 2012; Kothari, 2004). Research methods are part of the research methodology (Kothari, 2004). However, the research methodology also includes the logic behind choosing certain methods and an explanation of why this choice was made (Kothari, 2004). The specific research methods translate a research design into practice (Creswell, 2014). Research design and methods thus seem to describe how the research was conducted at a more concrete level than research approach and methodology.

The high-level research approach and methodology is the same for the research conducted for developing the context-aware architecture and for developing the method. Determining where the research efforts for developing the two artefacts overlap is a

strategic effort. Furthermore, the research into both artefacts involves the act of designing something and it is part of a single research project. It would not make sense to have a different theoretical view on how to design artefacts within the same project. Especially, when researching them partially overlaps. Therefore, it makes sense to have the same underlying approach and methodology for both artefacts.

The research design in which the different activities in the design process are specified is more concrete. As the artefacts we design in this research are different, the methods applied in their design processes need to be different. For example, designing and evaluating an architecture or a method requires different types of data and thus different types of methods for gathering data. Therefore, they require a different design process in which different methods are applied. However, these should be derived from the same research approach and methodology.

1.4.2 Parts of this dissertation

In this dissertation, we will use the structure of an overarching research approach and methodology, and specific methods that are applied in a specific design process for each of the two artefacts. To do so, we have separated the dissertation into four parts. In Part I, we present the overall introduction into the research, including a problem statement and an objective for the overall research. In addition, we present the overarching research approach and methodology.

Part II presents the research specific to the development of the method. In chapter 4, we discuss the different steps we performed in its design process, our choice of methods in each step and how they are related. In addition, we discuss the results of the different phases of this process up to and including the design of the method itself. The demonstration and evaluation of the method overlap with the development and evaluation of the architecture and thus are not included in this part. In each section where we present results of a step in the design process, we also specify the way we applied the methods in chapter 4 for that step in the process.

In Part III of this dissertation, we present the design process and the results of the different phases of this process for the context-aware architecture, up to and including its demonstration. As the new method is applied to design the context-aware architecture, Part III also demonstrates the use of the method. Part III of the dissertation is structured similarly to part II. Namely, in chapter 8 we present the design process. In the different sections in part III, we present the results of the different steps in the design process and we discuss how we applied the method in further detail.

Part IV of this dissertation contains the evaluation of the architecture as well as the evaluation of the proposed method, as these overlap. Furthermore, it contains the overall conclusions for the research. The reader that is interested in focusing on certain parts of the dissertation can use table 2 (p. 54) to get an overview of the dissertation.

2 Research approach and methodology

In section 1.4, we discussed that in this research we have an overarching research approach and methodology. We will discuss this research approach and methodology in this section. The design process and the methods applied in this process are presented in chapter 3 for the new method and in chapter 7 for the architecture.

We start by discussing the research philosophy in section 2.1. For our research, we use a design science approach. We discuss this approach in section 2.2. In section 2.3, we present our research strategy and an outline of the dissertation.

2.1 Research philosophy

Research has underlying philosophical assumptions about what research is valid and what methods are appropriate (Myers, 1997). The decision for a research approach and methodology is based on the research philosophy of the researcher and the assumptions related to this philosophy (Creswell, 2014; Holden & Lynch, 2004). Therefore, the assumptions underlying the research should be made explicit (Creswell, 2014; Myers, 1997).

In this section, we discuss our research philosophy and the assumptions that underlie our research. We start with describing and comparing the different paradigms in information systems research in section 2.1.1. In section 2.1.2, we motivate our choice of paradigm for this research.

2.1.1 Paradigms in information systems research

Several paradigms can underlie information systems research. The positivist paradigm is highly dominant (W. Chen & Hirschheim, 2004; Goles & Hirschheim, 2000; Gregg, 2001; Orlikowski & Baroudi, 1991; Vaishnavi, 2007). The interpretive paradigm is often argued to be a valid alternative to the positivist perspective (W. Chen & Hirschheim, 2004; Goldkuhl, 2012; Goles & Hirschheim, 2000; Klein & Myers, 1999; Vaishnavi, 2007; Walsham, 1995b). Some of the earlier work considers the critical paradigm as an alternative as well (Orlikowski & Baroudi, 1991). However, it is often left out in the more recent works on philosophical paradigms in information systems research (e.g. (W. Chen & Hirschheim, 2004; Vaishnavi, 2007)). Instead, in the more recent work, several researchers advocate for using a pragmatist paradigm in information systems research (Goldkuhl, 2012; Goles & Hirschheim, 2000; Marshall, Kelder, & Perry, 2005). In fact, some argue that information systems research often already relies on a pragmatic research philosophy, without making this explicit (Goldkuhl, 2012). In this section, we therefore discuss positivism, interpretivism, and pragmatism as alternative research philosophies in information systems research.

There are different types of philosophical assumptions that can be used to describe and compare the paradigms. Most often mentioned in literature are ontological assumptions about the nature of reality and epistemological assumptions about the nature of knowledge (Burell & Morgan, 1979; W. Chen & Hirschheim, 2004; Hirschheim & Klein, 1989; A. S. Lee & Baskerville, 2003; A. S. Lee & Hubona, 2009; Mingers, 2001; Orlikowski & Baroudi, 1991; Vaishnavi, 2007; Wahyuni, 2012). Furthermore, often the

literature mentions the axiological assumptions about what is of value implicitly or explicitly as well (Vaishnavi, 2007; Wahyuni, 2012). In addition, paradigms are usually compared on a methodological dimension (W. Chen & Hirschheim, 2004; Mingers, 2002; Vaishnavi, 2007; Wahyuni, 2012). What research methods and techniques are deemed to be appropriate for conducting the research depends on the methodological assumptions (Orlikowski & Baroudi, 1991).

In concordance with the information systems research literature, we use these notions to describe and compare the different paradigms. In addition, we discuss their strengths and weaknesses. Based on this, we motivate our choice of a pragmatic paradigm for this research.

2.1.1.1 Positivism

In positivism, the natural sciences are viewed as a model for the social sciences (Gregg, 2001; A. S. Lee & Baskerville, 2003; A. S. Lee & Hubona, 2009; Orlikowski & Baroudi, 1991). Information systems research can be considered positivist if it contains evidence for formal propositions, quantifiable measures of variables, hypothesis testing, or the drawing of inferences about a phenomenon from a representative sample to a population (Orlikowski & Baroudi, 1991).

The ontological position of a positivist is that of realism (Goles & Hirschheim, 2000). Positivists assume that reality consists of immutable objects and structures (Goles & Hirschheim, 2000). These are objectively given, knowable and probabilistic (Burell & Morgan, 1979; W. Chen & Hirschheim, 2004; Goles & Hirschheim, 2000; Gregg, 2001; Myers, 1997; Orlikowski & Baroudi, 1991; Vaishnavi, 2007). They can be described by measurable properties that do not depend on an observer or instrument (Burell & Morgan, 1979; W. Chen & Hirschheim, 2004; Goles & Hirschheim, 2000; Hirschheim & Klein, 1989; Lincoln & Guba, 1985; Myers, 1997; Orlikowski & Baroudi, 1991; Vaishnavi, 2007). Relationships within phenomena are considered to be a priori fixed (Orlikowski & Baroudi, 1991). In the information systems research domain, the unit of analysis in a positivist study often is a single complex socio-technical system (Vaishnavi, 2007).

Epistemologically, scientific knowledge is objective according to positivists and it should allow for verification and falsification (W. Chen & Hirschheim, 2004; Goles & Hirschheim, 2000; Orlikowski & Baroudi, 1991). Positivist researchers assume a one-to-one correspondence between their scientific constructs and objects and structures in reality (Orlikowski & Baroudi, 1991). Research should be aimed at discovering universal or generalizable laws that are independent of context (A. S. Lee & Baskerville, 2003; Lincoln & Guba, 1985; Mingers, 2001; Orlikowski & Baroudi, 1991). In positivist studies, typically theories or hypotheses are generated and empirically tested with the aim to explain and to predict (Goldkuhl, 2012; A. S. Lee & Baskerville, 2003; Myers, 1997; Orlikowski & Baroudi, 1991). Positivist researchers thus search regularities and causal relationships and they aim for generalisability and replicability (W. Chen & Hirschheim, 2004; Gregg, 2001; A. S. Lee & Baskerville, 2003; Orlikowski & Baroudi, 1991). More specifically, they aim for generalisation to different settings (A. S. Lee & Baskerville, 2003).

Axiologically, a positivist researcher values truth (Vaishnavi, 2007). Methodologically, positivist researchers aim to use objective and structured measurements and instrumentation to collect data (Orlikowski & Baroudi, 1991). A positivist researcher is an objective, neutral and value-free observer that does not intervene with the phenomenon under investigation (Goles & Hirschheim, 2000; Gregg, 2001; Orlikowski & Baroudi, 1991; Vaishnavi, 2007). However, they do prefer methods that allow them to control data collection and analysis through control over research design parameters and procedures (Orlikowski & Baroudi, 1991). This means that they can use a variety of research methodologies, such as experimental, quasi-experimental, correlational and causal-comparative methodologies (Gregg, 2001; Orlikowski & Baroudi, 1991). Typically, positivist researchers rely on quantitative and statistical methods (W. Chen & Hirschheim, 2004; Gregg, 2001; Vaishnavi, 2007).

In the natural sciences, the positivist paradigm is quite successful (Goles & Hirschheim, 2000). Its application in other areas of science has enforced criteria for validity, rigour and replicability (Orlikowski & Baroudi, 1991). As the positivist paradigm is dominant in the domain of information systems research, its strengths are often not made explicit. Naturally, the status quo has a lower pressure for justification (Walsham, 1995b). On the other hand, a variety of researchers point out the issues with applying the positivist paradigm to the social sciences and information systems research in particular (Goles & Hirschheim, 2000; Orlikowski & Baroudi, 1991; Walsham, 1995b).

Orlikowski and Baroudi (1991) criticise the positivist paradigm for its aim to find universal laws, as this disregards the impact of historical and social context. Therefore, it cannot provide certain knowledge, such as contextual requirements (Orlikowski & Baroudi, 1991). At the same time, information systems are embedded in such a context and this makes the positivist view incomplete (Orlikowski & Baroudi, 1991). This impacts the development of theories, the understanding of the phenomena and it makes it harder to translate the results of the research into prescriptions for action in practice (Orlikowski & Baroudi, 1991). Gregg (2001) provides similar criticism and states that the positivist paradigm does not take into consideration the software process that makes it possible for technology to be applied. Furthermore, Lee and Baskerville (2003) note that social units are unique and therefore require specified theories instead of universal laws.

The neutral and value-free stance of positivist researchers towards their research requires them to discern fact from value. However, according to Orlikowski and Baroudi (1991), positivist researchers often do not acknowledge that making this distinction is itself a value judgement. In addition, according to Orlikowski and Baroudi (1991), independence between researcher and the phenomenon under study cannot be assumed in the case of social science, as the results of social science enter into the discourse of human reality and thereby transform the nature of these phenomena. In fact, impacting information systems practice can be viewed as the goal of information systems research (Orlikowski & Baroudi, 1991). Instead of independent, the relationship between the researcher and reality is thus reciprocal and reflexive (Orlikowski & Baroudi, 1991). This

leads Orlikowski and Baroudi (1991) to conclude that positivist studies cannot live up to the claim of objectivity and neutrality.

Orlikowski and Baroudi (1991) also criticise the focus on validity and control over research procedures in positivistic research. This makes it difficult for the positivistic researcher to discover and understand the non-deterministic and reciprocal relationships in the domain of information systems (1991). According to Orlikowski and Baroudi (1991), the positivist paradigm is not complex enough and does not provide the sufficient variety in methods to reflect all complexity, ambiguity and instability of the phenomena studied in information systems research.

2.1.1.2 Interpretivism

Interpretivism is based on hermeneutics and phenomenology and knows many different forms (Goldkuhl, 2012; Myers, 1997). A study can be considered interpretivist, when there are no deterministic perspectives imposed by the researcher, participants perspectives are the primary source of understanding, and the phenomena are studied with respect to their context (Walsham, 1995a).

From an ontological perspective, interpretivists have a constructivist view of reality (Goldkuhl, 2012; Orlikowski & Baroudi, 1991). Instead of consisting of a given and fixed constitution of objects, reality is viewed as an emergent social process that is produced and reinforced by humans (Burrell & Morgan, 1979; Orlikowski & Baroudi, 1991). Therefore, reality cannot be understood independently from the researcher and other social actors that construct and make sense of reality (Burrell & Morgan, 1979; Orlikowski & Baroudi, 1991). In interpretivism, there thus is an emphasis on subjective meaning and the processes through which reality is constructed (Burrell & Morgan, 1979; G. Morgan, 1983). Some even view subjective meaning as the objective reality that is the subject matter of an interpretivist study (A. S. Lee & Baskerville, 2003). The elements that are important in the world are relations and cognitive elements (Goldkuhl, 2012).

Epistemologically, an interpretivist aims for understanding phenomena in depth through interpretation (W. Chen & Hirschheim, 2004; Goldkuhl, 2012; Orlikowski & Baroudi, 1991). Where in the natural world a researcher imposes meaning, the social world already has meaning (Goldkuhl, 2012; A. S. Lee & Baskerville, 2003). In interpretivism, scientific knowledge is based on understanding and interpreting the already existing subjective meanings and to reconstruct them and theorise about them (Goldkuhl, 2012; Orlikowski & Baroudi, 1991; Walsham, 1995a). The interpretivist researcher aims at a holistic understanding of the phenomenon under investigation and not only at understanding its parts (Goldkuhl, 2012). In the interpretivist paradigm, there can be different views of the world at the same and interpretations of reality can change over time (Goldkuhl, 2012; Orlikowski & Baroudi, 1991).

The focus of the interpretivist researcher is not so much on discovering universal laws and on generalisation to different settings (A. S. Lee & Baskerville, 2003). Instead, scientific knowledge is considered valid while only pertaining to a specific setting (A. S. Lee & Baskerville, 2003). In addition to the researcher, the subjects researched in an interpretive study are also interpreters and producers of meaningful data (W. Chen & Hirschheim, 2004; Goldkuhl, 2012). They provide facts (i.e. first level constructs) that

are interpreted by the researcher, resulting in theories about the facts (i.e. second level constructs) (A. S. Lee & Baskerville, 2003; Walsham, 1995a). This generalisation from facts to theories is the same as the analytical generalisation described by Yin (1994) (A. S. Lee & Baskerville, 2003). Thus in interpretivist research, generalising within a setting is possible (A. S. Lee & Baskerville, 2003). In addition, it is still possible for an interpretivist researcher to extend their theories to different settings (A. S. Lee & Baskerville, 2003).

Axiologically, what is of value to interpretivists is understanding (Vaishnavi, 2007). Furthermore, they value what is interesting (Goldkuhl, 2012).

Qualitative research often is interpretive research, however, they are not synonymous (Goldkuhl, 2012; Myers, 1997). In interpretivism, data is generated through the interpretation of social constructs such as language and shared meanings (Goldkuhl, 2012; Klein & Myers, 1999). As researchers are not viewed as independent from the phenomenon they study, they cannot be viewed as a neutral and value-free (Orlikowski & Baroudi, 1991). In interpretive studies, constructs, instruments and dependent and independent variables are not defined a priori; they are derived by examination and exposure to the phenomenon under study (Myers, 1997; Orlikowski & Baroudi, 1991). A typical method used in interpretive research is the field study (Goldkuhl, 2012; Orlikowski & Baroudi, 1991).

Where the literature focuses more on the weaknesses of the positivist paradigm than its strengths, it does the opposite for the interpretivist paradigm. As it is not the dominant paradigm in information systems research there is a higher pressure for justification and making explicit the strengths of interpretivism (Walsham, 1995b).

There are different perspectives on the compatibility of interpretivism and positivism (Goldkuhl, 2012; Orlikowski & Baroudi, 1991; Walsham, 1995b). From a weak constructionist perspective in which the researcher merely aims to understand meaning, positivism and interpretivism can be combined (Orlikowski & Baroudi, 1991). Some researchers view interpretivist studies as exploratory and requiring their findings to be subject to a positivistic approach (Orlikowski & Baroudi, 1991; Walsham, 1995b). Others view them as equal and complementary or view the interpretivist paradigm to be able to answer questions that the positivist paradigm is incapable of (W. Chen & Hirschheim, 2004; Orlikowski & Baroudi, 1991; Walsham, 1995b). From a strong constructionist perspective in which the researcher in part creates the reality they study, interpretivism and positivism cannot be integrated (Orlikowski & Baroudi, 1991; Walsham, 1995b).

The strengths of interpretivism seem to mirror the weaknesses of positivism. Interpretivism does not aim for universal laws and does not require a priori establishing of constructs (A. S. Lee & Baskerville, 2003; Myers, 1997; Orlikowski & Baroudi, 1991). Therefore, it can be used to reveal social relations and allows for capturing social phenomena in their context, where this is not possible in positivism (Orlikowski & Baroudi, 1991). Furthermore, the researcher is not assumed to be independent of the phenomenon under study and this thus avoids the problems with that (see section 2.1.1.1) (Orlikowski & Baroudi, 1991).

There have been identified several weaknesses of interpretivism. First of all, interpretivist research often is more time consuming than positivist research (Walsham, 1995a). In addition, based on the work of Fay (1987), Orlikowski and Baroudi (1991) describe 4 deficiencies of interpretivism, viz. not examining conditions giving rise to meaning, not explaining the unintended consequences of actions, not addressing structural conflicts within society and organisations, and not explaining change over time.

2.1.1.3 Pragmatism

Pragmatism is grounded in pluralism (Goles & Hirschheim, 2000). It is bound to neither the natural sciences nor the social sciences as a model (A. S. Lee & Nickerson, 2010). The focus of pragmatism is on actions and changes and on what is useful to bring about a desired change (Goldkuhl, 2012; Goles & Hirschheim, 2000; A. S. Lee & Nickerson, 2010). The work on pragmatism in information systems research is more limited than the work on positivism and interpretivism. However, as noted by Goldkuhl (2012), information systems research is largely based on pragmatism, but this is rarely made explicit.

In pragmatist ontology, reality is objective and external, but grounded in the environment and individual experiences (Goles & Hirschheim, 2000). The ontology of pragmatism is actions and change in a world that is in a constant state of change (Goldkuhl, 2012; Ormerod, 2006). Actions are viewed as a way to change existence and must be guided by purpose and knowledge to bring about the desired changes (Goldkuhl, 2012).

From an epistemological perspective, knowledge is constructive in pragmatism (Goldkuhl, 2012). In pragmatic research, knowledge is constructed to support purposeful change and improvement (Goldkuhl, 2012). Just as for interpretivism, pragmatists do not view knowledge as universal, they view it with respect to a certain setting or context (Marshall et al., 2005). However, in contrast with interpretivism, theories are only considered true for the period of time and the context in which they are considered useful (Marshall et al., 2005). Morgan (2007) argues that complete objectivity or complete subjectivity is impossible in practice. Pragmatism relies on an intersubjective approach instead in which there is a focus on achieving mutual understanding between parties involved in the research (D. L. Morgan, 2007). According to pragmatism, objectivism and subjectivism are not mutually exclusive and pragmatists view the world as consisting of different elements that are objective, subjective or mixed (Feilzer, 2010; Wahyuni, 2012).

In the case of pragmatism, knowledge not only includes explanations and understanding, but, for instance, also prospective, normative, and prescriptive knowledge (Goldkuhl, 2012). These different types of knowledge can be important in the different phases of an action (i.e. pre-assessment, intervention, monitoring and post-assessment) (Goldkuhl, 2012). Design theories can be viewed as a kind of pragmatist theory (Goldkuhl, 2012). Similarly, according to Romme (2003), the pragmatist epistemology underlies design. The aim of a pragmatist researcher is intervention and change (Goldkuhl, 2012). The values of a pragmatic researcher impact what to study and how to

do so (Goles & Hirschheim, 2000). This also allows them to incorporate ethical considerations in the research (A. S. Lee & Nickerson, 2010; Marshall et al., 2005).

From an axiological perspective, according to pragmatists, understanding of reality is imperfect and a view of reality is chosen on basis of whether it is useful and results in a desired change (Goles & Hirschheim, 2000; Rorty, 1996). The emphasis is on 'what difference' believing something in a certain way would make, or what the consequences are of an action (Denzin, 2012; D. L. Morgan, 2007). The worth of a theory or model thus is not only evaluated by its truthfulness, but also and equally by the (ethical) consequences of accepting it and based on whether it is useful (A. S. Lee & Nickerson, 2010; Marshall et al., 2005). Pragmatists do not view knowledge or theories developed by scientific researchers as being worth more than other forms of knowledge, such as subjective knowledge by experts based on their experience, and they study these forms of knowledge as well (A. S. Lee & Nickerson, 2010).

Data is obtained by assessing reality and intervening in it (Goldkuhl, 2012). Methodologically, pragmatism fits well with research approaches that intervene in the world, such as action research and design research (Cole, Purao, Rossi, & Sein, 2005; Goldkuhl, 2012; A. S. Lee & Nickerson, 2010). The usefulness of a theory is established in dialogue with stakeholders and supported by arguments (Marshall et al., 2005). A proposition in pragmatism must be supported by an evaluation of its long-term consequences and its coherence with other theories, beliefs, and ethical implications (Marshall et al., 2005). According to Morgan (2007), the pragmatist approach relies on converting observations into theory and then evaluating the theories by taking action. Just as in the natural sciences, in pragmatism, observations consistent with a theory cannot prove it, but a single observation that is inconsistent is enough to reject it (A. S. Lee & Nickerson, 2010).

The attitude towards methodology in pragmatism is pluralist; research approaches and methods are picked according to their usefulness, considering the purpose of the research, the research questions and the situation in which the research takes place (Bryman, 2006; Goldkuhl, 2012; Goles & Hirschheim, 2000; Wahyuni, 2012). Pragmatists, therefore, can choose to use quantitative or qualitative methods (Goles & Hirschheim, 2000; D. L. Morgan, 2014). Pragmatism particularly often is used as a basis or rationalisation for mixing quantitative and qualitative methods, explicitly or implicitly (Bryman, 2006; Denscombe, 2008; Feilzer, 2010; D. L. Morgan, 2007, 2014). The main type of investigation of the pragmatist is inquiry, i.e., investigating a part of reality to generate knowledge for a controlled change in this part of reality (Goldkuhl, 2012).

According to Marshall et al. (2005), pragmatism is useful in information systems research, as it allows to evaluate knowledge in terms of practical significance and allows for incorporating the ethical acceptability of knowledge to stakeholders in a situation. According to Lee and Nickerson (2010), it is especially useful in information systems research which incorporates design. They mention several ways in which pragmatism offers a 'fresh perspective', compared to positivism. First of all, Lee and Nickerson (2010) state that pragmatism allows for examining the knowledge held by the audience towards which the researcher finds their research relevant. They argue that pragmatism allows for a more broad view of knowledge that can be studied and also that it allows for

studying its consequences and their usefulness and its moral rightness (A. S. Lee & Nickerson, 2010; Ormerod, 2006). According to these authors, pragmatism allows for, but does not restrict researchers to performing applied research, and that it frees them from being bound to both the natural sciences and the social sciences as a starting point (A. S. Lee & Nickerson, 2010). Furthermore, they argue that it is an advantage that it recognises the role of the researcher and their community in the research process (A. S. Lee & Nickerson, 2010).

The literature on pragmatism in information systems research does not provide explicitly and clearly an overview of the weaknesses of pragmatism. This is itself a weakness, as leaving them implicit makes it more difficult to deal with them. Outside of the domain of information system's research, weaknesses of pragmatism are mentioned that are relevant to information systems research as well.

In pragmatism, the focus is on what is useful instead of on what is true. However, it is difficult to define the notion of utility (Feilzer, 2010). Pragmatist researchers are often not precise enough about what usefulness means and for whom pragmatic solutions should be useful (Johnson & Onwuegbuzie, 2004; Mertens, 2003). Furthermore, pragmatism does have difficulty with dealing with propositions that are true, but not useful and vice versa (Johnson & Onwuegbuzie, 2004). If what is useful, or what questions to ask, is determined by consensus amongst peers, such as proposed by Morgan (2007), then this might lead to conservatism (Feilzer, 2010).

According to Sundin and Johannisson (2005), the main criticism of pragmatism is relativism, which might lead to an 'anything goes' attitude. However, they argue that this criticism is not valid, as they believe that solving the ontological issue of whether reality exists is not productive in the social sciences (Sundin & Johannisson, 2005). Instead, they claim that the more important issue is to develop a perspective that is useful for the study at hand (Sundin & Johannisson, 2005).

Denzin (2012) criticises the use of pragmatism as a rationalisation for mixed methods research. Pragmatism relies on determining the consequences and meanings of actions. However, according to Denzin (2012), it does not focus on combining methods. Furthermore, he criticises 'what-works pragmatism' for ignoring the paradigmatic, epistemological and methodological differences between qualitative and quantitative research (Denzin, 2012). He argues that because of this, mixed methods research "*offers few strategies for assessing the interpretive, contextual level of experience where meaning is created*" (Denzin, 2012, p. 83). Similarly, Johnson & Onwuegbuzie (2004) argue that pragmatism fails to provide a solution to some philosophical disputes.

According to Ulrich (2007), pragmatism does not provide guidance on how to ensure that practice is ethical from a methodological point of view and according to Denzin (2012) it does not address social justice issues explicitly. Furthermore, they state that pragmatism is strong in making a difference that matters, but weak in securing methodological rigour (Ulrich, 2007). Johnson and Onwuegbuzie (2004) also argue that pragmatism might lead to incremental change, instead of fundamental change.

2.1.2 Motivation for choosing pragmatism as our research philosophy and dealing with its weaknesses

The research philosophy the research in this dissertation is based upon is pragmatism. In this section, we discuss and motivate our choice for this paradigm. Furthermore, we discuss how we deal with its weaknesses.

2.1.2.1 Motivations for a pragmatist research philosophy

There are two distinct elements in the research in this dissertation. The first is design as a process in the case of the method, or design as the result of that process in the case of both the method and the context-aware architecture. The second is context, as the architecture should sense and adapt to context and the method should support investigating context. Pragmatism was chosen as the research philosophy for this research as it is compatible with and supports design research. Furthermore, it allows us to develop a new view on context that helps to deal better with its complexity.

It is important to note that there is some work on design and research philosophy that view design itself as one of the philosophical paradigms (Vaishnavi, 2007). However, other researchers position design and design science research as an approach that can be based on the philosophical paradigms mentioned in the previous subsections. For example, several papers mention that evaluation in design science research can be based on a positivist, interpretivist, pragmatist or critical paradigm (Hevner & Chatterjee, 2004; Pries-Heje, Baskerville, & Venable, 2008). In addition, Bauer et al. (2014) state that a design process is based on different paradigms in the earlier and later stages of the design process. In this research, we will take the point of view that design science research is an approach that can rely on a pragmatist research philosophy.

Several researchers claim that pragmatism fits well with design research, or that design research is implicitly rooted in pragmatism, as it puts an emphasis on the practical and consequential and allows for a broader notion of knowledge (Cole et al., 2005; Goldkuhl, 2012; Hevner, 2007; A. S. Lee & Nickerson, 2010; Romme, 2003). In our research, we are concerned with the design of a context-aware architecture in a complex socio-technical environment (see section 1.3). To do this, we need knowledge about what the environment looks like. For example, we need to understand what parties are part of the environment and the different laws and regulations that play a role. However, we are not just concerned with what is, such as in positivism or interpretivism. We are concerned with designing or creating something that does not exist yet. Therefore, we need to generate knowledge on what should be and on what is useful. The broader notion of knowledge in pragmatism allows for this.

Both the method and the architecture have a close relationship with the complex environment in which they should be used. For the method, the environment in which it is applied is important, as what the environment looks like affects what the designed artefact should look like. The method should help designers to decide what part of the environment is part of the context that they should take into account for the design of their context-aware system. The method is itself the result of a design process as well, in which the environment in which it is used should be taken into account. This means that it should be ensured that it is useful to designers. For the context-aware architecture, the

environment is important as it should sense and adapt to things in the environment that are relevant. This means that studying or conceptualising the environment is an integral part of investigating the artefacts. As we focus specifically on complex environments in this research, this is perhaps even more important.

According to Bauer et al. (2014), both positivist and phenomenological perspectives exist in the practice of designing context-aware systems, sometimes even within the same project and by the same designer. They suggest that designers often have a phenomenological perspective at the beginning of the design process and later on change to a positivist perspective (Bauer et al., 2014). Toolkits and frameworks for building context-aware applications often require significant specification and therefore fit with the positivist perspective (Bauer et al., 2014). This might make them less useful in the early stages of the design process (Bauer et al., 2014).

As we describe in chapter 3, we found that in the related work, there exist a variety of tools and frameworks from such a positivist perspective. However, there is a lack of knowledge on how to investigate context in the earlier stages of the design process. An investigation of the context is even more important in the case of this research, as it is not clear what belongs to the context that should be taken into account in the design of context-aware systems in complex environments. For this a clear definition of context is required that can be used to make this distinction. Such a definition of context, of course, is related to the ontological and axiological views of the research philosophy chosen. We argue that pragmatism provides for a view of context that fits with what we need for this research.

The positivist account of context is that it is an enumerable set of attributes (Bauer et al., 2014). Correspondingly, the work of Schilit, Adams and Want (1994) and that of Dey (2001) is considered positivist, for example. As we discuss in section 5.1, the problem with these definitions of context is that they are static, while in complex environments what is the relevant context of a context-aware system can be different. Dourish (2004), provides an alternative view that seems to be related to the interpretivist as well as the pragmatist world view. According to Dourish (2004) context is interactional and it consists of relational properties. In his view on context, he establishes a central role for *“the meanings that people find in the world and the meanings of their actions there in terms of the consequences and interpretations of those actions for themselves and for others”* (Dourish, 2004, p. 13).

The view on context of Dourish (2004) is a step into the right direction for what we require, as context is no longer viewed as static, but as part of an ongoing process of interpretation (Lamsfus, Wang, Alzua-Sorzabal, & Xiang, 2015). However, this still does not allow us to decide what does and does not belong to the context of a context-aware system at design time. This is where the difference between the axiology of interpretivism and pragmatism plays a role.

What belongs to the relevant context is hard to determine by looking at what is interesting, as in interpretivism. One could find things interesting for a variety of reasons and this might vary over time. However, looking at what is useful, as in pragmatism, can offer a solution. A way to make a distinction between what is and what is not part of the relevant context to take into account is to determine what is *useful* to take into account,

considering the goal of the designer. In this research, in chapter 5, we provide a new definition of context that is more oriented toward pragmatism than the definition of Dourish (2004), and in which the relevance of context is based on what is considered useful and in which we make usefulness explicit.

2.1.2.2 Dealing with the weaknesses of pragmatism

As discussed in section 2.1.1.3, pragmatism has several weaknesses. In this section, we discuss how we deal with these in this research in an attempt to reduce their effects on the quality of the research.

First, we ascertained that there is a lack of knowledge specifically on the weaknesses of pragmatism in the domain of information systems research. While we cannot completely solve this problem, we did derive the weaknesses of pragmatism from literature in other domains. Criticism of pragmatism often seems to be aimed at its use for rationalising mixed methods research, (e.g., (Denzin, 2012)). However, in this research, we use only qualitative methods. We thus do not use it to defend mixed methods research. Instead, we use it because its ontology, epistemology and axiology fit with our view on context and our focus on design.

An important critique of pragmatism in the literature is that it does not offer much methodological rigour (Ulrich, 2007). To ensure an appropriate level of rigour in this research, we follow a research methodology that is well established in design science research and for which special attention is paid to ensuring rigour. We discuss this approach and methodology in section 2.2. Furthermore, we provide an in-depth description of how we apply this methodology and what methods we use to do so in section 2.3.2.

We further improve rigour by addressing another weakness of pragmatism, namely that pragmatist researchers are often not precise enough about what usefulness means and for whom pragmatic solutions should be useful (Johnson & Onwuegbuzie, 2004; Mertens, 2003). To do so, we can make this explicit. In this research, we are concerned with developing artefacts. These artefacts are developed with one or more goals. We will consider something useful if it contributes to reaching these goals. Of course, reaching the goals should be beneficial, for example to society.

In addition, pragmatism is criticised for not providing guidance on ensuring that practice is ethical from a methodological point of view (Ulrich, 2007). We do not see a clear role for ethics for the method that we propose. A designer or researcher using the method should follow the guidelines for how to conduct ethical research, just as they should when using any other method.

The context-aware architecture supports information sharing. Information sharing can be unethical in numerous ways, for example by harming the right for privacy (The European Parliament and the Council of the European Union, 2016) or by supporting unfair competition practices by businesses. We cannot exclude the possibility that the architecture is used for information sharing that is unethical. However, here it is also the case that usual guidelines for how to conduct ethical research should be followed.

In addition, pragmatism is said to lead to incremental change instead of fundamental change (Johnson & Onwuegbuzie, 2004). However, we fail to see why

incremental change could not lead to fundamental change over time. Furthermore, fundamental, revolutionary change does not always seem to be the better alternative.

The same authors also criticise pragmatism for having difficulty with dealing with propositions that are not true, but useful and with propositions that are useful, but not true (Feilzer, 2010; Johnson & Onwuegbuzie, 2004; Mertens, 2003). In this research, we conceptualise the notion of context. This conceptualisation is based on what is considered useful for the designer to reach their goal. This, in turn, is determined by what affects whether their goal is reached. In this way, it is made explicit what is considered useful. Furthermore, in this research, we require that the truthfulness, as well as the usefulness of our propositions, be established. We view usefulness and truth as two sides of the same coin; we are not interested in that what is true, but not useful. The usefulness of something that is not true might diminish when it is known that it is not true, as it is no longer believed.

The last weakness is that pragmatism does not provide a solution to some philosophical disputes (Johnson & Onwuegbuzie, 2004). We believe that this is not an issue specifically with pragmatism. If they were solved by other paradigms, they would not be disputed anymore. We thus do not consider addressing this as part of our research. In addition, we agree with Sundin and Johannison (2005), that in our area of research, it is more fruitful to determine what perspective of reality is useful than endlessly debating whether reality exists. However, we do concede that this argument seems somewhat circular as it seems to be based on the pragmatic assumption that it is of paramount importance whether something is useful.

2.2 Design science research approach

In his book ‘The sciences of the artificial’ Simon (1996) describes the role of design and the artificial in science. According to Simon (1996), where natural science provides knowledge about natural objects and phenomena, artificial science provides knowledge about artificial objects and phenomena. Instead of only being concerned with what ‘is’, a designer of the artificial is also concerned with attaining a goal and with how things ‘ought to be’ (Simon, 1996).

Design science is a problem-solving paradigm rooted in artificial science (Hevner et al., 2004). At the foundation of design science research is the work of Nunamaker et al. (1991), Walls et al. (1992), March and Smith (1995) and the work of Hevner et al. (2004) (Venable, Pries-Heje, Bunker, & Russo, 2010). Design science research is aimed at creating innovative artefacts that solve problems, achieve a goal, or serve human purposes (Hevner et al., 2004; A. S. Lee, Thomas, & Baskerville, 2015; March & Smith, 1995; Pries-Heje et al., 2008). Correspondingly, design is viewed as problem solving (Wieringa, 2014). In this research, we use a design science approach to contribute to solving the research problem (see section 1.3) by designing a context-aware architecture and a method for designing context-aware systems.

There are two different types of contributions in design science research (Hevner et al., 2004; Winter, 2008). The first type of contribution is providing a view on constructing and evaluating artefacts in general (Cross, 2001; Hevner et al., 2004; Winter, 2008). The second type of contribution is a description of the construction and evaluation

of a specific artefact (Cross, 2001; Hevner et al., 2004; Winter, 2008). The method for designing context-aware systems is of the first type. However, the design of the context-aware architecture is of the second type.

Before we further discuss how we use the design science approach in this research, we need to examine what the nature is of the problem that is solved in design science research and whether this conforms to the nature of our research problem. Furthermore, we need to establish what types of artefacts are studied in design science research and what the type is of the method and of the architecture. We discuss these topics in section 2.2.1. In section 2.2.2, we discuss the view we adopt on theory building.

2.2.1 The Design problem and the artefacts to solve them

Design science research involves the answering of two types of questions (March & Storey, 2008; Wieringa, 2014). The first is a design problem, or a design-based problem solving question (March & Storey, 2008; Wieringa, 2014). Such a design problem calls for a change in the real world and their solution is a design (Wieringa, 2014). The other type is a knowledge question, or a theory-based causal-related question (March & Storey, 2008; Wieringa, 2014). To answer these knowledge questions, knowledge and understanding is required about the world and phenomena as they are, and their answer is a theory or proposition (Hevner et al., 2004; Wieringa, 2014).

According to Wieringa (2014), the research can start with a design problem that gives rise to a knowledge problem. In a similar vein, several researchers describe that the designed artefact relies on kernel theories, which are applied, tested, modified and extended by the researcher (Gregor & Jones, 2007; Hevner et al., 2004; Walls et al., 1992). Our take on this is that a design problem can give rise to both knowledge problems and to new design problems. Our research provides an example of this, as we started out with a design problem of the context-aware architecture and based on this, we identified a lack of a method or the design problem of the method.

The solutions to design problems are evaluated according to their utility and the solutions to knowledge questions are evaluated according to their truth (Hevner et al., 2004; Wieringa, 2014). This conforms to the pragmatic research philosophy that we adopted in this research that evaluates research according to its truth and utility. Just like Hevner (2004) and Cole (2005), we view truth and utility as inseparable.

Considering the central role of utility, we should make clear to whom solutions should be useful. In the field of information systems research, the problem that the design of an artefact should solve is often viewed as an organisational or business IT problem (Hevner et al., 2004; March & Storey, 2008; Venable et al., 2010). This would make the organisation or the business the party to whom a solution should be useful. However, Venable et al. (2010) argue that researchers in the information systems research field should focus more on the not-for-profit human benefit of solutions instead. We wholeheartedly agree with this point of view. The solutions investigated in this research should not only benefit certain organisations or businesses, but ideally should benefit society.

According to Simon (1996), an artefact is something that is artificial and constructed by humans, instead of occurring naturally. In the literature, the types of

artefacts that can be used to solve a problem or reach a goal are categorised in different ways. Based on the work of March and Smith (1995), Hevner et al. (2004, p. 77), define IT artefacts as “*constructs (vocabulary and symbols), models (abstractions and representations), methods (algorithms and practices), and instantiations (implemented and prototype systems)*”. These artefacts can be presented in various forms, such as formal logic and natural language descriptions (Hevner et al., 2004). According to a literature study by Offermann et al. (2010), in practice, artefacts in design science research in the information systems domain can be classified as system design, requirements, methods, algorithms, patterns, guidelines, language or notations and metrics.

Hevner et al. (2004) describe methods as effective development practices. Furthermore, Offermann et al. (2010) describe methods as the activities to support systems development. March and Smith (1995) describe a method as a set of steps that are used to perform a task. According to these authors, the method for designing context-aware systems in complex environments is an artefact of the type ‘method’.

Categorising the context-aware architecture is slightly more difficult. We do not intend to provide a physical implementation of the architecture or test it based on a prototype. It is thus not an instantiation as defined by Hevner et al. (2004). According to Gregor and Jones (2007), such an instantiation represents the principles behind it. We believe that our design encompasses such principles. The context-aware architecture does clearly fall under the category of system design as defined by Offermann et al. (2010). According to these authors, a system design is a description of an IT-related system and can include a description of its architecture.

2.2.2 Theory building

Venable (2006) provides an overview of the work on theories and theory building in design science research. What is interesting to see in his analysis, is that there is a variety of viewpoints on what constitutes a design theory and whether theories should be the output of design science research or even the primary output of such research. In the literature, it is often argued that the building of theories discerns design science from design practice (Venable, 2006). This is sometimes done implicitly by requiring design science to produce new knowledge or theories, instead of only applying existing knowledge (Gregor & Jones, 2007; Hevner et al., 2004; B. Kuechler & Vaishnavi, 2008; March & Smith, 1995) or by stating that in design science research classes of requirements and artefacts should be included in design science (Venable, 2006; Walls et al., 1992). We agree that making a distinction between design practice and design science is important and that this distinction should be that design science should generate knowledge, just like any other approach to science. However, what constitutes knowledge can be different for the design science approach based on a pragmatic paradigm, and for different research approaches based on different paradigms.

Kuechler and Vaishnavi (2008) provide an overview of the views on theory in design science research. Here, we limit ourselves to those relevant to this research. Design theories are often viewed as prescriptive (Gregor & Jones, 2007; B. Kuechler & Vaishnavi, 2008; Markus, Majchrzak, & Gasser, 2002; Walls et al., 1992). According to

Walls et al. (1992), they are about how a design process can be carried out. According to Gregor and Jones (2007), a design theory has several components, viz. 1) purpose and scope, 2) constructs, 3) principles of form and function, 4) artefact mutability, 5) testable propositions, 6) justificatory knowledge (kernel theories), 7) principles of implementation and 8) an expository instantiation. Their broader view of theory in this way also encompasses constructs, models and methods that are viewed as the output of the design science research by Hevner (2004) as part of a design theory (Gregor & Jones, 2007). An instantiation they view as representing the principles behind it (Gregor & Jones, 2007).

Venable (2006), in contrast with Gregor and Jones (2007), Walls et al. (1992) and Markus et al. (2002) does not require design theories to include kernel theories. He argues that kernel theories can be used to form new design theories, but that they are not necessarily part of them. The core is that a solution works, not why it works. We agree with this from a pragmatic perspective; we can establish that a design theory is useful, without requiring knowledge of why it is useful. Furthermore, we agree with Hevner (2007) that it can often be difficult to find kernel theories to base the research on and to modify and extend and that this is not a sufficient reason to reject good research.

In addition, Venable (2006) shares our view that a design method can be an artefact itself that can be designed. They state that design methods themselves are designed according to a ‘meta-design’ and have their own goals. In this research, we also found it more useful to view the method for designing context-aware systems as a separate artefact, than as part of the design theory related to the context-aware architecture. This allowed us to research and design the method according to its own research design process. Furthermore, based on literature, we identified a gap in knowledge not only on how to design context-aware B2G information sharing architectures in complex environments, but also on how to design context-aware systems in such environments in general. Viewing the method as a separate artefact allows us to design and evaluate the method from this more general point of view and thereby providing a more general contribution.

In addition, we do adopt the view of, for example, Markus et al. (2002) and Hevner et al. (2004) that the evaluation of the artefact provides feedback on the method that was used for its design. The method not being part of the design theory of the research into the context-aware architecture does not mean that they cannot be related at all. Therefore, we use the method we propose in this research to design the context-aware architecture. Furthermore, the evaluation of the architecture is part of the evaluation of the method.

2.3 Research methodology

As discussed in the previous subsection, we use a design science approach in this research. Within this approach, we apply the methodology of Peffers et al. (2007). We discuss how we do so in section 2.3.1. In this section, we also provide an outline of the research. In section 2.3.2, we discuss how we ensure rigour and relevance by applying this methodology. The methods we used to perform the activities described by Peffers et al. (2007) are discussed in chapter 4 for the new method and in chapter 8 for the context-aware architecture.

2.3.1 Methodology and outline of the dissertation

Peffers et al. (2007) propose a methodology for performing design science research based on the consensus on the procedure for performing design science research in the previous literature. From this literature, they derive six design science activities, viz. 1) problem identification and motivation, 2) define the objectives of a solution, 3) design and development, 4) demonstration, 5) evaluation, and 6) communication.

The first activity described by Peffers et al. (2007) actually consists of two parts. First, the problem to be solved is defined explicitly (Peffers et al., 2007). In addition, the value of a solution is justified to provide motivation for the research and to improve understanding of the problem (Peffers et al., 2007). The second activity Peffers et al. (2007) describe involves deriving the objectives from the problem specification. The third activity is the creation of the artefact itself (Peffers et al., 2007). This activity is followed by a demonstration of how the artefact can solve one or more instances of the problem (Peffers et al., 2007). For the evaluation, it has to be established how well the artefact provides a solution to the problem and it has to be compared to the objectives (Peffers et al., 2007). The last activity is communicating the research to researchers and other relevant audiences (Peffers et al., 2007).

We perform the research into the method and the context-aware architecture by following the procedure as defined by Peffers et al. (2007). However, the methods used in the design process to perform these activities are different for the method and the architecture, as they are different artefacts. As discussed in section 1.3, the overall research problem that is at the core of this research is that there is a lack of knowledge on what the design of context-aware B2G information sharing architectures in complex environments should look like. To contribute to solving this problem we design two artefacts, namely a method and a context-aware architecture.

For each of these artefacts, we need to define the problem that they solve and justify the value of a solution. Of course, each artefact helps to solve the overall research problem. However, what part of this problem they help solve needs to be specified. Furthermore, the extent to which the artefacts help to solve problems that are more general than the research problem needs to be determined as well. For the method, activity 1 is performed based on an analysis of the related literature in chapter 3. Based on the definition of the problem, we made explicit the objectives of a solution, which we present in the same section.

As part of the problem specification, we did not only find that we require a method for investigating context in complex environments, but that such a method should have a clear definition of context at its basis. We could not find such a definition in the literature, leading to another problem that needed to be solved as well. For the activity of design and development, we thus not only developed a method, but also provided a definition of context that it can rely upon. The definition can be viewed as a construct and thus as an artefact as well. However, as it is very closely related to the method, we do not see the need for defining a separate process for designing it. We assume that if the method meets its objectives, then the definition is useful. The definition of context is provided in chapter 5 and the method is presented in chapter 6.

For the context-aware architecture, we further specify the problem by investigating how the problem manifests in the domain of B2G information sharing in international container shipping. This domain provides a typical example of B2G information sharing in complex environments in which a balance is required between benefits and risks. Furthermore, the ample literature in this domain allows us to investigate and understand the problem in depth. The design problem for the context-aware architecture is discussed in chapter 7.

We use the new method we propose to design the context-aware architecture. More specifically, we use it to define its objectives (activity 2) and to design the architecture (activity 3). The way in which the method is used in activity 2 and activity 3, can thus be used to demonstrate the method. Correspondingly, chapter 7, chapter 9 and chapter 10 present the objectives of the architecture and the part of its design for which the new method is used, as well as a demonstration of the method. In chapter 11, we discuss the additional components of the context-aware architecture and the overall design of the context-aware architecture.

In section 12.2, the architecture is demonstrated using scenarios. The results of evaluating the architecture are presented in chapter 13. The results of evaluating the method are described in chapter 14. The evaluation of the architecture contributes to the evaluation of the method. This is based on the idea that if the method is not effective, then it can be expected that the architecture that was designed using it, is not effective either.

Table 2 provides an overview of the different activities by Peffers et al. (2007) and in which sections their results are discussed in this dissertation.

	Part I: Introduction and overarching research approach	
	1 Introduction	
	2 Research approach and methodology	
	Part II: A method for designing context-aware systems	Part III: A context-aware architecture for B2G information sharing in the container-shipping domain
Activity 1: problem identification and motivation	3 The need for a method	7 B2G information sharing in the international container-shipping domain
Design process	4 Design process for the method for designing context-aware systems	8 Design process for the context-aware architecture
Activity 2: define the objectives for a solution	3 The need for a method	7 B2G information sharing in the international container-shipping domain. 9 The context of B2G information sharing in the container-shipping domain
Activity 3: design and development	5 A definition of context 6 A method for designing context-aware systems	10 Sensors, adaptors and context rules 11 The basic components for context-awareness 12 A context-aware architecture for B2G information sharing
Activity 4: demonstration	7 B2G information sharing in the international container-shipping domain 9 The context of B2G information sharing in the container-shipping domain 10 Sensors, adaptors and context rules	12.2 Demonstration of the architecture
	Part IV: Evaluation, conclusion and future research	
Activity 5: evaluation	13 Evaluation of the architecture 14 Evaluation of the method	
	15 Conclusions	

Table 2: Outline of the dissertation

2.3.2 Ensuring relevance and rigour

As we discussed in the previous section, pragmatism and design science research fit well with each other. However, in concordance with the criticism of Ulrich (2007), Hevner (2007) states that pragmatism alone does not provide for good design science research as it ensures relevance, but not rigour. Hevner (2004, p. 88) state that *“rigor is derived from the effective use of the knowledge base—theoretical foundations and research methodologies.”*

Hevner (2007) describes three cycles within design science research should support relevance an rigour of the research, viz. the relevance cycle, the rigour cycle and the design cycle. In accordance with the work of March and Smith (1995), the design cycle involves the building and evaluating of the artefact. Both the building and the evaluating of the artefact should be based on relevance and rigour (Hevner, 2007).

In the relevance cycle, the research is initiated in its application environment (Hevner, 2007). In this cycle, the problems to address are specified, as well as the acceptance criteria of a solution (Hevner, 2007). The output of the research is evaluated within the application environment in the relevance cycle as well (Hevner, 2007).

In this research, we address the relevance of the research into the architecture by describing and involving relevant stakeholders and domain experts during the building phases as well as the evaluation phases (see chapter 8). For the method, we can ourselves be considered stakeholders, as we will use the method and initially established the need for the method in our own work. We involved additional designers in the evaluation of the method (see chapter 4).

In the rigour cycle, the research uses past knowledge to establish that they are in fact providing a new contribution to the existing research (Hevner, 2007). For both artefacts, we provide an overview of the literature and describe the gap in knowledge that we intend to support filling. For the method, we do so in chapter 3 and for the architecture in chapter 7.

Additionally, rigour involves the selection and application of relevant theories and methods to design and evaluate the artefacts (Hevner, 2007). The design of the context-aware architecture is in part based on existing principles for information sharing in the application environment developed in previous work that we discuss in chapter 7, such as the piggybacking principle. However, concerning the methods, we encountered a problem. The existing work did not provide a method for supporting the systematic investigation of context that we needed to design the architecture. To ensure rigour in spite of this, we developed the new method we present in this research as well.

Ensuring the rigour for developing the method proved a difficult task. As the method is used to investigate context and determine what belongs to context, a conceptual model and a clear definition of context are vital. However, the literature also does not provide a definition of context that is suitable to rely on (see chapter 5). Therefore, we set out to provide such a definition ourselves. To ensure the suitability of the new definition, we formalised it.

The design cycle iterates between the construction of an artefact and its evaluation, which provides feedback to refine the design (Hevner, 2007). This cycle

depends on the other two cycles, as the construction of the artefact, as well as its evaluation, should be relevant and rigorous (Hevner, 2007).

For the context-aware architecture, there were several iterations of the design cycle. This resulted in two designs preceding the design presented in this dissertation. These intermediate results are published in van Engelenburg, Janssen and Klievink (2015) and van Engelenburg, Janssen and Klievink (2017a). For the method, we started with developing a new definition of context, which is published in van Engelenburg, Janssen and Klievink (2017b). The method and a formalised and adapted version of this definition are presented in this dissertation.

Hevner et al. (2004, p. 77) provide guidelines for “*conducting and evaluating good design-science research*”. According to Peffers et al. (2007), their methodology, which is applied in this research, is consistent with these guidelines and other processes described in prior literature.

PART II: A METHOD FOR DESIGNING CONTEXT-AWARE SYSTEMS

3 The need for a method for designing context-aware systems

The aim of this section is to specify the design problem we investigate, and thus to perform activity 1 defined by Peffers et al. (2007) (see table 2). To do so, we first discuss the difficulties of designing context-aware systems in complex environments and the problems associated with this. In section 3.2, we discuss the objectives of the method that follow from this analysis. In addition, we discuss the gap in knowledge that we intend to contribute to filling by reaching those objectives. We do so by showing that the related work does not deal with these difficulties.

Parts of this chapter have been published in van Engelenburg, Janssen and Klievink (2017b) and van Engelenburg, Janssen and Klievink (2019).

3.1 Designing context-aware systems for complex environments

In section 1.1, we stated that the complexity of the environment of systems is determined by the variety of elements in the environment and their properties and relationships. For example, the environment of a system would be considered more complex when there is a higher variety of stakeholders, systems and data involved. We discussed that developments in ICT, namely big data, IoT and blockchain technologies give rise to new benefits and risks of B2G information sharing and that they make the environment more complex.

This is not only the case for B2G information sharing, but it is the case in other domains as well (see section 1.1). The higher variety of data from different sources is an attribute of big data, not of B2G information sharing. Similarly, the use of IoT in different domains leads to the generation of new types of information by a variety of devices owned by a variety of parties. Additionally, the use of blockchain technology leads to including a variety of parties as nodes in any system that relies on it. In these other domains, these developments thus can also lead to new benefits, risks and higher complexity.

An example of a complex environment outside of the domain of B2G information sharing are those involving devices that are used to maintain balance on a smart energy grid (Gubbi et al., 2013). This requires the monitoring of energy points in houses (Gubbi et al., 2013). This contextual information might then be used to coordinate the charging of electronic vehicles (X. Fang, Misra, Xue, & Yang, 2012; Gubbi et al., 2013). Such a system has to function in a highly complex environment. To achieve its purpose, it should interact with a variety of systems that monitor energy consumptions and switch on and off the charging of the vehicles. Furthermore, the system should be designed such that it meets the requirements of a variety of businesses, families and other parties that use and provide energy and information. In addition, it needs to be able to maintain a balance under a variety of circumstances, such as a heatwave, which leads to high-energy consumption.

The benefits and risks resulting from developments such as the IoT and big data thus can lead to higher complexity of the environment in other domains as well. As discussed in section 1.1, to ensure that the systems in such environments meet

requirements in a variety of situations, they need to sense context information and adapt. Additionally, the developments in ICT provide new opportunities to generate context information (e.g., by using IoT devices), use context information (e.g., by using big data analytics), and share and store context information (e.g., using blockchain technology).

The environment in which a context-aware system exists can be viewed as open and infinite, especially when it is complex. At the same time, what belongs to the relevant context influences what the design of a context-aware system should look like (Shishkov & van Sinderen, 2007). This means that the designer of such a system should determine what parts of the environment belong to the relevant context to take into account in the design of their system.

Consider the example of the balancing of the energy grid. In this environment, there is a high variety of elements with different attributes and relationships that could be relevant to the system design. Examples of things in the environment that could be relevant are the privacy preferences of the parties that own the IoT devices, whether these parties are persons, businesses or government organisations, the error margins of the measurements of the devices, temperature fluctuations outside and inside of homes, the location of devices, what safety regulations are applicable, whether IoT data is personal, the format of the IoT data, the weather forecast, and we could go on and on.

The high number of elements in the environment means that it is not obvious what belongs to the relevant context to take into account in the design of the system. Because the environment is highly complex, there is a risk of the design process becoming inefficient because a lot of effort is spent on making decisions on whether elements are relevant. On the other hand, there is a risk of the design process becoming ineffective, because the designer bases these decisions not on a thorough investigation of the context, but on assumptions. This could lead to not reaching design goals.

3.2 Objectives for the method

The high number of elements that have a variety of attributes and relationships that all could possibly belong to the relevant context to take into account when designing a context-aware system, could thus pose the risk of an inefficient or ineffective design process. However, if we have a method to decide easily and systematically whether something belongs to the relevant context, then these risks could be reduced. Furthermore, the part of the design of the context-aware system that depends on the context should be derived from a model of context based on this insight.

According to definition 9 (p. 31), a context-aware architecture interacts with context in three ways: 1) it senses context, 2) it makes decisions on what operations they are required to perform to meet the system goal based on the context information, and 3) it adapts to context. What sensor-components the system requires depends on what it should sense in the context. What rules for making decisions are required depends on the situations in which the systems should perform different operations. Furthermore, what adaptor components the system requires depends on the same thing. The parts of the design of a context-aware system that depend on the context that is taken into account are thus the components for sensing (i.e., sensors) and adapting (i.e., adaptors) and the rules

used for decision making (i.e., context rules). This is thus what the designers should be supported in deriving from their knowledge of context.

Objectives of the method:

1. Supporting the designer in systematically investigating and modelling the relevant context for their system.
2. Supporting the designer in deriving the sensors, adaptors and context rules their system requires from their model of context.

In section 3.3, we show that the relevant literature does not provide a method that meets these objectives and that thus can be used as part of the design process of the context-aware systems in complex environments. In fact, some other researchers also conclude that designers require a way to investigate and model context, but that a way for doing so is not described in the literature.

3.3 Related work on designing context-aware systems

In this section, we present the related literature on designing context-aware systems. We cover the literature on design science research and context awareness, existing guidelines, tools and frameworks, context awareness and requirements engineering, and representing context information and context rules. In our discussion of the literature, we focus on determining whether there already exist methods that can be used to design context-aware systems in complex environments and on whether the literature supports that there is a need for such a method.

3.3.1 Context awareness and design science

In section 2.2, we discussed the design science approach that we use in this research. In that section, we also describe that in this approach, there is an emphasis on the rigour and relevance of design. Especially the work of Hevner (2007) focuses on this and he describes design science as consisting of a design cycle, relevance cycle and rigour cycle. In the rigour cycle, design science activities are connected with scientific foundations, experience and expertise (Hevner, 2007). In the design cycle, there is an iteration between building and evaluating an artefact (Hevner, 2007). In the relevance cycle, a connection is made between the contextual environment and the design activities of the designer (Hevner, 2007).

The connection to the environment made in the relevance cycle is to ensure that the design problem is relevant, that the artefact is usable in practice and that it actually offers a solution to the problem (Hevner, 2007; Hevner et al., 2004). This cycle involves determining the requirements for the system and field testing of the system (Hevner, 2007). The distinctive property of context-aware systems is that they sense and adapt to context to deliver a solution to the design problem. Designing such systems thus involves determining what the system needs to sense, what adaptations it needs to make, and in response to what context information from the sensors (Shishkov & van Sinderen, 2007). We argue that this requires an extension of the activities performed in the relevance cycle.

Context awareness mainly affects the design process in the relevance cycle, as this is where the connection to the environment is made. To design a context-aware system, certain non-functional requirements need to be met. For example, in the domain of context-aware composition of services, these might involve controllability, flexibility and adaptability (Colombo, Mylopoulos, & Spoletini, 2005). However, the difference between systems that are context-aware and those that are not seems to lie mainly in what kind of functional requirements they have.

A context-aware system, like any other artificial artefact, is designed with a goal (Simon, 1996). To reach this goal, a context-aware system might need to provide different functionality in different situations. To illustrate, consider an example of a context-aware B2G information sharing architecture. A design goal for this architecture could be to ensure that sharing is lawful. This goal is the same in different situations. However, the actions performed to reach it might need to be different in different situations. For example, according to European competition law, it is not lawful to share competitively sensitive data with a competitor if it disturbs the competitive positions of the businesses in the relevant market. Whether this is the case is highly dependent on the situation in which the information is shared. For example, sharing the current prices of goods in a supermarket will not disturb the market, as these prices are already public. This might be different in the case of plans for future prices, as this is not public and could disturb the market. If the market is different, for example, the market for logistics services, pricing might not be public and sharing it could disturb the market. To reach the goal of ensuring lawfulness, in the latter situation, the system should perform actions to control access to the data. This is not necessary for the former situation.

In their work on context-aware services, Finkelstein and Savigni (2001) make a distinction between fixed goals and requirements. They state that goals are fixed objectives and requirements are volatile and can be influenced by context (Finkelstein & Savigni, 2001). Due to their dependence on context, the functional requirements in the case of a context-aware system can be viewed as conditional. For example, ‘filtering pricing information from the data’ could be a requirement that needs to be met under the condition that a certain situation happens at runtime, e.g., when sharing it disturbs the relevant market.

This conditionality requires situations to be found in which the functional requirements for the system are different. This insight into context is necessary to establish what adaptors are needed to fulfil the functional requirements in those situations. Furthermore, relating these situations to the functional requirements is necessary to ascertain according to what rules the system should adapt. In addition, it needs to be determined what elements in those situations need to be sensed in order to identify the situation the system is in at runtime.

In the relevance cycle, there is a focus on connecting to the contextual environment to determine the relevance of the problem and determine whether it is solved. This connection is necessary to determine the effects of the artefact on the environment. The additional connection that needs to be made for a context-aware system is to determine what conditions should be included for the conditional requirements. In other words, it needs to be determined what design could deliver the solution to the

problem in different situations. When such a connection is not made, then the context taken into account is based on assumptions of the designer. These assumptions could be checked to some extent in the relevance cycle by determining whether a solution to the problem is found. However, this checking is indirect and thus leads to an inefficient design process. Furthermore, this way of testing makes it very hard, if not impossible, to rule out that parts of the context are included that do not help in solving the problem. This can lead to a design of a system that is needlessly complex.

Peffers et al. (2007) describe several activities that are common in design science research, viz. 1) problem identification and motivation, 2) define the objectives for a solution, 3) design and development, 4) demonstration, 5) evaluation, and 6) communication. To ensure that the problem, objectives and solution are relevant, a connection to the environment needs to be made in the design process during activity 1 and 5. In activity 3, the artefact's desired functionality and its architecture are determined (Peffer et al., 2007). Based on this, the actual artefact is created (Peffer et al., 2007).

The design of context-aware systems should follow the same steps. However, the additional connection to the environment that is necessary for developing context-aware systems should be made in activity 2 and 3, because for context-aware systems the functionality and architecture partially depend on what elements of the environment are parts of the relevant context. For functionality, insight into context is needed to determine what the system needs to be able to sense, what adaptations it needs to be able to make and what situations should lead to what adaptations. The architecture should have the necessary sensors and adaptors to be able to offer this functionality.

3.3.2 Context awareness and requirements engineering

Nuseibeh and Easterbrook (2000) describe the requirements engineering process as consisting of context and groundwork, eliciting requirements, modelling and analysing requirements, and communicating requirements. The stage of context and groundwork is viewed as preparation and is used to determine the feasibility of the project and to select methods for further development (Nuseibeh & Easterbrook, 2000). It thus does not involve the systematic investigation of context that we need. The stage of requirements elicitation involves identifying stakeholders and goals (Nuseibeh & Easterbrook, 2000). This also does not involve linking situations to functional requirements.

The work specifically on requirements engineering in light of context awareness is limited. The notion of context-aware systems is often not mentioned explicitly in this work. Instead, the work discusses, for example, context-aware services, dynamic adaptive systems or self-adaptive systems (Berry, Cheng, & Zhang, 2005; Finkelstein & Savigni, 2001; Sawyer, Bencomo, Whittle, Letie, & Finkelstein, 2010). This literature does, however, acknowledge the importance and different type of relationship between context and requirements in the case of context awareness (Berry et al., 2005; Finkelstein & Savigni, 2001; Sitou & Spanfeller, 2007). In some cases, it even goes as far as viewing adaptation as requirements engineering by the system itself or as calling for requirement awareness by the system (Berry et al., 2005; Sawyer et al., 2010).

Even though the existing work appreciates the complexity of requirements engineering in the case of context awareness and the need to have insight into context, it

does not provide a way to get this insight systematically. Instead, it proposes the use of existing usability methods or interviews, for instance (Kolos-Mazuryk, Poulisse, & van Eck, 2005; Sitou & Spanfelner, 2007). Of course, such techniques might be useful in gathering the appropriate data necessary for getting insight into context. Nevertheless, these methods do not provide the structure necessary to investigate context in the case of complex multi-actor environments. More specifically, they do not provide a way to decide easily on what belongs to the context.

3.3.3 Guidelines, tools and frameworks for designing context-aware systems

In 2009, Hong, Suh and Kim (2009) found that about 5.5% of the papers on context-aware systems provide guidelines for development. A portion of this work and more recent similar work focuses on solving issues in a specific application domain or with specific types of sensors (see e.g. (Bolchini, Schreiber, & Tanca, 2007; Casas, Cuartielles, Marco, Gracia, & Falcó, 2007)). The majority of the work is on providing technical tools, frameworks and infrastructures that a designer can use to build context-aware applications (see e.g. (Anhalt et al., 2001; Augustin et al., 2006; J. Hong & Landay, 2001; Qiu, Chang, Lin, & Shi, 2007; Salber, Dey, & Abowd, 1999; van Sinderen, van Halteren, Wegdam, Meeuwissen, & Eertink, 2006; Wei, Farkas, Prehofer, Mendes, & Plattner, 2006)). In this ‘infrastructure-centred approach’ there is an assumption that the complexity of developing the systems can be reduced by using an infrastructure that can gather, manage and distribute context information (Henricksen & Indulska, 2006). These tools and frameworks for designing context-aware systems can be quite useful to designers, helping them to elaborate on the technical details and create the system. However, this assumes that the context is known.

In a more recent survey, Alegre, Augusto and Clark (2016) provide an overview of methods for engineering context-aware systems. An analysis of the described methods shows that they do not include the investigation of context as an explicit and fundamental stage in the design process. In concordance, Alegre, Augusto and Clark (2016) conclude that the work does not include techniques and tools for understanding the context.

The research by Alegre, Augusto and Clark (2016) does show that insight into context is important. On the basis of the results of a questionnaire completed by 750 researchers, they determine that among the most important features of a method for developing context-aware systems is the ability to represent situations in which the system should adapt in order to better understand them (Alegre et al., 2016). On the basis of an analysis of the literature, they state the following: *“All context information modelling and reasoning techniques need to enable the situation representation, but there is no support for understanding the situations and the contexts that they are going to be represented, stemming from the requirements.”* (Alegre et al., 2016, p. 24). Our method, in which getting insight into context is a fundamental stage in the design process, thus addresses a problem that is important to researchers involved in the design of context-aware systems.

3.3.4 Conceptual models and representing context

Investigating context could rely on a clear conceptual model. Such a model could help designers to determine what to look for in the environment and could provide clarity on what is and is not relevant. Furthermore, systematically investigating context could be supported by having a clear and systematic way to express the results. It is therefore relevant to look at the related work on conceptual models for context and representing context information and rules for adapting as well.

The area of context representation is very extensive and the literature already contains several overviews (e.g. (Bettini et al., 2010; Perttunen, Riekki, & Lassila, 2009; Sagaya Priya & Kalpana, 2016; Strang & Linnhoff-Popien, 2004)). Here, we will not provide a redundant overview, but we will discuss the extent to which the existing work on this can aid designers in investigating context. The main difference between our work and the existing work is in the way the representations and rules for reasoning are established. In our case, these follow directly from an investigation into the context of a context-aware system.

To explain the relationship with the existing works on representation and reasoning, we start with Winograd (2001, p. 417), who states the following: *“The hard part of this design will be the conceptual structure, not the encoding. Once we understand what needs to be encoded, it is relatively straightforward to put it into data structures, data bases, etc.”* Winograd (2001) stresses the importance of having a conceptual model of context. This requires insight into the nature of context. However, there is currently no shared understanding of context (Alegre et al., 2016). This lack of a shared understanding is associated with imprecise definitions of context in literature and a lack of consensus on context definitions.

Winograd (2001) notes that encoding or representing context itself is not the hard part. Yet, Perttunen, Riekki and Lassila (2009) note that conceptual models have received little attention in the literature, compared to context representation. There is even less work on determining what context and rules should be represented in a specific context-aware system. The research that does focus on this issue covers only a specific domain, such as m-commerce (Benou & Vassilakis, 2010) or web engineering (Kaltz, Ziegler, & Lohmann, 2005). This work is thus of limited application in other domains.

Shishkov and van Sinderen (2007) are an exception in the sense that they do provide support to investigate the context. In the method that they propose, first, the occurrence probability of different context states are determined to establish what the default behaviour of the system should be. Then they determine what parameters need to be observed to recognise these states. While their method is highly useful in many cases, it does require obtaining information to measure the occurrence probability of the context states. This might not always be possible. Furthermore, it does not provide support on how to identify the context states and the parameters. While this might be evident in many environments, in highly complex environments that contain a high variety of possible context states and parameters, this might be more difficult to establish.

The work of Crowley (2003) focuses on a specific domain as well, viz. observing human behaviour. However, he mentions some ideas that we will adapt and extend. Crowley (2003) states that designers should only include entities and relationships that

are relevant to a system task to prevent the system from becoming very complex. The relevant entities and relationships are selected by *“first specifying the system output state, then for each state specifying the situations, and for each situation specifying the entities and relations”* (2003, p. 112). The relations that he refers to are the properties of the entities. They are closer to what we define as context elements than to the context relationships that describe the impact of context in our method. Determining the context relationships would be akin to determining what situation should be specified for an output state in the work of Crowley (2003). In contrast to our work, Crowley (2003) does not provide explicit guidance on how to investigate this and on how to express context relationships as rules with which the system can reason.

4 Design process for the method for designing context-aware systems

As discussed in section 2.3, the design process of the method relies on the steps for designing artefacts defined by Peffers et al. (2007), viz. 1) problem identification and motivation, 2) define the objectives for a solution, 3) design and development, 4) demonstration, 5) evaluation, and 6) communication. In this section, we describe our choice of methods to perform these activities and their relationships. In the sections where the results of each of the steps are presented (for an overview see table 2, p. 54), we discuss in detail how exactly we applied the methods for that step.

Parts of this chapter have been published in van Engelenburg, Janssen and Klievink (2017b) and van Engelenburg, Janssen and Klievink (2019).

4.1 Activity 1 and 2: Problem identification, motivation and objectives of a solution

We identified the problem of a lack of a method for designing context-aware systems in complex environments in practice when designing the context-aware architecture. When developing the context-aware architecture, we found a high variety of elements in the environment that could be taken into account in the design, but for which it was not immediately apparent whether they should be taken into account. When investigating the literature for a systematic approach for dealing with such a complex environment, we could not find anything suitable.

To establish that this is not just a problem for designing this specific context-aware architecture, we made an analysis of what was the root of the problem. We concluded that this was the complexity of the environment and not the type of artefact. For further support, we illustrated our line of reasoning with an example outside the domain of B2G information sharing. The result of this analysis is described in section 3.1.

According to Peffers et al. (2007), the objectives should be inferred rationally from the problem specification. To derive the objectives for the method, we analysed the problem. The objectives for the method are presented in section 3.2.

Next, we analysed the literature to determine whether there are existing solutions to this problem or other methods that meet the objectives. Furthermore, we used the literature to determine whether other authors believe that a solution to this problem is important as well. We established that the work on design science is relevant, as it provides guidelines on how to ensure that the research is relevant and thus it prescribes making a connection to the environment. We found that the work on requirements engineering for context-aware systems is relevant as well, as we expect the environment or context to affect what the requirements for a system are. Furthermore, the work on existing guidelines, tools and frameworks for developing context-aware systems is relevant, as they might include guidelines for investigating context as well. In addition, systematically investigating context could be made much easier if there is a clear conceptual model to rely on, making the research on conceptual models and representation of context relevant as well. The results of analysing the related work are presented in section 3.3.

4.2 Activity 3: Design and development

The design and development of the method were performed in two steps. First, we provided and formalised a definition of context. Based on this definition, we developed the method.

4.2.1 Defining context

In section 3.3.4, we already discussed that a conceptual model might help with systematically investigating context and representing the gained insight, but that these models have received little attention in the literature. A definition of context could provide for such a model. Furthermore, such a definition of context is necessary to be able to decide whether something in the environment is context.

We searched the literature for existing definitions of context and established whether they support easy decision making on what belongs to context and provide a model of context that is suitable for use in the new method. As the literature could not provide a definition that was suitable, we set out to develop a new definition of context.

In this research, we adopt the pragmatic paradigm as our research philosophy (see section 2.1). According to this paradigm, there is an objective reality but understanding this reality is imperfect and a view on reality is chosen based on what is useful (Goles & Hirschheim, 2000). Knowledge is thus something that is constructed and valued for its usefulness (Goldkuhl, 2012).

In correspondence with the pragmatic paradigm, we view defining context as a constructive effort. This means that for developing the definition, we have to rely to some extent on creativity and trial and error (Verschuren & Hartog, 2005). This is the same as for other things designed, such as the method or the context-aware architecture.

Of course, it is important not to rely merely on creativity. We established that the definition should be the basis of a method that meets the objectives in section 3.2. This guides the development of the definition. We also formalise the definition of context. One of the reasons for doing so is to enforce a certain level of preciseness. The new definition of context is presented in chapter 5.

There is no explicit evaluation of the definition of context. We presume that if the method that is based on the definition of context reaches its objectives, then the definition that it is based upon is a suitable basis. In this way, the evaluation of the method can be viewed as evaluating the definition as well.

4.2.2 Developing the method for designing context-aware systems in complex environments

In this section, we discuss how we developed the method for designing context-aware systems in complex environments. In section 3.2, we present the objectives of the method, which are: 1) supporting the designer in systematically investigating and modelling the relevant context for their system and 2) supporting the designer in deriving the sensors, adaptors and context rules their system requires from their model of context. To meet these objectives, the method should support designers to perform the following steps: 1)

getting insight into context, 2) determining the components necessary to sense and adapt to context, and 3) determining the rules for reasoning with context information.

The crucial step is of course step 1, as the other steps rely on this. However, as we discussed in chapter 3, the literature did not provide starting points for investigating context systematically, with the exception of the work of Crowley (2003). According to Crowley (2003, p. 112), the entities that are relevant can be selected by specifying the system output state and then specifying the situations and then specifying the entities and relations for each situation. As we stated in section 3.2, the work of Crowley (2003) is not enough to fully rely on for our method. For example, it is not explained how the situations are selected. Nevertheless, for our method, we can follow this structure starting with an overall goal for the system and ending with context elements. We only need to fill in how to perform each of the sub-steps. For this, we can rely on the definition of context and its formalisation and make the designer build a model of the relevant context step-by-step. The results of the design efforts for the first step are presented in section 6.1.

The knowledge on context gained by the designer in step 1, should allow for deriving the components for adapting and sensing context in step 2. If this knowledge in step 1 is structured well, based on the definition of context, this should not be a difficult exercise; if we know what the relevant context is, we know what to sense and adapt to. Still, this derivation should be performed systematically as well. The results of the design efforts for the second step are presented in section 6.2.

For the last step, we need to describe how the knowledge of context should be translated into rules that the system can use to determine how to adapt in different situations. This should become an easy exercise as well, after step 1. Especially, since the formalisation of the definition of context allows for expressing knowledge of context in a format that can be used by a logic program. The results of the design efforts for the third step are presented in section 6.3.

4.3 Activity 4: Demonstration

The aim of this activity is to show how the method can be used to solve an instance of the problem (Peffer et al., 2007). We used the method to design a context-aware B2G information sharing architecture. While this is an architecture and not a system, it requires establishing the necessary adaptors and sensors that should be included. Furthermore, as in this case the architecture itself is context-aware (see definition 10, p. 32) we established the rules according to which the architecture should adapt as well. Therefore, we performed all the steps in the method and the design of the context-aware architecture thus provides a demonstration of how the method can be used. The process of designing the architecture, including the use of the method, is described in chapter 8. The results of using the method are presented in sections 9 and 10.

4.4 Activity 5: Evaluation

Evaluating the method is a highly difficult, albeit important task. The method was designed aiming to meet the following objectives (see section 3.2): 1) supporting the

designer in systematically investigating and modelling the relevant context for their system and 2) supporting the designer in deriving the sensors, adaptors and context rules their system requires from their model of context. The method describes the steps designers can use for systematically investigating and modelling context, as well as deriving the sensors, adaptors and rules from insight into context. By describing these steps, we aim to provide the support to designers that they need to meet these objectives. If the method succeeds in doing so, we expect a design process that is considered efficient and effective by designers in spite of the complex environment and thus that we solved the research problem (see section 3.1). To evaluate the method, we thus should determine whether designers consider the design process when using it in complex environments efficient and effective.

Evaluation of an artefact is often performed by performing measurements when the artefact is used and when it is not used (Verschuren & Hartog, 2005). The usefulness of such a comparison would be questionable in our case, as many other variables influence the design process that cannot be controlled (e.g., properties of the designers applying the method and of the system to be designed). Gathering quantitative data is not feasible, as designing systems often is quite laborious. Gathering existing data on context-aware systems in complex environments that already have been developed without the method also is not possible, as the developments in ICT that gave rise to the complex environments are recent.

This means that we have to look into other possibilities. Yang and Padmanabhan (2005) categorise *ex post* evaluation methods according to the dimensions of setting and computation of quality measures. Considering the setting, a system can either be implemented in a real situation or not (Y. Yang & Padmanabhan, 2005). Analogously, we have to choose whether we evaluate the use of the method in a real situation or not. Concerning the computation of quality measures, there is the option of automatically computing them from data, or obtaining input from human subjects (Y. Yang & Padmanabhan, 2005).

In our case, it is preferable to observe the use of the method in practice or to perform a naturalistic evaluation (Venable, 2006). Using the method in a purely experimental setting would not provide the level of complexity of using the method in practice, for example. This would be a problem because the method was developed specifically to deal with such high levels of complexity. It is also feasible to use the method in practice. To do so, a designer should use the method to design a context-aware system in a complex environment.

According to Pries-Heje et al. (2008) the evaluation of a process, such as a method, can rely on the idea that a good process will lead to a good product. In our case, the product is a context model on which then the design of a context-aware system is based. Effectiveness of the method then can be viewed as the extent to which the system designed using the method meets its requirements (Pries-Heje et al., 2008).

It would interfere quite a lot with the efficiency of the design process, if a designer had to keep track of all the different elements they considered as a candidate to be part of the context at some time (maybe even only in their mind) during the design process. The effectiveness of the context-aware system does provide an indication of

whether the method is effective. However, it does not provide information on whether the effectiveness of the system is due to the use of the method. This makes effectiveness and efficiency hard to measure automatically. Therefore, for the computation of quality measures, we have to rely on human input.

In the end, the method should be useful. The parties that the method should be useful to, by supporting an efficient and effective design process, are primarily designers. This makes their experience with using the method relevant to the evaluation of the method. More specifically, this makes their views on whether the method supports an efficient and effective design process in complex environments relevant. The view of the usefulness of a method by a designer is subjective. However, subjective experience can be viewed as a form of knowledge as well (A. S. Lee & Nickerson, 2010). In our case, this is relevant knowledge. Furthermore, it is knowledge that is attainable.

We use a case study to evaluate the method. To determine whether a case study is an appropriate method, we first need to establish what we mean by ‘case study’. In the case study method, a contemporary phenomenon is investigated in depth in its natural context (Benbasat, Goldstein, & Mead, 1987; Kothari, 2004; Yin, 1994). The boundaries of the phenomenon and its context are often not clear at the start of the research (Benbasat et al., 1987; Yin, 1994). In case studies, often data is collected on one or a few entities using multiple methods, with the goal of obtaining knowledge on a larger class of units (Benbasat et al., 1987; Kothari, 2004; Seawright & Gerring, 2008; Yin, 1994). Yin (1994), adds to this that data gathering and data analysis should be guided by theoretical propositions. Benbasat et al. (1987), furthermore, state that no experimental control or manipulation is used.

Benbasat et al. (1987) and Yin (1994) provide criteria for when the case study method is appropriate. First of all, case studies are useful when a phenomenon should be investigated in its natural context or setting (Benbasat et al., 1987; Yin, 1994). As we discussed, this is the case, as the complexity that the method is designed to deal with results from a complex environment. Without a complex environment, we cannot evaluate the method.

In addition, the case study method is appropriate when there should be a focus on contemporary events (Benbasat et al., 1987; Yin, 1994). This is also true for the evaluation of the method. The complexity of the environment in which context-aware artefacts are designed depend on recent developments in ICT (see section 1.1 and section 3.1). This means that it is not possible to use historical data, for example. The case study method can also be appropriate when no manipulation or control over variables is necessary or possible (Benbasat et al., 1987; Yin, 1994). This is true in our case as well.

Finally, the type of knowledge that needs to be obtained can make the case study method appropriate (Yin, 1994). Case studies are often used and suitable for explanatory research in which causal relationships are investigated. The evaluation should measure how well the artefact supports a solution to the problem (Peffers et al., 2007; Verschuren & Hartog, 2005). It involves determining to what extent the artefact leads to a preferred new situation (Verschuren & Hartog, 2005). This means corroborating a causal relationship between the artefact on the one hand and the effects of its use on the other (Verschuren & Hartog, 2005).

To summarise our approach to evaluating the method, we can use the framework of Pries-Heje et al. (2008). *What* we evaluate is a method for designing context-aware systems in complex environments. *How* we evaluate the method is by performing a naturalistic case study in which data was obtained from humans. *When* we evaluated the method, is after it was developed, i.e. ex post. The results of the evaluation of the method are presented in chapter 14.

4.5 Activity 6: Communication

We published two scientific papers with the results of this part of the research:

- van Engelenburg, S., Janssen, M., & Klievink, B. (2017b). What belongs to context? A definition, a criterion and a method for deciding on what context-aware systems should sense and adapt to. In A. Cerone & M. Roveri (Eds.), *Software Engineering and Formal Methods 2017* (Vol. 10729 LNCS, pp. 101–116). Springer International Publishing AG. http://doi.org/10.1007/978-3-319-74781-1_8
- van Engelenburg, S., Janssen, M., & Klievink, B. (2019). Designing context-aware systems: a method for understanding and analysing context in practice. *Journal of Logical and Algebraic Methods in Programming*, 103, 79–104. <http://doi.org/10.1016/J.JLAMP.2018.11.003>

The first paper presents the new definition that we propose. However, at this stage, the definition was not formalised yet. Furthermore, later on, we made some changes to the terminology used, as in retrospect, the terms used in this paper could lead to confusion. The second paper presents the formalised version of the definition of context and the method.

5 A definition of context

In chapter 3, we established that we need a method to decide easily and systematically what belongs to context and that supports systematically deriving the design of the system from insight into context. To be able to decide clearly what belongs to context, a clear definition of context is required. Furthermore, this definition should be used to model context in such a way that the sensors and adaptors a context-aware system needs and according to what rules it should adapt can be easily determined. We thus need to establish what definition of context we can use as a basis for the method.

First, we describe how we searched the literature for definitions of context in section 5.1.1. In section 5.1.2, we provide an overview of these definitions and discuss their suitability for our purposes. We could not find a definition of context that is suitable to base our method on. Therefore, we developed a new definition of context. We make a distinction between semantics and syntax. We provide our basic syntax in section 5.2.2. Here, we also motivate our choice for formalising the definition and for using a logic-programming paradigm to do so. Next, in section 5.3, we define some basic notions that our definition of context is built upon. In section 5.4, we provide our definition of context.

In this chapter and in chapter 6, we use two systems as running examples, viz. a context-aware tour guide and a context-aware B2G information sharing system. The example of the tour guide will conform with the idea that most readers have of what a traditional context-aware system is, as there is quite some work on it (Schwinger, Grün, Proll, Retschitzegger, & Schauerhuber, 2005). It provides an easy to understand and familiar example. However, the method and definition will be most useful in cases that are more complex.

The B2G information sharing system that we also use as an example is a system that supports B2G information sharing in the international container-shipping domain. This domain is the same domain as for which we developed the context-aware architecture presented in Part III of this dissertation. This domain is highly complex. The complexity is mainly in the high number of things that could belong to context. We use this example to illustrate how the method can be used in practice, but we reduce the complexity in some cases, as our primary aim is still to enhance understanding of the method and definitions. Part III of this dissertation, in which we apply the method and use the definition, shows how they are used fully in a complex environment.

Parts of this chapter have been published in van Engelenburg, Janssen and Klievink (2017b) and van Engelenburg, Janssen and Klievink (2019).

5.1 Definitions of context in literature

In this section, we provide an overview of the existing definitions of context. We start by describing how we searched the literature. Then we provide an overview of definitions in the literature. We also discuss to what extent they are suitable to base a method on that meets the objectives in section 3.2.

5.1.1 Literature review

First, we identified what research is relevant for searching for definitions of context. The definition should be used as a basis for a method for designing context-aware systems and thus should relate to the notion of context-awareness. Other uses of ‘context’ (e.g., in the domain of linguistics) will likely not provide a definition of context that is useful for this research. Therefore, we used ‘definition of context’ and ‘context-awareness’ as search terms.

We made a list of synonyms of these terms based on common sense (e.g., ‘context-aware’) and the synonyms we came across in the literature (e.g., ‘context-based’). We used the search terms to search in the title, abstracts and keywords of papers. The rationale for this is that if a paper presents a new definition of context as part of their contribution, then they will likely make this clear in the title, abstract or keywords. The resulting query that we used to search the literature can be found in appendix A. We chose Scopus as the database to search. This is one of the largest databases of scientific papers and it includes 22,800 titles on a variety of subjects (Elsevier B.V., 2017).

Using the query in Scopus resulted in a list of 493 papers for which the title, abstract or keywords mention ‘definition of context’ and ‘context-awareness’ or their synonyms from 1977 up until the 24th of March 2017 (the date that the search was performed). To reduce the number of papers and to make it feasible to study the ones that are relevant, we excluded all papers that were never cited and thereby excluding 97 papers. Next, we determined for each of the remaining papers based on their abstract whether they include a definition of context as one of their contributions, or provide an overview of the literature on definitions of context or context-awareness. This resulted in a selection of 75 papers. An overview of these papers can be found in appendix A.

For each of the 75 papers, we determined how they defined context or what definition they used. Then we determined whether we could base our method on this definition. We found that these papers did not contain a definition that was suitable for our purposes. Often the definitions were specific to a certain domain and thus do not provide a precise definition of context in general, but merely a list of things that belong to context in that domain. The definitions used in the other papers were directly derived from a handful of other definitions that are not precise enough as well. In the next subsection, we focus on the literature that provides these definitions that are often used in other work. We discuss why these definitions are not suitable for our purposes.

5.1.2 Existing definitions

The large volume of literature on context-aware systems contains many different definitions. There is currently no consensus on the definition of context in the literature (Alegre et al., 2016). The earlier work on context awareness contains definitions that use synonyms for context (e.g. situation, environment) or use examples (e.g. location) (Abowd et al., 1999). This leads to generality in the former case and to incompleteness in the latter case (Zimmermann, Lorenz, Oppermann, & Augustin, 2007). For designers of context-aware systems, such definitions thus respectively provide too little guidance for investigating context or could exclude parts of context that should be included in the design of the system.

In the literature, several attempts have been made to define context for operational use without relying on synonyms or examples. Especially the work of Dey and Abowd is often used as a basis for application-specific or domain-specific definitions (see e.g. (Benou & Vassilakis, 2010; Crowley, Coutaz, & Reigner, 2002; Khedo, 2006; Wang, 2004; Z. Yang, Qilun, & Fagui, 2008)). Dey and Abowd (1999, p. 3) define context as follows: *“Context is any information that can be used to characterize the situation of an entity. An entity is a person, place, or object that is considered relevant to the interaction between a user and an application, including the user and applications themselves.”*

According to the definition given by Dey and Abowd (1999), the most important qualities for belonging to context are 1) characterising a situation and 2) being considered relevant. However, the definition cannot be used as a basis for, for instance, quickly deciding whether something belongs to context. Their definition still leaves implicit what it means to be considered relevant to an interaction and to characterise a situation. We need to know what the notions ‘relevance’ and ‘characterising’ mean to be able to decide whether something belongs to context.

Winograd (2001) argues that the definition given by Dey and Abowd (1999) is too broad. He states that: *“Something is context because of the way it is used in interpretation, not due to its inherent properties”* (Winograd, 2001, p. 405). Zimmerman et al. (2007) also mention this issue with the definition given by Dey and Abowd (1999). Their solution is to categorise context into the fundamental categories of individuality, activity, location, time and relations. According to them, the activity predominantly adds the relevance of context elements.

According to Winograd (2001) and Zimmerman et al. (2007), something is context because of its relationship to something else. This conforms with the interactional view on context described by Dourish (2004). According to Dourish (2004, p. 5), when viewed as an interactional problem, *“contextuality is a relational property that holds between objects or activities”*. The interactional view implies that something belongs to context when it has a relational property with something else. However, we still have no certainty about when exactly this is the case.

Brézillon (2005) states that context cannot be spoken about out of its context. Context thus is always a context of something. According to Brézillon (2005), this ‘something’ is a focus of an actor. Brézillon (2005, p. 57) explains focus as follows: *“Context surrounds a focus (e.g. the task at hand or the interaction) and gives meaning to items related to the focus. The context guides the focus of attention, i.e. the subset of common ground that is pertinent to the current task.”* He views context as knowledge and the focus helps to discriminate irrelevant external knowledge from relevant contextual knowledge. However, as Brézillon (2005, p. 57) himself states, *“the frontier between external and contextual knowledge is porous”*. When in his model for task accomplishment, a discrepancy is found between the model and what a user does, the user is simply asked for an explanation (2005). The new knowledge is then added to the model. This means that Brézillon (2005) does not make explicit what belongs to context either. This decision is ultimately left to the user.

The notion of a contextual element is central to the definition of context given by Vieira et al. (2011, p. 5), namely that a contextual element is *“any piece of data or information that can be used to characterize an entity in an application domain”*. In a similar vein as in the work of Brézillon (2005), contextual elements are relevant to a focus, which is determined by a task and an agent (Vieira et al., 2011).

A contextual element is an attribute of a contextual entity (Vieira et al., 2011), which is an entity that should be considered for the purpose of context manipulation (Vieira et al., 2011). Contextual elements can be identified from the attributes and relationships the entity has (Vieira et al., 2011). Vieira et al. (2011) already noted that the criterion for identifying a property as a contextual element in their case is subjective and depends on the context requirements and a conceptual model. Therefore, the question of what belongs to context becomes a question of what should be in the conceptual model. The problem of determining what belongs to context has thus been moved rather than solved.

In the work above, there is a focus on the relevance of something as arising from an activity or actor (e.g., (Abowd et al., 1999)). Similarly, other work discussing relevance focuses on determining the relevance of something at runtime and dynamically defining context for the specific task or activity at hand (see e.g. (Brézillon & Pomerol, 1999; Vieira et al., 2011)). It is important to make clear the distinction between such work and our work. The work presented in this dissertation is concerned with supporting the determination of what is relevant and what should be included in the context at design time. We thus focus on what a context-aware system should be able to sense and what adaptations it should be able to make, and in what possible situations.

Overall, we could not find a definition in the literature that is precise enough to allow for making easy decisions on what belongs to context, let alone support a method for designing context-aware systems. Therefore, we developed our own definition of context. In the next subsection, we first describe the syntax we used for formalising the definition of context and we describe the reasons for providing such a formalisation. In the last two subsections of this section, we present our new definition of context.

5.2 Formalisation

In this section, we discuss the need for formalising the definition of context and the syntax we used.

5.2.1 The need for a formalisation of the definitions

There are several important benefits of representing the definition of context and related concepts using a formal notation. We set out to define context as precisely as possible as this can help with clearly identifying what belongs to context and this can improve efficiency and effectivity of the design process. The process of formalising enforces a certain level of precision.

The other reasons for formalising the definitions have to do with the way in which we intend to use the definitions in the method later on. Consistently and systematically expressing knowledge about context might help designers to deal with the

high complexity of the environment. More specifically, it could help them with detecting inconsistencies and gaps in knowledge. The second reason for formalising the definitions is thus that it provides designers with the language and semantics to model consistently and systematically the context of their system and thereby this supports systematic investigating and modelling of context.

Furthermore, a formal expression of insight into context could provide for systematically deriving the sensors, adaptors and the rules for adapting to context of the model of context that a designer can build. In addition, the same formalisation could be used in the system to model context at runtime, to express context information gathered by the sensors, and to express what adaptations are required by an adaptor. In addition, it could allow for expressing rules according to which the system should adapt. A formalisation thus also supports deriving sensors, adaptors and rules from the model of context.

We base the formalisation on logic programming, as described by Lifschitz (1996). By using the logic-programming paradigm for our formalisation, the model of context made by the designer can almost directly be translated to rules in a logic program. The context information can then be translated into facts in such a program. This logic program can then be used at runtime to make decisions on what adaptations to make, based on the context information gathered by the sensors.

Using logic programming to formalise our definitions, makes the step of going from a model of context to a logic program that can be used by the reasoning component of a context-aware system very small. This makes it easier for the designer to make such a translation. Furthermore, it helps to ensure that the logic program is very close to what it should be according to the model of context. Our definition of context and related concepts, including a formalisation, can be found in sections 5.3 and 5.4.

5.2.2 Syntax

In early work on logic programming, Kowalski (1974) noted the usefulness of predicate logic in programming. Warren, Pereira and Pereira (1977) implemented a more efficient version of the logic programming language Prolog, based on this work and the work of Colmerauer (1975) and van Emden (1975). Subsequently, many others build on this fundament to further extend and refine logic programming. Lifschitz (1996) provides an extensive survey of the foundations of logic programming with classical negation and negation as failure. Unless otherwise stated, we follow the notation and terminology of Lifschitz (1996) to describe our syntax.

We start with a non-empty set of atoms A . The choice of A depends on the language used. In our case, the atoms are simple predicates, as defined below. We directly introduce variables in the language and introduce the notion of schematic atoms.

Definition 11 (atom):

Given a set of constant symbols $C = \{a, b, c, \dots\}$ and a set of variable symbols $V = \{X, Y, Z, \dots\}$, any constant $c \in C$ is a term, and any variable $X \in V$ is a term.

Given a set of predicate symbols $S = \{p, q, r, \dots\}$, an expression $p(t_1, \dots, t_n)$ where $p \in S$ is an n -ary predicate symbol and t_1, \dots, t_n are terms, is an atom.

If one or more terms of t_1, \dots, t_n are variables, then $p(t_1, \dots, t_n)$ is called a schematic atom.

If terms t_1, \dots, t_n are all constants, then $p(t_1, \dots, t_n)$ is called a ground atom.

Atoms are called positive literals. Atoms preceded by a classical negation symbol ' \neg ', are called negative literals. Following Lifschitz (1996) again, we refer to a positive literal or a negative literal as a literal. A schematic atom is called a schematic literal. A ground atom is called a ground literal. We follow the convention that terms with a capital as their first letter denote variables.

Example 1 (atom):

- $isUser(User)$ is an atom and a positive schematic literal
- $\neg isUser(mary)$ is a negative ground literal

Context rules are the same as basic rules defined by Lifschitz (1996). Again, we also introduce the notion of schematic context rule in the same definition.

Definition 12 (context rule):

A context rule is an ordered pair $Head \leftarrow Body$, where $Head$ is a literal and $Body$ is a finite set of literals. A context rule with head L_0 and body $\{L_1, \dots, L_n\}$ can be written as $L_0 \leftarrow L_1, \dots, L_n$. If the body is empty, then \leftarrow can be dropped. If one or more literals of L_0, \dots, L_n are schematic literals, then $L_0 \leftarrow L_1, \dots, L_n$ is called a schematic context rule.

If L_0, \dots, L_n are all ground literals, then the rule $L_0 \leftarrow L_1, \dots, L_n$ is called a ground context rule.

Example 2 (context rule):

The following schematic context rule expresses that when the sight that is the subject of information and a user are less than 150 metres from each other, the information should be provided to the user by the system.

$isProvidedTo(I, U) \leftarrow$
 $isUser(U),$
 $subjectOf(S, I),$
 $hasLocation(U, L_1),$
 $hasLocation(S, L_2),$
 $distance(L_1, L_2, Distance),$
 $Distance < 150$

The following ground context rule expresses that when Mary is at the location with coordinate $29^{\circ}58'45.03''\text{N } 31^{\circ}08'03.69''\text{E}$, then *info* should be provided to Mary by the system.

isProvidedTo(info, mary) ←
isUser(mary),
hasLocation(mary, 29°58'45.03"N 31°08'03.69"E)

In addition to context rules, we need to express context relationships. The definition of a context relationship rule is similar to that of a context rule and it is also based on the definition of a basic rule by Lifschitz (1996). However, it uses a different operator to connect the head and the body of the rule.

The use of a different operator signifies that context rules and context relationship rules are different types of rules. Context rules are used to express what adaptations should be made in different situations. The body of the rule expresses the situation in which the adaptation needs to be made. The head expresses what adaptation should be made by the system, namely an adaptation that makes the head true. Context relationship rules express a dependency of the truth of the head of the rule on the situation expressed in its body. In other words, context relationship rules express that their head is true when their body is true. This is a different type of relationship. Context rules are further discussed in section 6.3. Context relationships are further discussed in section 5.4.

Definition 13 (context relationship rule):

A context relationship rule is an ordered pair $Head \Leftarrow Body$, where *Head* is a literal and *Body* is a finite set of literals. A context relationship with head L_0 and non-empty body $\{L_1, \dots, L_n\}$ can be written as $L_0 \Leftarrow L_1, \dots, L_n$. If one or more literals in L_0, \dots, L_n are schematic literals, then $L_0 \Leftarrow L_1, \dots, L_n$ is called a schematic context relationship rule.

If L_0, \dots, L_n are all ground literals, then the rule $L_0 \Leftarrow L_1, \dots, L_n$ is called a ground context relationship rule.

Example 3 (context relationship rule):

The following schematic context relationship rule expresses that when the sight that is the subject of information and a user are less than 150 metres from apart and the information is provided to the user, then it is relevant.

hasRelevance(I, U, relevant) ←
isProvidedTo(I, U),
isUser(U),
hasLocation(U, L₁),
hasLocation(S, L₂),
distance(L₁, L₂, Distance),
Distance < 150,
subjectOf(S, I)

The following ground context relationship rule expresses that when the Pyramid of Cheops is the subject of information *info* and Mary is at a

location with coordinate 29°58'45.03"N 31°08'03.69"E and *info* is provided to Mary, then information *info* provided to Mary is relevant.
 $hasRelevance(info, mary, relevant) \Leftarrow$
 $isProvidedTo(info, mary),$
 $isUser(mary),$
 $hasLocation(mary, 29°58'45.03"N 31°08'03.69"E),$
 $subjectOf(pyramidOfCheops, info)$

According to Lifschitz (1996), $Ground(R)$ stands for all the ground instances of a schematic rule R . The same function can be applied to the literals, context rules and context relationships in our case to make them stand for the set of their ground instances. For example, $isUser(mary)$ can be a ground instance of $isUser(User)$.

A program is a set of rules. In this case, we will use the context rules to reason with in a logic program to derive what adaptations need to be made based on the context information sensed. As we will further explain in section 5.4, context relationship rules have a different function and we do not need to use them in a logic program.

Definition 14 (logic program):

A logic program is a set of context rules.

Example 4 (logic program):

The expression below is a logic program. Its meaning is discussed in detail in section 6.3 (parentheses used to separate the context rule from the literals).

$$\left\{ \left(\begin{array}{l} isProvidedTo(I, U) \leftarrow isUser(U), subjectOf(S, I), hasLocation(U, L_1), \\ \quad hasLocation(S, L_2), distance(L_1, L_2, Dist), Dist < 150 \\ isUser(mary), hasLocation(mary, 29°58'45.03"N 31°08'03.69"E), \\ \quad hasLocation(pyramidOfCheops, 29°58'27.00N 31°08'2.21E), \\ \quad subjectOf(pyramidOfCheops, info) \end{array} \right) \right\}$$

5.3 Environment elements, situations and the focus of a context

Context-aware systems should sense context and adapt to context. To sense, there needs to be something in the real world that can be observed. For instance, the GPS coordinates of a user can be observed. Furthermore, to adapt, there needs to be something in the real world that can be manipulated; for instance, the information provided to a user can be manipulated. These things should be part of the context. First, we thus have to define what elements of the environment can be observed and manipulated by a system. These elements are a candidate for being part of the relevant context of the context-aware system.

In previous work on defining context (van Engelenburg, Janssen, & Klievink, 2017b), we focussed on which attributes of objects do belong to context and which do not. The attributes were viewed as possibly having different values. For instance,

according to this work, the attribute ‘colour’ of an object ‘apple’ could have values such as ‘green’ or ‘red’. Relationships were also reduced to attributes, as the benefit of including them was unclear at the time and attributes seemed easier to deal with. Furthermore, we built on other work that also describes context as attributes (Dourish, 2004; Vieira et al., 2011).

As the research progressed, we realised that the definition of context can be used not only for deciding what belongs to the context of a system, but also as a basis for guiding the information gathering process for getting insight into context. While investigating the context of the B2G information sharing system, we came across complex relationships that should be taken into account in the context-aware system. Using a definition that only includes attributes requires each of these relationships to be reduced to an attribute. When relationships are complicated, this is counterintuitive and makes the investigation of context more complex.

We illustrate the counterintuitivity of reducing relationships to attributes with an example. For the B2G information sharing system, we want to ensure that it supports flows of information in which businesses are willing to participate. In some cases, business A might not want business B to have certain data elements, because A and B are competitors. When data is shared in such a way that B can access it, A might not be willing to participate. However, business A might want to share other data elements with business B that are not sensitive. Sensitivity, in that case, can most intuitively be described as a relationship between businesses A and B and a data element.

Attributes can be easily described as relationships. For instance, an apple can have a ‘has colour’ relationship with ‘green’ or ‘red’. For the context-aware tour guide, we want to provide information about sights that is relevant to the user. What information is relevant probably depends on where the user is. In the previous work, the user would have had an attribute ‘location’ that has a coordinate as a value. In the new definition of context, the user has a ‘has location’ relationship with a coordinate.

In addition, it only makes sense to try to sense or manipulate something that could change or vary. For example, the location of a user can vary as they move around. When multiple people use a system, then who is the user might vary as well. Both of these things might be useful to sense. However, other things are less likely to vary or are not variable at all. For example, the speed of light is not variable. For the tour guide, it might turn out that (almost) all users prefer being provided with the information in the same language. In that case, it is not useful to sense what language users prefer.

The relationships are the elements of the environment for which we want to decide whether they are part of the relevant context. Therefore, we will refer to such a relationship as an environment element. It is important to note, that by ‘environment’ here we refer to the environment of the system. We made a selection of what in the environment could be manipulated or sensed. We have not yet selected what of those things are part of the relevant context that should be manipulated or sensed. In this case, ‘environment’ thus should be interpreted in the broadest sense possible.

Informally, an environment element is a relationship between objects in the environment of a system. The objects in the environment could include physical objects,

but also other things, such as qualities, or locations. Syntactically, they are represented by literals.

For example, the literal *hasLocation(mary, 29°58'45.03"N 31°08'03.69"E)* expresses that a user Mary is at the location with coordinate *29°58'45.03"N 31°08'03.69"E*. As another example, business A can have a 'willingness to participate' relationship with a flow of information and a level of willingness 'willing'. This environment element is expressed as the literal *willingnessToParticipate(businessA, informationFlow, willing)*.

The semantics of environment elements, i.e., whether the literal is true or not, will be determined in the system using sensors or will be manipulated by an adaptor. In our modelling, we only want to include literals that can have different truth-values.

Definition 15 (environment element):

An environment element is expressed by a ground literal $p(s_1, \dots, s_n)$, where p is a predicate symbol and s_1, \dots, s_n denote environment objects. A schematic literal L can be used to express the set of environment elements expressed by the literals in $Ground(L)$.

A state of the world, or a situation, is different from another state of the world or situation when the truth value of at least one environment element changes. For a system to be context-aware, it should sense or adapt to these differences when they are relevant. Furthermore, it should only consider situations that could exist in the real world, and for instance not situations that are inconsistent.

A situation is a state of the world determined by the environment elements that are true. Syntactically, they are represented by a set of ground literals.

For example, there could be a situation in which Mary has a 'has location' relationship with coordinate *29°58'45.03"N 31°08'03.69"E* and in which she is a user of the context-aware tour guide. This can be described by the set $\{hasLocation(mary, 29°58'45.03"N 31°08'03.69"E), isUser(mary)\}$.

This situation is different from one in which the location of Mary is *29°58'31"N 31°08'16"E*. This situation can be described by the set $\{hasLocation(mary, 29°58'31"N 31°08'16"E), isUser(mary)\}$. In addition, the set of literals $\{hasLocation(mary, 29°58'31"N 31°08'16"E), \neg hasLocation(mary, 29°58'31"N 31°08'16"E)\}$ does not express a situation, as it is inconsistent.

As another example, for the B2G information sharing system, a situation could exist in which business A has a sensitivity relationship with data element 'client name' and business B. This situation is different from one where the client name is not sensitive from B according to A. Respectively, these situations are expressed by the sets $\{isSensitive(businessA, clientName, businessB)\}$ and $\{\neg isSensitive(businessA, clientName, businessB)\}$.

Definition 16 (situation):

A situation is expressed as a nonempty and consistent set of ground literals $\{L_1, L_2, \dots\}$, where each L_i expresses an environment element that is true. A finite set $\{L_1, \dots, L_n\}$ describes part of a situation and stands for all situations in which L_1, \dots, L_n are true.

The origin of the notion of context lies in the domain of linguistics (Oxford Dictionaries, 2018). In this domain, the meaning of a text has to be constructed based on the surrounding text (Winograd, 2001). This surrounding text can be viewed as the context of the text for which the meaning is constructed.

Outside of the domain of linguistics, context is always a context of something as well. We need to identify what this something is and what to call it. In linguistics, this ‘something’ (i.e. the text for which the meaning is constructed) is called a ‘focal event’ (Goodwin & Duranti, 1992). In the domain of context-aware systems, for Dey (2001), it is the interaction between a user and an application, and for Brézillon (2005) and Vieira et al. (2011), it is a focus. Accordingly, we refer to the entity that context is a context of as a focus. Context is thus the context of a focus.

Now we have a name for it, we need to determine what a focus is in the case of context-aware systems. We want our definition of context to be generic enough to make our method useful for a variety of application domains in which a context-aware system is designed. Therefore, we believe that limiting the focus to the interaction between a user and an application, like Dey (2001) does, is too restrictive. We have to look more broadly at and examine what the nature is of the relationship between contexts and their focus for context-aware systems.

The designer of a context-aware system has a goal and they want to design the system to reach that goal (Finkelstein & Savigni, 2001; Simon, 1996). This design goal is determined by the designer based on the design goals of the different stakeholders in the context-aware system. The design goal is the same in all situations (Finkelstein & Savigni, 2001). However, whether the design goal is reached can depend on the situation. Everything that could affect whether the design goal is reached at runtime should belong to the context; that is, the situations in which the design goal is not reached should be sensed. This should then result in an adaptation by the system that changes the situation in such a way that the design goal is reached. Something that does not affect whether the design goal is reached should not belong to the context that the designer should take into account. The focus of the context should thus be related to the design goal of the designer.

The focus of context is the environment element that a designer of context-aware systems needs to be true to reach their design goal. As it is an environment element, it is syntactically represented by a literal, in the same way as other environment elements.

For example, a designer can have the design goal of developing a context-aware tour guide that provides information about sights that is relevant to users. This goal is reached when the information provided is relevant to users. This focus can be expressed by the literal *hasRelevance(ProvidedInfo, User, relevant)* when there are several different levels of relevance. Alternatively, it can be represented by the literal

isRelevant(ProvidedInfo, User), when the information is viewed as either relevant or not relevant.

Note that what information is relevant is the part that is important here. This is what should be adapted to in the end. This is not the case for what information is provided to the user. Just providing information without requiring relevance of the information does not require context awareness. Therefore, the predicate we use is *hasRelevance* and not *isProvidedTo*. The decision for designing a system that is context-aware for reaching a goal is in part one that should be made by the designer on beforehand. As the focus is directly dependent on the goal of the designer, the designer has as much freedom to select a focus, as they have to select a goal.

In the case of the B2G information sharing system, a designer can have a design goal of developing a context-aware system that supports flows of information in which businesses are willing to participate. This goal is reached when the system supports flows of information in which businesses are willing to participate. The focus is expressed by *willingnessToParticipate(Business, SupportedInformationFlow, willing)*.

It is important to note the use of schematic literals in the examples here. The design goals of a designer are usually high level. The designer's goal is not to provide Mary and Bob with relevant information, but to provide all users with relevant information. Schematic literals can be used to reflect this.

Definition 17 (focus):

A focus of context can be expressed using a literal in the same way as other environment elements.

5.4 Context relationships, context elements and context

At first sight, achieving a definition of context seems problematic, since what belongs to context might be different for each context-aware system. However, context is determined by its relationship with its focus. In fact, something only is context if it has some *context relationship* with the focus. For instance, the weather forecast only belongs to the context of the tour guide if it has a context relationship with the relevance of the information provided to the user. The type of relationship is not specific to a certain focus, but the same for all foci. In this way, it can be used to formulate a definition of context from which what belongs to the context of a specific focus can be derived.

A context relationship is a relationship between a focus and a set of environment elements, where in each situation where these environment elements have the same truth-value, the focus has the same truth-value. We say that these situations restrict the focus. Syntactically, context relationships are represented by context relationship rules.

Note that the context relationship is not the same type of relationship as the environment elements. It connects different environment elements with each other. The context relationship is thus on a higher level and can more naturally be represented by an operator than by a predicate.

For the context-aware tour guide, the focus is the relevance of the information provided to the user. Let us assume that information about a sight is relevant when the user is close to the sight and the information is about the sight. This means that in all

situations in which the ‘has location’ relationship of the user has a coordinate within a few metres of a sight and the ‘subject of’ relationship of the data provided is this sight, the value of the focus is such that the information provided is relevant to the user. In that case, there is a context relationship. For completeness, it should be noted that, for instance, the ‘is user’ relationship between the user and the context-aware system and the ‘provided to’ relationship restrict the focus as well.

For the B2G information sharing system, the focus is the relationship of willingness to participate between a flow of information, a business and a level of willingness. We found that businesses are not willing to participate in a flow of information when the data is sensitive to them and when a system in the flow of information broadcasts it. Therefore, the previously mentioned sensitivity relationship has a context relationship with the focus. The same is the case with the broadcasting relationship a system has with information and the ‘being part of’ relationship of a system and a flow of information.

This context relationship can be expressed as follows:

$\neg \text{willingnessToParticipate}(\text{Business}, \text{SupportedInformationFlow}, \text{willing}) \Leftarrow$
 $\text{sensitive}(\text{Business}, \text{Competitor}, \text{Data}),$
 $\text{partOfFlow}(\text{SupportedInformationFlow}, \text{System}),$
 $\text{broadcasts}(\text{System}, \text{Data}).$

It is important to note that businesses might be either willing or unwilling to share when the information is not sensitive, the system is not part of the flow of information or the system does not broadcast the information. This is possible, considering the context relationship we have identified. There is only a dependence on the focus when all the environment elements have the values mentioned in the example. Context relationships, however, in other cases might constrain the value of a focus for another truth-value of their context elements as well. In addition, there might be multiple sets of context elements that have a context relationship with a single focus.

For example, what the author of this dissertation has for dinner does not have a context relationship with either the focus of the tour guide or the focus of the B2G information sharing architecture, as this does not restrict them.

Definition 18 (context relationship):

A context relationship can be represented by a ground context relationship rule $L_0 \Leftarrow L_1, \dots, L_n$, where L is a literal representing the focus and where L_1, \dots, L_n are literals representing the context elements that restrict the focus.

A schematic context relationship rule R can be used to represent the set of context relationships represented by $\text{Ground}(R)$.

It is important to note the effect of using schematic rules to represent context relationships. For example, the distance between two locations will usually not be subject to change. This means that the ‘distance’ relationship between locations will not be part of the context relationship in the case of the context-aware tour guide. This means that in theory, expressing that the location of the user and that of the sight have to be within a

certain distance could be done by listing the same rule over and over again, each time with different locations that are within this distance. In practice, this would be impossible. It would be much easier to just use a schematic rule that stands for this set of rules and includes the relationship of ‘distance’ to constrain the instances that are represented by the schematic rule. There is no obvious prohibition against allowing this, as long as it is clear that from a semantic point of view, that these literals represent constraints and are not part of the set of environment elements that have a context relationship with the focus. In the case of the context-aware tour guide, such a schematic rule, including the constraints could be expressed as follows (constraints are underlined):

hasRelevance(ProvidedInfo, User, relevant) \leftarrow
isUser(User),
hasLocation(User, Location1),
providedTo(User, ProvidedInfo),
hasLocation(Sight, Location2),
distance(Location1, Location2, Distance).
Distance < 10.
subject(ProvidedInfo, Sight).

Using the notion of context relationship, we can determine whether an environment element belongs to context. A context element of a focus is an environment element that is part of a set of environment elements that have a context relationship with the focus. As it is an environment element, it is syntactically represented by a literal in the same way as other environment elements.

For example, the location of a user is a context element of the focus in the tour guide example. If it is unlikely that the location of a sight will change, or if it is impossible for it to change (e.g. in the case of the Pyramid of Cheops), then the ‘has location’ relationship of the sight with a coordinate does not have a context relationship with the focus, because its truth value never changes and therefore it is not an environment element.

Furthermore, the sensitivity relationship is a context element of the focus of the B2G information-sharing example, as it has a context relationship with that focus. In contrast, what the author of this dissertation has for dinner is not a context element, as it does not have a context relationship with the focus.

Definition 19 (context element):

A context element can be expressed by a literal in the same way as other environment elements.

When an environment element is a context element of a focus, this means that it is relevant to the designer. A designer achieves their design goal when the focus has a certain value. To achieve the design goal, they thus have to design the context-aware system such that the focus has this value when it is used. A context element of the focus influences the value of that focus. Therefore, the system needs to be designed such that it can sense the context elements and manipulate them if the focus has an undesired value. This makes the context element relevant to the design and therefore to the designer.

The definition of context is based on the other notions defined above. The context of a focus is the set of all its context elements.

For example, the context of the focus of the tour guide example includes the location of a user. In addition, the context of the focus of the B2G information-sharing example includes the sensitivity relationship. Syntactically, context is represented by a set of literals.

Definition 20 (context):

The context of a focus can be expressed by a set of literals $\{L_1, L_2, \dots\}$, where each L_i expresses an environment element.

6 A method for designing context-aware systems

In this section, we present the three steps of the proposed method for designing context-aware systems based on insight into context, viz. 1) getting insight into context, 2) determining the components needed to sense context and adapt to context, and 3) determining the rules for reasoning with context information. The method takes a practical problem that a designer wants to solve as a starting point. In step 1, it is determined what context relationships and context elements the system should take into account to solve the problem and attain the designer's goal. In step 2, the list of context elements is used to determine what sensors and adaptors are needed. In step 3, the list of sensors and adaptors, together with a list of context relationships from step 1, is used to derive and express the rules according to which the system needs to adapt. Steps 1 and 2 consist of several sub-steps. For each sub-step, we provide an illustration of how it can be performed for the examples of the tour guide and the B2G information sharing system.

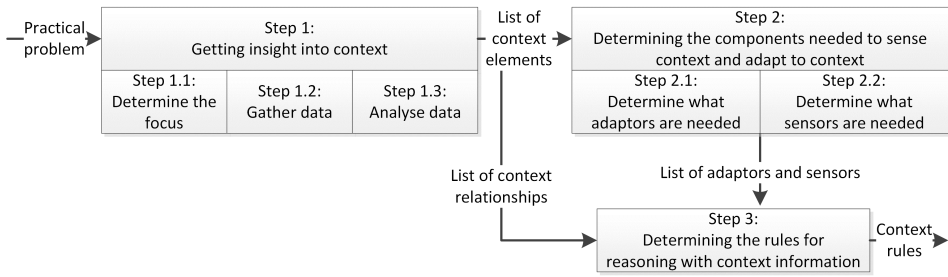


Figure 1: Overview of the steps in the method and their input and output

Parts of this chapter have been published in van Engelenburg, Janssen and Klievink (2017b) and van Engelenburg, Janssen and Klievink (2019).

6.1 Step 1: Getting insight into context

The objective of the designer in the first step is to get insight into context. The first step of our method should be preceded by the identification of the practical problem that the context-aware system should solve and a determination of the relevance of that problem. How to do this is already described extensively in the scientific literature (see e.g. (Hevner et al., 2004; Peffers et al., 2007)). This is thus not new, and therefore not part of our method.

However, what context should be taken into account in the design of the system is related to the design goal of the designer. We can derive this design goal from the problem that is identified. The specification of the practical problem is thus the input for this step.

The sensors and adaptors of a context-aware system and the rules that the system should reason with depend on the context that should be taken into account. This means that in this step insight into context needs to be gained to determine what belongs to context and what the impact of different situations is. Information about this should thus be the output of steps 2 and 3. This information should be structured in such a way that

the necessary sensors and adaptors for the architecture, as well as the rules, can easily be derived from it.

For the first step of our method, we propose that the designer performs the following sub-steps: 1.1) determine the focus, 1.2) gather data, 1.3) analyse the data.

6.1.1 Step 1.1: Determine the focus

The overall process for determining the focus (step 1.1) is shown in figure 2. The input of this step is the practical problem and the output is a focus.

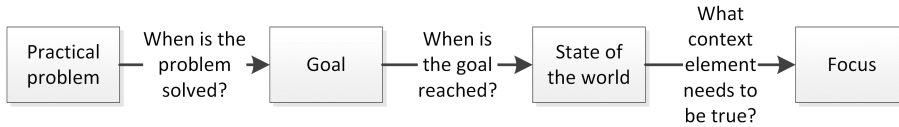


Figure 2: Determining the focus

The focus of the context is related to the design goal of the designer. The design goal of the designer, in its turn, is based on the problem that they want to solve with their system. To perform step 1.1, the designer can use the specification of their problem to specify their design goal.

The design goal of the designer can be viewed as solving the practical problem. The design goal can thus be expressed as what the system should be able to do at a very high level in order to solve the problem. A design goal is reached when the world is in a certain state. Therefore, the next step for the designer is to describe the state of the world when their design goal is reached.

According to definition 17, a focus of a designer is the environment element that the designer needs to be true to reach their design goal. This relationship can be identified from the description of the state of the world. Figure 3 shows the steps for deriving the focus for the tour guide example.

It is possible that a more complex design goal can lead to multiple foci, or that multiple design goals exist. This should not be a problem. However, each of the foci will have its own context relationships and context elements and will require its own investigation by the designer.

Example 5 (step 1.1, tour guide):

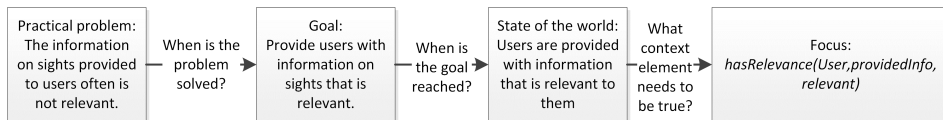
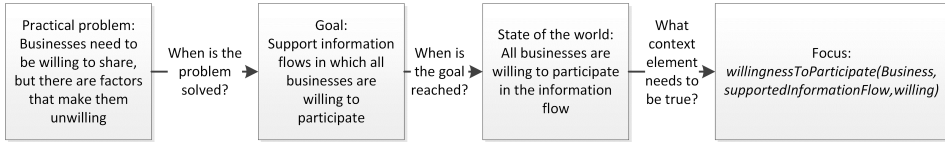


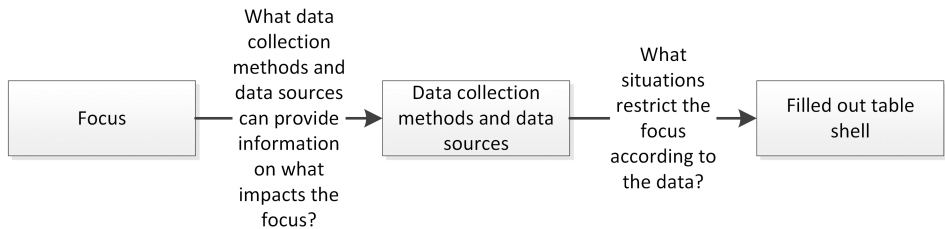
Figure 3: Deriving the focus for the example of the context-aware tour guide

Example 6 (step 1.1, B2G information sharing system):


Figure 4: Deriving a focus for the example of the B2G information sharing system

6.1.2 Step 1.2: Gather data

The overall process of performing step 1.2 is shown in figure 5. The input is the focus from step 1.1 and the output is a table with situations that restrict the focus.


Figure 5: Gathering data on the context of a context-aware system

After the designer has identified the focus, they should first select the data collection methods and sources that they will use to investigate context. An important requirement is that the data that the designer gathers should provide information on what environmental elements restrict the focus. Many different approaches could be taken. A designer could, for instance, perform case studies in which the focus has certain values and then determine why the focus has these values. Another possible approach is to do a literature search, using a description of the focus. Furthermore, a designer could conduct interviews and ask the interviewees directly or indirectly what they think impacts the focus. In addition, the usual considerations, such as the accuracy and accessibility of data, will also play a role in making a choice.

Example 7 (step 1.2, tour guide):

For the tour guide example, the selection should depend on what data says something about the level of relevance of information about sights to users. This could very well be the users themselves, and the designer might ask them to complete questionnaires or interview them about in what cases they find information relevant. Among the other possibilities are interviewing experts or doing a literature study on the matter.

Example 8 (step 1.2, B2G information sharing system):

For the willingness focus, we needed information about in what situations businesses are and are not willing to participate in information flows. We

did a secondary analysis of case study data from a project in which different systems for information sharing were implemented and tested in the container-shipping domain. The systems had different designs and were used by different parties. The case study data thus offered insight into the reasons for these differences and the concerns of businesses. We studied different deliverables and reports describing the design of the systems, the progress of the project and the obstacles the researchers came across. Furthermore, we studied some of the case study notes. The data provided a broad perspective on what things affect the willingness of businesses. Furthermore, this data was rich and accessible.

It is hard to identify the best data source and the best data collection method without knowing the focus for which data should be collected. However, as we are investigating what belongs to context, we can say that data gathering should be of an exploratory nature. This is most important in the early stages of the investigation to ensure that no important parts of the context are excluded. In later stages, additional support should be sought for the context elements and relationships found. Unfortunately, the exploratory approach can be in conflict with the feasibility and efficiency of the data gathering. In a large search space, such as in the case of the complex environments discussed in the introduction, it would be easy to end up investigating many things that later turn out to be irrelevant to the design of the system.

We propose several strategies to minimise the time that the designer spends on investigating things that turn out not to matter for the design of the context-aware system. First, we already restricted the search space by using the focus to select appropriate data collection methods and data sources. Second, we can restrict the kind of information that the designer should gather. The designer only needs to know in what situations the focus is restricted. The designer will need very specific descriptions of the situation and the way in which the focus is restricted. However, they do not need to do a more in-depth analysis of why the focus is restricted in that situation than is necessary to determine that there is a connection and that the information is reliable. For instance, the designer needs to know in what situations a user finds information about sights relevant, but does not need to know the details of theories on human information processing that make that information relevant. After all, this is not something that the context-aware system can directly take into account.

An additional way to increase the efficiency of the data gathering process is to provide a way for designers to decide quickly whether something has a context relationship with the focus. Therefore, we provide a criterion for deciding whether a relationship in the environment has a context relationship with the focus. Furthermore, we provide a simple test to decide whether the criterion is met for a set of environment elements. This criterion can be used to discern pieces of data that are interesting for further analysis and those that are not. The criterion follows directly from the way we defined the notion of environment elements and context relationships in chapter 5.

Criterion:

A relationship in the environment has a context relationship with a focus if and only if:

- whether the relationship exists can vary (making it an environment element), and
- it is part of a set of environment elements, such that in each situation where these environment elements have the same truth value, the focus has the same truth value.

The criterion is met when a situation restricts the truth-value of the focus and the relationship is part of an environment element that is part of this situation. Of course, we cannot list all possible situations to check whether the criterion has been met, because in the real world there are far too many other environment elements that vary. Therefore, it is also not possible to be certain that all environment elements belonging to a set that have a context relationship with a focus have been found.

The solution is to reduce the testing of the criterion to testing whether the information collected or analysed by the designer supports the conclusion that the criterion is met. The designer should thus determine that their information supports the conclusion that the focus is restricted to a certain truth-value in a specific situation. If information is found that indicates that the truth-value is restricted for a focus to either true or false in a certain situation, the criterion is met for all the environment elements in the situation. To test whether the criterion is met, it is thus enough to describe the situation that impacts the focus. This is more intuitive and efficient, and at this stage, we do not need to discern the different context elements.

Furthermore, depending on what data the designer gathered, they may need to generalise. Consider an example in which Mary is interviewed by the designer and she states that she thinks information about the Pyramid of Cheops is relevant when she is near the pyramid. Based on this, we could say that in the situation in which Mary uses the system, her location is near the Pyramid of Cheops, and the information provided is about the Pyramid of Cheops, the information provided to Mary is relevant. However, unless the system is designed specifically for Mary and the pyramid, this is not very informative. Generalisation, in this case, is easy: replace ‘Mary’ with ‘the user’ in the description of the situation and ‘the Pyramid of Cheops’ with ‘the sight’. By generalising, the description stands for a whole set of situations with particular properties in which the designer believes that the focus is restricted.

Of course, the designer should be confident that they can make generalisations based on the information that they gathered. If a relationship meets the criterion but the designer is not sure whether they can generalise, then gathering further data on the situation that relates it to the focus and similar situations and their impact on the focus might be fruitful. It is not always necessary for designers themselves to generalise. For instance, when scientific research shows that location and relevance are related, designers do not need to generalise.

To ensure that the designer has found the appropriate information, they should describe explicitly and precisely the restriction on the value of the focus as well as the

situations. Designers should ensure that everything they say in the description of the situation follows from the information that they have. We propose that designers fill in a table shell similar to that in table 3. If they can fill this in based on the information they have gathered, they have found environment elements that have an impact on the focus and that meet the criterion. If they cannot fill it in, then they cannot conclude that the criterion is met, based on the information they gathered. When filling in the table shell, designers should keep in mind that situations should be possible in the real world.

Restriction on focus	Situation	Support
Description of to what value focus is restricted	Description of the situation in which the focus is restricted	Reference to a data source or citation

Table 3: Table shell for testing whether the criterion is met

Example 9 (step 1.2, tour guide):

Restriction on focus	Situation	Support
The information provided to the user is relevant	The user is near the sight. The information provided is about the sight.	Interview 1, 00h21m: Interviewee: <i>"I think the information about the Pyramid of Cheops is relevant to me when I am near it."</i>

Table 4: Testing whether the criterion is met for the tour guide

Example 10 (step 1.2, B2G information sharing system):

For the investigation of context for this example, we went through the case study data. For everything we thought might restrict willingness, we attempted to fill in the table shell and find additional information. We explicitly made the step to generalise from specific situations by replacing objects in our descriptions with their type (e.g. 'freight forwarder' with 'business'). When different situations generalised to a similar overall description, we added the new situation to the older one as support. The table below shows an example of the results of filling in the table shell. We took an example for which the support was already generalised.

Restriction on focus	Situation	Support
Businesses are not willing to participate in the flow of information.	The data in the flow is sensitive for a business to another business. The data is shared with the business that the data is sensitive to.	Doc 1: <i>"A freight forwarder that books a container transport on behalf of an exporter at an ocean carrier, is not willing to share the name of this exporter, because they fear that the exporter and ocean carrier might bypass him ("disintermediation"), and make a direct container transport booking with the carrier."</i>

Table 5: Testing whether the criterion is met for the B2G information sharing system

By filling in the table shell, designers can determine what things are and what things are not interesting to look into further. Then, they can concentrate their data-gathering efforts on further specifications of the context elements and relationships that they found, and on finding more support for them. In fact, to make everything computable, the description of the situation and the value for the focus need to be as specific as possible. In the tour guide example, the designer, for instance, might attempt to try to find out what exactly is near enough to the sight for the information to be relevant. The filled-in table shells are further analysed in step 1.3. This means that it is important for the designer to add further specifications to the table shell.

Filling in the table shell not only serves as a test that the criterion is met, but is also used in further steps to determine what the context relationships and context elements are of the focus. As the information in the table is further processed later on, it seems wise to include a reference to the data source or citation of the text that the information is based on. This can help to ensure reliability and allow for reinterpretation in later steps.

6.1.3 Step 1.3: Analyse the data

The overall process of performing step 1.3 is shown in figure 6. The input is a table with descriptions of situations from step 1.2. The output is a list of context elements and context relationships.

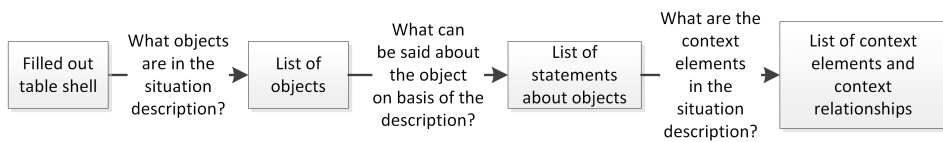


Figure 6: Analysing the data on the context of a context-aware system

As the output of step 1, we require the information found about the context to be structured in such a way that it is easy to identify what sensors and adaptors the system should need and such that rules can be abstracted from it. The first thing that needs to be done to achieve this is to abstract the environment elements and their truth-values from the situations described in the table shell that is filled in step 1.2.

There are many ways to do this. One is by looking at the descriptions of the situations and for each one identifying all the physical things in the real world that are mentioned. Physical things are physical objects in the world, such as a user. Other things are not physical. For example, qualities are things that are attributed to those objects, such as colour or speed (Rettler & Bailey, 2017), but are not physical objects themselves. Both physical things and qualities can be related in an environment. To find the environment elements, everything that is true about the physical objects in the situation can be listed and then it can be determined what of these things vary.

Example 11 (step 1.3, tour guide):

On the basis of the filled-in table shell (table 4), we can identify the following objects: the user, the location of the user, the sight, the location of the sight, and the information. Then we can list everything that is true

about these objects in the situation: there is a user, there is a sight, there is information, the user has a location, the user is provided with the information, the location of the user is near the location of the sight, the information is about the sight, and the sight has a location.

Example 12 (step 1.3, B2G information sharing system):

For the first situation in the table shell resulting from step 1.2 (table 5), we can identify the following objects: the data, the flow of information, a business, and another business. What we can say about these objects is the following: there is data, there is a flow of information, there is a business, there is another business, the data is sensitive from one business to the other, and the data is shared with the other business in the flow.

In the previous section, we discussed how to express environment elements (definition 15). We can use this here to express the environment elements that are found during the analysis. We can use ground literals when we know about or want to express a specific instance of an environment element. For instance, when the location of Mary is a specific coordinate, we express this using a literal, for example, *hasLocation(mary, 29°58'45.03"N 31°08'03.69"E)*.

When we want an adaptor to provide Mary with a piece of information *info*, we can express this using ground literals as well, for instance, *isProvidedTo(info, mary)*. The ground literal then expresses what the desired situation is and the adaptor should perform an action to achieve this situation, for example, provide *info* to Mary.

In the previous step, we already generalised the information we found on context. For instance, we replaced ‘Mary’ with ‘the user’. This generalised information can be expressed using schematic literals that represent a set of instances. As discussed in section 5.4, this avoids having to provide a very long list with similar context relationships. It also introduces the need to express constrictions in some cases, which can be added as literals as well.

We can add the schematic literals expressing each of the environment elements to extend the table shell we used for testing the criterion (table 3). In each row, we describe a situation and the way in which the focus is restricted in that situation. We are thus describing the relationship between context and the focus, or in other words, a context relationship. Therefore, we can also add a column to name the context relationship. The resulting extended table shell is shown in table 6.

Name	Restriction on focus	Situation	Context elements	Support
Name of the context relationship	Description of to what value focus is restricted	Description of the situation in which the focus is restricted	List of environment elements identified in the description	Reference to a data source or citation

Table 6: Table shell for recording information on context relationships

Example 13 (step 1.3, tour guide):

Table 7 shows the literals that can be assigned to the statements for the tour guide example.

Name	Restriction on focus	Situation	Context elements	Support
Proximity of the user to the sight	The information provided to the user is relevant	The user is within 150 metres of the sight and the information provided is about the sight	$isUser(U)$	Interview 1, 00h21m: Interviewee: “ <i>I think the information about the Pyramid of Cheops is relevant to me when I am near it.</i> ” Field-testing: a sight and a user are near when they are closer than 150 metres.
			$subjectOf(S, I)$	
			$haslocation(U, L_1)$	
			$haslocation(S, L_2)$	
			$isProvidedTo(I, U)$	
			$distance(L_1, L_2, Dist)$	
			$Dist < 150$	

Table 7: Recording information on context relationships for the tour guide example

Example 14 (step 1.3, B2G information sharing system):

Table 8 shows the recorded information on context relationships for the B2G information sharing system. The column with the support was left out to save space.

Name	Restriction on focus	Situation	Context elements
Do not share sensitive data	Businesses are not willing to participate.	The data in the flow is sensitive for a business to another business. The data is shared with the business that the data is sensitive to.	$isBusiness(BusinessA)$
			$isBusiness(BusinessB)$
			$isSensitiveTo\left(\begin{smallmatrix} Data, BusinessA, \\ BusinessB \end{smallmatrix}\right)$
			$isSharedWith(Data, BusinessB)$

Table 8: Recording information on context relationships for the example of the B2G information sharing system

When filling in the table shell, it could turn out that information is still missing. In that case, steps 1.3 and 1.2 should be alternated until all information that needs to be recorded in the table shell has been acquired.

6.2 Step 2: Determining the components needed to sense and adapt to context

In this step, we only need to look at the environment elements that are classified as context elements, because these are the environment elements that impact the focus and thus should be sensed and possibly manipulated by the system. We can thus discern two types of context elements, namely sensor elements and adaptor elements. In this step, we should determine which is which. The output of this step is a high-level, partial description of the architecture of the context-aware system that includes only the sensors and the adaptors, and their input and output and connections to the environment.

In the literature, there are several proposals for what the overall architecture of a context-aware system should look like (see e.g. (Baldauf et al., 2007; Saeed & Waheed, 2012; Schmohl & Baumgarten, 2008; Winograd, 2001)). It is not up to us to decide which one of them is best or which one the designer should choose. However, any architecture that a designer considers most apposite for the design of their system can be used in our method. This is possible, as all architectures for context-aware systems have some components and connections in common. The design choices we want to help the designer with only concern these common components and connections.

The similarities between the possible architectures are dictated by the nature of context-aware systems. Ultimately, a context-aware system should sense context information and adapt to it. Its architecture should thus always include sensors, adaptors and a component for reasoning with the information from the sensors to derive what adaptors should make what adaptation. This means that the sensors and adaptors should have some direct or indirect connection to the environment and to the reasoning component.

It is exactly these sensors, adaptors and connections to the environment that are common to the architectures that we want to support making design decisions on. We want to support designers in basing their design on insight into context. What sensors and adaptors should be included in the architecture is directly determined by what context the system should take into account. Furthermore, the same is the case for what things in the environment the sensors and adaptors should connect to directly or indirectly. Therefore, we only want to guide designers in making choices on this.

The functionalities that a system offers can be divided into basic functionalities and adaptive functionalities (Sitou & Spanfelner, 2007). The sensors and adaptors provide these adaptive functionalities. The architecture of the system should also include components for providing these basic functionalities. For example, for the tour guide system, the basic functionality is to provide information about sights. The architecture needs to include the components that can deliver that functionality, such as a database with information about sights and a screen to present the information. We assume that the system to provide the basic functionalities is already designed at the start of this step.

What functionalities belong to basic functionalities or to the adaptive functionalities is in part a design choice that a designer has to make. A designer has to make a choice on what goals they want to reach using context awareness. This choice should be based on whether they believe that reaching the goal requires providing different, or adaptive, functionality in different situations. Only after they make this choice, the proposed method plays a role. However, in case they base a focus on such a goal but they have a hard time finding context relationships for that focus, it is a sign that the way in which the goal is reached probably does not depend on context. In that case, they might go back on their choice.

Step 2 of the method can be divided into two sub-steps, namely 2.1) determine what adaptors are needed and 2.2) determine what sensors are needed.

6.2.1 Step 2.1: Determine what adaptors are needed

The overall process of performing step 2.1 is shown in figure 7. The input for this step is a list of context elements. The output is a list with descriptions of adaptors that can manipulate context elements.

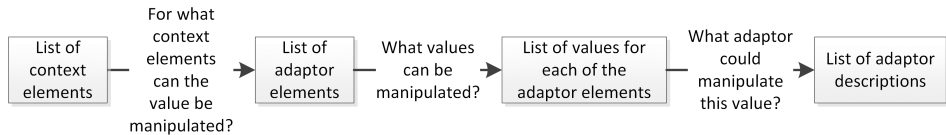


Figure 7: Determining the adaptors needed for a context-aware system

To complete this step, we need to identify for each context element whether it could and should be manipulated by the system. By manipulation, we mean that the system performs an action that changes the truth-value of the context element. This allows the system to adapt and change the situation, such that the value of its focus corresponds to its design goal. We call the component of the system that performs this action an adaptor. The input of the adaptor is a decision of the reasoning component on what value the adaptor needs to achieve for the context element. This value is expressed as a literal.

It is important that for each context relationship, at least one context element can be manipulated by the system. Otherwise, the system cannot adapt and it is not possible for the system to take into account the context. Of course, there can be more than one context element in a context relationship that can be manipulated. However, whether we need more than one context element to be manipulated depends on the type of context relationship.

Context relationships describe in what situations a focus is restricted to a certain value (e.g. the information provided is not relevant). The design goal of the designer is to ensure that the focus has a certain value (e.g. the information provided is relevant). On the basis of this, we can distinguish two types of context relationships, namely negative and positive context relationships. A positive context relationship restricts the focus to a value that conforms to the design goal of the designer. A negative context relationship restricts the focus to a value that does not conform to the design goal of the designer.

For example, the restriction on the focus such that the information provided to the user is relevant in a situation in which the user is within 150 metres of the sight and the information provided is about the sight, is a positive context relationship for the tour guide example (see table 7). The restriction on the focus to ‘not willing to participate’ in a situation in which the data in the flow is sensitive for a business to another business and this data is shared with the business that the data is sensitive to, is a negative context relationship for the example of the B2G information sharing system (see table 8).

For a positive context relationship, we need to ensure that all its environment elements have the value specified. This means that the system should contain adaptors for each context element for which this is possible. For a negative context relationship, we only need to ensure that at least one context element has a value different from that specified. This means that it is sufficient for the designer to choose one context element that should be manipulated and that the system should contain this adaptor.

In principle, it could also be possible to manipulate multiple context elements of a negative context relationship. However, for negative context relationships, this causes these manipulations to have a disjunctive relationship with each other. Either one can be performed to ensure that the focus is not negative. This offers the advantage of having multiple options for manipulation. However, it also leads to complications, as it introduces a form of nondeterminism. It thus requires a more complex reasoning mechanism to deal with this, which might not weigh against the advantages of having multiple options. Therefore, we choose one context element to manipulate for negative context relationships.

What context element to manipulate often will be an obvious choice. It is often clear what context elements cannot be manipulated because it is not possible or it is undesirable or too costly. These options can be eliminated. The remaining context elements are then the adaptor elements.

Example 15 (step 2.1, tour guide):

It would not be possible for the tour guide system to change the location of the Pyramid of Cheops. It might be undesirable to tell users to go to another location to ensure that the information they receive is relevant. Manipulating what information is provided to the user clearly is feasible and desirable for the tour guide.

Example 16 (step 2.1, B2G information sharing system):

It is not possible to manipulate the sensitivity of the information in the system. However, it is possible to manipulate with whom the data is shared in the information flow.

To determine what adaptors the context-aware system requires, the designer needs to determine first how the value of an adaptor element could be manipulated to achieve the value that is appropriate for the situation. For this, the designer needs to look at the terms of the environment element and see what needs to be changed, and then determine what components the system requires to perform the appropriate manipulations. As there might

be several possibilities, for which it is easier or harder to find a component that can perform them, the designer might need to alternate between the selection of possible manipulations and finding accompanying components, before an appropriate component is found. Each component should be described and incorporated in the overall architecture of the context-aware system.

Example 17 (step 2.1, tour guide):

In the tour guide example, we found a context element in which there is a relationship between a user and information, such that the user is provided with the information. This is represented in table 7 as *isProvidedTo(I, U)*. The context relationship is positive, so if *isProvidedTo(I, U)* has the value ‘false’, it should be manipulated such that it has the value ‘true’. The system cannot influence who the user is. However, it would not be hard to make the system manipulate what information a user is provided with. To do so, a component should perform the actions of finding information and providing it to the user. We do not need a very complex component to do this in this case. Let us assume that the architecture providing the basic functionality of the tour guide includes a screen to show users information about sights. It also includes a database with information about sights. The reasoning component will decide what information the user should be provided with and thus how to change what *I* is instantiated with. The required instantiation for *I* is provided to the adaptor. All the adaptor needs to do is search the database for the appropriate information and send this information to the screen.

Example 18 (step 2.1, B2G information sharing system):

The adaptor element for this example is represented as *isSharedWith(Data, BusinessB)* in table 8. Because the context relationship is negative, the system needs to adapt such that $\neg isSharedWith(Data, BusinessB)$. In other words, the adaptor should ensure that the data is not shared with business B. There are several ways in which the sharing of data can be prevented, for instance by blocking the sending of the data to other parties. However, in our case, the B2G information sharing system shares information using a distributed ledger. This means that if data is added to the ledger, all parties get a copy. In such a design, to ensure that the data is not shared with business B, the data can be encrypted and business B is not provided with a key.

To make such a manipulation possible, two components are needed, namely a component that encrypts data and a component that controls access by providing or not providing a key.

The adaptor components need to connect directly or indirectly to all objects connected in the context element to be able to manipulate it, and to the reasoning component to get the input necessary for it to know what actions to perform. Thus, in the tour guide example,

it needs to connect in some way to the information and to the user. For the information, this is done by searching the database with the information about sights. For the user, this connection is less direct and is done via another component, namely the screen of the system. In the example of the B2G information sharing system, the adaptor components need to connect to the data, which is done by encrypting it, and to the businesses, which is done by the access control component.

6.2.2 Step 2.2: Determine what sensors are needed

The overall process of performing step 2.2 is shown in figure 8. The input for this step is a list of context elements. The output is a list with descriptions of sensors that can sense the context elements.

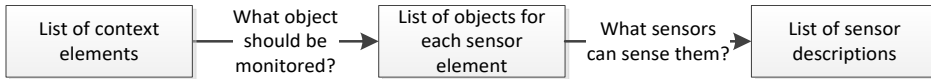


Figure 8: Determining the sensors needed for a context-aware system

To complete this step, we need to identify for each of the context elements that are not adaptor elements how it can be determined whether they are true in a situation. First, a decision should be made on what object in the world will need to be monitored. Then a measurement for establishing whether the object has a certain relationship should be found. Subsequently, it needs to be determined what component could carry out the measurement. Just as in the case of the adaptors, this might require some alternations between identifying the object to monitor, identifying possible measurements and finding an appropriate sensor. The last step is to determine what connections the sensor should have to the environment.

Each environment element, according to definition 17, connects things in the environment. For a sensor to be able to monitor, it should monitor a physical object in the environment. The object that the sensor should monitor should be one of the objects in the context element. If there are multiple objects that could be monitored to obtain the same information, then the designer should choose which ones to monitor. The object that is the most appropriate to monitor will be different for different context elements and might be influenced by the available techniques and by practical limitations.

Example 19 (step 2.2, tour guide):

We identified the location of the user, represented as *hasLocation*(U, L_1), as a context element. Its two arguments refer to objects, viz. the user and a location. A designer could choose to monitor the user and determine their location. Alternatively, they could monitor a location and establish what users are there. However, this could harm the privacy of the people that do not use the tour guide and the former option thus is more suitable.

Example 20 (step 2.2, B2G information sharing system):

The sensitivity of data from business A to business B is represented as *isSensitiveTo(Data, BusinessA, BusinessB)* .. Data is not inherently sensitive in this case, so it is not useful to monitor the data. *BusinessB* might know what data is sensitive, but it is not in their interest to open up about this. However, *BusinessA* will know exactly what data they do not want to share with what other businesses because of its sensitivity. Therefore, the context-aware system should monitor businesses to establish what data they consider sensitive from what other parties.

A designer who wants to improve the accuracy of the context information might choose to monitor more than one type of object for a context element; for instance, they might monitor both the user and the location. However, it is important to note that this will come at additional costs, as it requires the addition of some kind of conflict resolution for dealing with contradictory sensor information.

Once an object to monitor has been chosen, the designer needs to decide how to measure what it is connected to according to the relationship in the context element. It is important to be aware that it is often useful to use the same type of measurement to measure things that are similar. In the tour guide example, it would make sense to use the same measurements for the location of the user and for the location of the sight.

The next step is identifying what kind of sensor components could provide the measurement and how this could be done. The designer should provide a description of these sensor components and include them in the overall architecture. Like the adaptor components, the sensor components belonging to a sensor element should connect to all objects in that element.

Example 21 (step 2.2, tour guide):

For the location context element, it was decided to monitor the user. We now need to find a measurement for the location of the user. We could choose coordinates for this, as this is quite a common measurement for location. A very common way to determine coordinates is to use a GPS sensor. This would thus be an appropriate sensor to measure the location of the user. To connect to the user, the GPS sensor should be placed on something that the user carries with them; this could be the tour guide itself. To connect to the location, the sensor performs its measurement.

Example 22 (step 2.2, B2G information sharing system):

For the sensitivity context element, it was decided to monitor the business that thinks that the data is sensitive. We thus need to find a measurement for what data they consider sensitive, and from what businesses. For the businesses, we can assign IDs to the data that is shared and we can name businesses. The most obvious way to sense what data is sensitive and to what businesses, is to ask the business that thinks the data is sensitive. A sensor should thus request this data from businesses. In the case of the B2G

information sharing system, the component simply requests the information from businesses when new data is shared.

In section 5.4, we discussed that in some cases relationships can be included that are not context elements but express constraints. These relationships do not change over time. This means that to ‘sense’ them, no external connections need to be made. Instead, this information needs to be stored somewhere in the system itself or it needs to be calculated. For instance, for the distance between locations, the ‘sensor’ could be a component that calculates the distance between two coordinates.

6.3 Step 3: Determining the rules for reasoning with context information

The overall process of performing step 3 is shown in figure 9. The inputs of this step are the outputs of steps 1 and 2, namely a list of context relationships and a list of adaptors and sensors. The output of this step is a list of context rules that the context-aware system can use to derive what adaptations to make in different situations.

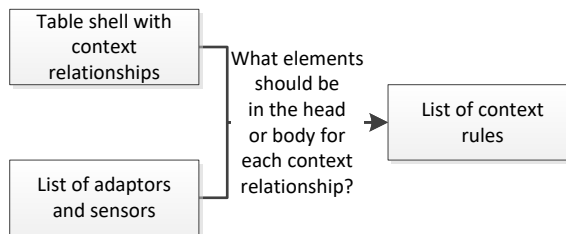


Figure 9: Determining the rules for reasoning with context information for a context-aware system

The input of the reasoning component of a context-aware system comprises information gathered by the sensors. They are expressed as ground literals in a logic program. This might require the raw data of the sensor elements to be translated into these literals by middleware. There are ample descriptions in the literature of how middleware can be used to process raw context information (for an overview, see e.g. (2007)).

The outputs of the reasoning component also are ground literals. They express the value that the adaptor elements should have to ensure the appropriate value of the focus. They are input for the adaptors and can be viewed as commands to perform an action that results in the adaptor element being true. For this, the middleware between the reasoning component and the adaptors themselves should include a mapping between literals and the actions of adaptors.

A context rule is a rule that expresses that the system needs to perform a manipulation to make the environment element in its header true in the situation that the environment elements in its body are true. The syntax of a context rule can be found in definition 12. Examples of context-rules can be found in example 2.

The context relationships provide information on what manipulations we want to perform in what situations. As discussed, for the positive context relationships, we

want the situation described for them to exist, and for the negative context relationships, we want the situation described for them not to exist. Based on this principle, we already established what context elements should be manipulated in step 2.1.

We can translate each positive context relationship into a context rule where the head is a schematic literal representing the required value of one of its adaptor elements, and the body is the set of all schematic literals representing the values of its relationships that are not adaptor elements. The number of different rules there are for a positive context relationship is the number of adaptor elements it has. A designer should derive all possible rules for each positive context relationship in this way.

Example 23 (step 3, tour guide):

The context relationship in table 7 can be translated into the following context rule:

isProvidedTo(I, U) ←
isUser(U),
subjectOf(S, I),
hasLocation(U, L₁),
hasLocation(S, L₂),
distance(L₁, L₂, Distance),
Distance < 150

We cannot translate this context relationship into any other rules, as it has only one adaptor element.

We can translate every negative context relationship into a single rule, where the body is again the set of all literals representing values for context elements in its situation that are not adaptor elements. As negative context relationships have only one adaptor element, and we do not want the situation in the negative context relationships to exist, the head of the rule is the negation of the schematic literal representing the value of the adaptor element in the situation described in the context relationship. The negation is that of classical logic, in the sense that the double negation of a literal is equivalent to the literal.

Example 24 (step 3, B2G information sharing system):

The context relationship in table 8 can be translated into the following context rule:

¬isSharedWith(Data, BusinessB) ←
isBusiness(BusinessA),
isBusiness(BusinessB),
isSensitiveTo(Data, BusinessA, BusinessB).

To complete step 3, the designer should generate all context rules in this way. These context rules can then be reasoned with in a logic program such as described by Lifschitz (1996), together with the context elements that are input for the reasoning component. From the logic program, it can be derived what manipulations the system should perform in a certain situation. Because we used the logic-programming paradigm for our

formalisation, the context rules and context elements are in the appropriate format to be part of the logic program.

We will illustrate the workings of such a logic program based on the simplified logic program in example 4 (p. 79). The ground literals in the logic program express the context information that the program receives from its sensors as input. Context information expressed in the example is that *mary* is a user and that her location is 29°58'45.03"N 31°08'03.69"E. Only if the situation is what it should be according to the body of a context rule, the head of the rule can be derived and a command to perform an action is provided. In this way, the logic program is used to derive the actions that need to be performed by the adaptors of the system based on context information.

In this example, it can be derived that the location of *mary* is less than 150 metres from the subject of *info*, namely *pyramidOfCheops*, by substituting *U* for *mary*, and *L₁* for 29°58'45.03"N 31°08'03.69"E, and so on. This means that the head of the rule can be derived and a command is provided to an adaptor to provide *info* to *mary*.

Of course, the ground literals in the program can be derived as well. However, they are already true, so no action needs to be performed to make them true. It would be easy to let them be filtered out by the middleware of the system by comparing the input and the output of the reasoning component.

Furthermore, the rule provided here includes some restrictions that are not context elements and that cannot be sensed or manipulated. This is, for example, the case for *distance(L₁, L₂, Dist)*. Evaluation of whether these literals are true can be easily done by adding some standard ground literals that never change (e.g. for *hasLocation(pyramidOfCheops, 29°58'27.00N 31°08'2.21E)*), or by calculating them using functionality that is usually build-in in a programming language (e.g. for *Dist < 150*).

In general, logic programs will be quite simple, because only a single rule, rather than a sequence of rules, is needed to derive a manipulation (not taking into account calculating the restrictions). However, what specific variant of logical program and corresponding reasoning mechanism will be chosen is up to the designer as there are a variety of practical factors that might play a role. For instance, a context-aware system that should respond to changes in the environment very fast and that contains only a couple of rules, could benefit from a reasoning mechanism that derives all manipulations in a bottom-up approach each time new sensor information is available. On the other hand, the designer of a system that contains a lot of rules and facts might prefer a top-down approach in which the adaptors periodically query the system for the next manipulation that they need to perform. Alternatively, it could be possible to let rules be triggered only when certain events happen, such as is the case for Event-Condition-Action (ECA) rules (Cano, Delaval, & Rutten, 2014). Furthermore, whether negation as failure is enough in the body of rules could depend on the required evidence based on which a manipulation needs to be derived.

It is important to note that it is possible that multiple values for environment elements for the same object are derived from the rules. Adaptors will not always be equipped to cope with these multiple values simultaneously (e.g. screen off and on at the

same time). Furthermore, if the heads of the rules contain classical negation, it is possible to derive contradictions that no adaptor could conform to. For instance, when based on one rule *isProvidedTo(info, mary)* is derived and when based on another rule \neg *isProvidedTo(info, mary)* is derived. It is not possible to provide *info* to Mary and not provide *info* to Mary at the same time.

The reason for these issues is that for the more complex cases, the insight into context from step 1 is typically incomplete. In section 6.1.2, we described that in practice it is not feasible to establish what comprises the complete set of environment elements that have a certain context relationship with the focus. This is not an issue specific to the method or definitions we propose; rather, it is a fundamental issue when investigating complex environments in the real world. There are just too many variables to take into account. In fact, defeasible logics were developed to deal with this same issue. Defeasible logic programs, for example, as described by García and Simari (2004), could also be used to reason with the rules.

Alternatively, this issue can be dealt with in the middleware between the adaptors and the reasoning mechanism. There are many ways to solve this, and again, the best way depends on many practical considerations. In some cases, it might be useful, for instance, to ask the user to choose between alternative manipulations. In other cases, it might be better not to bother the user with this and to implement an algorithm or component that makes a choice between alternative manipulations.

For other possible solutions, we can turn to the work of Shishkov (2019). He argues that in some complex cases, such as when the human mind is of concern, choices can be based on Bayesian modelling or semiotic norms. Bayesian modelling and machine learning could be useful when the context-aware system needs to predict the situation of the user (Shishkov, 2019). The Norm Analysis Method from organisational semiotics, as described by Kecheng (2000), could be useful in the case the behaviour of entities in an organisation is of concern, as it is possible to take into account deontic operators, such as ‘is permitted’ or ‘is obliged’ (Shishkov, 2019).

**PART III: A CONTEXT-AWARE ARCHITECTURE FOR B2G
INFORMATION SHARING IN THE CONTAINER-SHIPPING DOMAIN**

7 B2G information sharing in the international container-shipping domain

In section 1.3, we describe our research problem, namely, that there is a lack of knowledge on what the design of context-aware architectures that support business-to-government information sharing in complex environments should look like. To understand the problem better, we studied B2G information sharing in the international container-shipping domain. The international container-shipping domain provides a typical instance of a complex environment and the related issues with information sharing discussed in section 1.1. This allows us to get a deeper insight into the problem and it provides a more practical illustration of the complexities involved and the resulting need for a context-aware architecture to support information sharing. Furthermore, there is ample literature on this subject in this domain that we can base our description on.

We focus on the most important factors that increase complexity that are described in the literature. To keep the research feasible, we limit our scope to import of goods into the European Union (EU). In addition, the research focuses on information sharing with customs organisations on the import side and not on sharing with other government organisations.

We start this section with a discussion of the benefits of supporting additional B2G information sharing in international container shipping that motivates this research. In addition, we discuss the risks of such information sharing, which should be taken into account when supporting it. In section 7.2, we discuss the complexities of the environment in which information is shared in the container-shipping domain. In section 7.3, we then provide an overview of related work. In section 7.4, we discuss the need for a new context-aware architecture. For developing the context-aware architecture, we use the method presented in chapter 6. This method starts with deriving the foci of the context from the problem specification. In section 7.5, we present our foci.

Parts of this chapter have been published in van Engelenburg, Janssen and Klievink (2015), van Engelenburg, Janssen and Klievink (2017), van Engelenburg, Janssen and Klievink (2018), van Engelenburg, Janssen, Klievink, Rukanova and Tan (2018), Tan, Rukanova, van Engelenburg, Janssen and Ubacht (n.d.), and van Engelenburg, Janssen and Klievink (n.d.).

7.1 Possible benefits and risks of B2G information sharing in container shipping

In this section, we discuss the possible benefits and risks of supporting B2G information sharing in the container-shipping domain. Monitoring the goods flow and determining that businesses are compliant with rules and regulations is a central task of customs. We discuss the notion of compliance and the role of customs in section 7.1.1. In section 7.1.2, we discuss how B2G information sharing, in addition to the sharing of required declarations, could help to improve compliance. The immediate goal for this research is to support such additional B2G information sharing. In section 7.1.3, we discuss the incentives for businesses to participate in such additional information sharing. In section

7.1.4, we discuss the risks of additional B2G information sharing. To support additional B2G information sharing, these risks need to be taken into account.

7.1.1 Compliance and the role of customs

There are different institutions and agencies at the European level and the national levels that deal with goods that are shipped via containers internationally. Jensen et al. (2014), for example, describe that the shipping of fresh fruit to Europe involves customs, phytosanitary authorities, health authorities, veterinary authorities, and a party that performs a scanning inspection. The authorities that are involved can vary, e.g., based on the type of goods that are shipped. In this research, we focus on customs, who are involved in monitoring all goods.

Customs' work is vital for society. For instance, according to the European Economic and Social Committee (2003), ships and maritime transport are vulnerable to terrorist risks. More specifically, terrorists or weapons of mass destruction could be smuggled in containers (European Economic and Social Committee, 2003). Disasters such as the explosion on the ship MSC Flaminia show the possible consequences if safety is not sufficiently ensured when dangerous goods are involved (see e.g., (The British Broadcasting Corporation, 2012)).

The European Union Customs Union is an area consisting of 28 countries (until Brexit) that have a uniform system for handling import, export and transit of goods (DG Taxation and Customs Union, 2018). Each of these countries implements the rules from the Union customs Code (UCC) (The European Parliament and the Council of the European Union, 2013). In this way, the national customs organisations of these countries act as one (DG Taxation and Customs Union, 2018). The security analysis of goods that are shipped to the EU from outside the EU is performed by the national customs authority of the first port of call in the EU (Zomer, 2011).

The UCC provides a legal framework for the rules and procedures in the European Union (The European Commission, 2018). According to the UCC (2013), "*Customs authorities shall put in place measures aimed, in particular, at the following:*

- (a). protecting the financial interests of the Union and its Member States;*
- (b). protecting the Union from unfair and illegal trade while supporting legitimate business activity;*
- (c). ensuring the security and safety of the Union and its residents, and the protection of the environment, where appropriate in close cooperation with other authorities; and*
- (d). maintaining a proper balance between customs controls and facilitation of legitimate trade."*

According to Silveira et al. (2012, p. 1), compliance can be defined as "*the conformance to a set of laws, regulations, policies, best practices, or service-level agreements*". Corresponding with the responsibilities of customs described above, customs should ensure that businesses involved in supply chains are compliant with the applicable fiscal rules and regulations and the rules and regulations concerning security, safety, health and environment.

The UCC also specifies that businesses should share certain documents with customs that customs can use for compliance monitoring (see section 7.2.2 for an overview). For example, according to the UCC, the carrier of goods should submit to customs an Entry Summary Declaration (ENS) describing the goods it is carrying (The Commission Of The European Communities, 2006b). The UCC also specifies how such documents should be submitted, for example, 24 hours before loading the goods at the port of departure and with a goods description with a certain level of specificity (The Commission Of The European Communities, 2006b). Businesses should comply with these rules as well.

7.1.2 Additional B2G information sharing and improving compliance

An important problem in container transport is that it is impossible to view directly the contents of containers. The number of containers is so high, that it is not possible even to come close to inspecting each container (Levinson, 2010). While customs cannot open all containers, they can perform risk assessment and target high-risk containers for inspection by opening it or performing a scan, such as an X-ray scan (Customs Administration of the Netherlands, 2014; Rukanova et al., 2011). As we discussed in the previous section, customs does receive information from businesses that they are required to share and customs can base their risk assessment on this. The ENS is the main source for the risk assessment done by customs (Zomer, 2011).

The strategy of some customs organisations is to develop new and non-disruptive ways of scanning all containers (Customs Administration of the Netherlands, 2014). The analysis of the scans will need to be automated as well (Customs Administration of the Netherlands, 2014). This means that they need to be compared with reference files that contain information about the goods (Customs Administration of the Netherlands, 2014). This strategy thus also relies on customs obtaining information on the goods in containers. Furthermore, scans might not fully reveal what is in a container.

The information in documents businesses are obligated to share with customs, such as the ENS, is often not timely, has been altered, is inaccurate, or is vague (Hesketh, 2010; Klievink & Lucassen, 2013; H. L. Lee & Whang, 2000). This means that to perform their duties, customs needs additional information on which to base their risk assessment, in addition to these documents (Hesketh, 2010). Fortunately, the information that businesses gather and share with each other is of high quality since their own commercial operations depend on it (Bharosa et al., 2013). Customs could use this data to base their risk assessment on. For instance, the manufacturer of goods that are transported has a lot of details about them, such as their weight (Hesketh, 2010). If customs determines that the weight of a container is unexpected based on the weight of the goods, this might be a reason for physical inspection.

If customs receives additional high-quality information to base their risk assessment on, they might be better able to identify containers that are high risk. Performing big data analysis on this is one of the options explored to improve the way in which variances in the regular goods flow are identified (Customs Administration of the Netherlands, 2014). If risk assessment by customs improves, then containers for which

there are compliance issues will more often be targeted for inspection. In that case, the chances that a business that is not compliant is caught will increase. If this happens, we expect businesses to be compliant more often as not being compliant entails a higher risk for them.

Improving compliance reduces safety and security risks to society and helps to maintain financial and social stability ((Power, 2007) as cited in (Bharosa et al., 2013)). In addition, if customs uses the information businesses share with them to do their work more efficiently, then this might lead to more thorough monitoring with the same or less capacity. This might mean a lower financial burden for society.

7.1.3 Incentives for businesses to share additional information

Customs organisations, such as Dutch Customs, are expected to contribute to the competitiveness of their country and the European Union, and to maintain a balance between controls and facilitating trade (Customs Administration of the Netherlands, 2014; The European Parliament and the Council of the European Union, 2013). Obligating businesses to share additional information with customs for compliance monitoring would increase their compliance costs and this would have a negative effect on this competitive position and balance (Bharosa et al., 2013). Businesses thus need to provide additional information to customs voluntarily.

The willingness of businesses to share additional information with customs can be improved by making it easier. One way to do this is by reducing the efforts required for businesses to share the information with customs. Additional data processing to share data with a separate information sharing system can be avoided when government organisations piggyback on the existing data flow between businesses (Rukanova et al., 2011). This way of reusing data could also result in an improvement of information quality and a reduction of transaction costs (Klievink, Janssen, et al., 2012).

To make piggybacking easy, a system can support business-to-business (B2B) as well as B2G information sharing. Businesses need incentives to participate in B2B information sharing as well as B2G information sharing that piggybacks on this B2B flow in such a system. The benefits of B2B information sharing in supply chains are well established in the literature. Data sharing and integration of supply chains are often considered to provide competitive advantage (Hertz & Alfredsson, 2003). Reliable shipping information allows businesses to work together more effectively and efficiently and it allows for synchro-modality to optimise the goods flow (Fawcett et al., 2007; Overbeek, Klievink, Hesketh, Heijmann, & Tan, 2011). B2B information sharing can improve cooperation by aligning production and delivery schedules (H. L. Lee & Whang, 2000). Furthermore, the sharing of data on performance measures can help to identify bottlenecks and improve overall performance (H. L. Lee & Whang, 2000).

One example of the benefit of B2B information sharing to businesses is that it can help to reduce the bullwhip effect. The bullwhip effect is the effect of the amplification of the demand in the supply chain when there is no good overview of the demand expected in the supply chain and the information on this lags behind (H. L. Lee, Padmanabhan, & Whang, 1997a, 1997b). This results in businesses unnecessarily having duplicate safety inventory, excessive production, non-optimal scheduling of production

and large warehouses (H. L. Lee & Whang, 2000). The risks of having stock are not only that it needs capital, but also that stock might become obsolete. Numerous studies show that addressing the bullwhip effect requires better information sharing about demand between supply chain partners (Bray & Mendelson, 2012).

Businesses might benefit from B2G information sharing as well. If customs has sufficient confidence in the internal control systems and control mechanisms of a business involved in customs-related activities (e.g., a carrier or freight forwarder), they can grant them the status of Authorised Economic Operator (AEO) (Customs Administration of the Netherlands, 2014; The European Commission Directorate-General Taxation and Customs Union, 2016). Businesses with AEO status have to meet certain criteria, including criteria for information sharing with customs, and in return, they receive several benefits (The European Commission Directorate-General Taxation and Customs Union, 2016).

According to Article 39 of Regulation (EU) No 952/2013 (Union Customs Code) (The European Parliament and the Council of the European Union, 2013) the criteria that businesses need to meet in order to be certified as an AEO includes *“the demonstration by the applicant of a high level of control of his or her operations and of the flow of goods, by means of a system of managing commercial and, where appropriate, transport records, which allows appropriate customs controls”*. There are various conditions that businesses need to fulfil to meet this and other criteria. One of them is that the business allows customs access to their accounting systems and their commercial and transport records (The European Commission, 2015).

One of the benefits that businesses with AEO status receive is easier admittance to customs simplification (The European Commission Directorate-General Taxation and Customs Union, 2016). For example, businesses might be authorised to lodge a simplified customs declaration and the obligation to present goods to customs might be waived (The European Commission Directorate-General Taxation and Customs Union, 2016; The European Parliament and the Council of the European Union, 2013). Another benefit is that customs might provide AEOs with a notification prior to the inspection of a container (The European Commission Directorate-General Taxation and Customs Union, 2016). Furthermore, containers of AEOs might get fewer physical and document-based controls and if they do, they will get priority treatment and they can choose the location where the control takes place (The European Commission Directorate-General Taxation and Customs Union, 2016).

According to the European Commission Directorate-General Taxation and Customs Union (2016), there are several indirect advantages for businesses that result from an AEO status, viz., businesses get recognised as a secure and safe business partner, and businesses improve their relationships with customs and other government authorities. In addition, they claim that the standards that businesses need to meet in order to get AEO status has indirect benefits like improved planning, reduced safety and security incidents and fewer delayed shipments (The European Commission Directorate-General Taxation and Customs Union, 2016).

	Blue flow	Green flow	Yellow flow
Flow contains	- Goods from unknown traders	- Goods from known and reliable traders (with AEO status)	- Goods from the green flow - Goods from the blue flow that are consolidated by a yellow certified logistic service provider in a trusted trade lane
Monitoring by Customs	- Many physical and administrative inspections - Inspections based on risk analysis	- System-based supervision by monitoring whether businesses comply with their (approved) internal control procedures - Random physical and administrative controls	- Securing the entire trade lane by acquiring high-quality data to be cognisant of every link in the entire trade lane and by securing the physical integrity of the goods in the trade lane
Intervention in the goods flow	- Intervention with the flow of goods at the border	- Preference for checks that are less or not disruptive to the flow of goods	- One inspection at the time the goods are loaded
Aim of Customs	- Making this flow narrower	- Making this flow wider	- Researching how this flow can be given shape

Table 9: The blue, green and yellow flow of goods in the enforcement vision of Customs (Customs Administration of the Netherlands, 2014)

Some customs organisations of individual countries in the EU formulate their own strategies for risk assessment that even go further than formulated by the European Commission. For example, Dutch customs envisions supervising 100% of the transports of the goods that cross the borders, by scanning all containers and by determining for each transport whether the required notifications and declarations have been filed (Customs Administration of the Netherlands, 2014). They want to use this information and information from other sources to target transports for physical and administrative inspections (Customs Administration of the Netherlands, 2014). In addition, Dutch Customs wants to separate the flow of goods in a blue, green and yellow flow based on the quality of the information they receive from businesses and their trust in those businesses (see table 9) (Customs Administration of the Netherlands, 2014). For this enforcement vision, Dutch customs requires additional information from businesses to supervise the goods flow and to perform targeted inspections and they provided benefits for this additional information sharing (Customs Administration of the Netherlands, 2014).

In the European Union, customs organisations thus can reward businesses that share additional information with them and they can rely on this for their enforcement strategy. These rewards can provide businesses with benefits of B2G information sharing and thus provides them with an incentive to share additional information with customs.

To summarise, there are some clear benefits of supporting additional B2G information sharing for customs and society, which is the motivation to support such information sharing in this domain. Furthermore, there are some clear incentives for businesses to share additional data with customs. This makes it feasible to support B2G sharing of additional information in this domain. However, there are some obstacles to supporting this kind of information sharing as well, including the risks discussed in the next subsection.

7.1.4 Risks of information sharing in the container-shipping domain

Both B2G and B2B information sharing is associated with risks for society, businesses and customs. These risks need to be dealt with to effectively support B2G information sharing in international container shipping and to reap its benefits.

Concerning B2B information sharing, often it is difficult to align the incentives for information sharing of different businesses (H. L. Lee & Whang, 2000). For competitive reasons (e.g. fear of being bypassed in the chain) or security reasons (e.g. sharing information on high-value goods), businesses may be hesitant to share information with others (Fawcett et al., 2007; Klievink, Janssen, et al., 2012). Furthermore, parties may perceive a higher vulnerability to misuse or opportunism by the partners they share data with (Hart & Saunders, 1997; H. L. Lee & Whang, 2000). Cost data, such as production yield data and purchase prices of parts are often not shared for this reason (H. L. Lee & Whang, 2000).

In addition, businesses can also withhold information if they benefit from a position in which they have superior information (H. L. Lee & Whang, 2000). Sharing information, in that case, would pose the risk of losing this benefit. Often, they will make agreements with other parties that they cannot share their data with others for this reason (H. L. Lee & Whang, 2000). Businesses might also view it as a risk that others, especially competitors, will try to manipulate them and provide incorrect inventory data on purpose (H. L. Lee & Whang, 2000).

The sharing of additional information can in some cases result in liability for businesses that they want to avoid. For instance, the security analysis of goods that are shipped to the EU from outside the EU is performed by the national customs authority of the first port of call in the EU (Zomer, 2011). Customs organisations base this security analysis mainly on the ENS. This declaration has to be filed by the carrier 24 hours before the goods are loaded at the vessel at the port of departure (Jensen, Bjørn-Andersen, & Vatrapu, 2014; Jensen & Vatrapu, 2015; The Commission Of The European Communities, 2006b; Zomer, 2011). According to EU Regulation 1875/2006 of the European customs code (The Commission Of The European Communities, 2006b, p. 114), the ENS should contain a description of the goods that *“is precise enough for customs services to be able to identify the goods.”*

According to the Hague-Visby Rules, the liability of the carrier is limited to a certain amount per package, unless the value and a full description of the goods are declared by the shipper and included in the bill of lading (Hesketh, 2010). Were the carrier liable for the full costs, the shipping rates would increase (Hesketh, 2010). To prevent this, the shipper omits the value of the goods and makes the goods description vague (Hesketh, 2010). Since the information in the ENS is based on the information in the bill of lading, the goods description in the ENS is vague as well, even though it was originally intended to be precise enough to identify the goods. In practice, the carrier thus avoids having the detailed goods description because it might increase their liability. Of course, they cannot share information they do not have.

In addition, there is a risk that the sharing of information is unlawful. Legal considerations make information sharing between businesses in a supply chain and customs a highly complex process (Karampetsou, 2016). Different legal frameworks are applicable to different categories of data, for example, personal or impersonal, or confidential or public data (Karampetsou, 2016). With whom data can be legally shared depends on the country in which goods are moving (van Stijn et al., 2011), and different sources of law, such as national and European law, might be applicable. In cases where information sharing usually is unlawful, exceptions might be made on grounds of information sharing being vital to safety and security (Janssen & Smeele, 2013). Moreover, legislation may change frequently (Gong & Janssen, 2014). This complexity makes it difficult for parties to determine whether information sharing is lawful, which poses the risk of them sharing information unlawfully.

Unlawful information sharing risks punishment or liability for the business that shares the information. In addition, laws and legislation exist to protect society. Unlawful information sharing, therefore, might harm society as it could violate the rights of people and organisations in society. Examples of this are violations of the fundamental right to respect for private and family life for natural persons or unfair competition.

If a lot of information is shared with a single business or customs organisation, then they are provided with a lot of power. Such surveillance on a large scale might result in big brother issues and it might pave the way for a surveillance state (Jensen & Tan, 2015). This poses risks for society as well as businesses.

Customs	Society	Businesses
<ul style="list-style-type: none"> - Obtaining information unlawfully 	<ul style="list-style-type: none"> - Unlawful information sharing can violate the rights of people and organisations. - Surveillance on a large scale 	<ul style="list-style-type: none"> - Losing their competitive position or superior information position - Compromising the security of their goods - Misuse, manipulation or opportunism by other parties - Increased liability - Sharing information unlawfully - Unlawful information sharing can violate the rights of businesses. - Surveillance on a large scale

Table 10: The possible risks of information sharing in international container shipping

7.2 The complex environment of B2G information sharing in container shipping

The introduction of containers in maritime transport has had a profound impact on the world economy (Levinson, 2010). Containerisation makes shipping goods cheaper and faster, and it allows goods to be easily transported using different modalities (Levinson, 2010; Stopford, 2009). This makes it more often worth it to ship goods all over the world, which advances globalisation (Levinson, 2010). The volume of goods that are shipped to the European Union is large. In 2016, the value of goods imported to the European Union was 1.71 trillion euro (DG Taxation and Customs Union, 2018). This means that the customs offices in the European Union had to handle almost 313 million customs declarations (DG Taxation and Customs Union, 2018). In 2015, 51% of the goods were imported into the European Union via sea (DG Taxation and Customs Union, 2018).

This development, together with the developments in ICT, has significantly increased the volume as well as the variety of data that is shared between businesses and with customs. First, we discuss the parties that are involved in the information sharing process in this domain, and in what ways they vary. Then, we discuss other complicating factors, such as the different types of information that are shared and the different systems and infrastructures involved in information sharing. In section 7.4, we discuss why the complexity of this environment requires information sharing to be supported by a context-aware architecture.

7.2.1 Businesses involved in information sharing

A supply chain in container shipping can be viewed as a complex network, consisting of various stakeholders (van Baalen, Zuidwijk, & van Nunen, 2009). Mentzer et al. (2001) describe several different definitions of a supply chain. According to La Londe and

Masters (1994), a supply chain consists of businesses passing materials forward. A very straightforward definition of ‘supply chain’ is provided by Tsay et al. (1999). We use this definition as the definition of ‘supply chain’ for our research. However, for the sake of consistency, we use the term businesses instead of parties. In concordance with definition 21, besides a flow of goods, a supply chain also has an informational and financial flow (H. L. Lee & Whang, 2000; Tsay et al., 1999).

Definition 21 (supply chain):

A supply chain is two or more businesses linked by a flow of goods, information and funds. (adapted from (Tsay et al., 1999, p. 301)).

Structurally, a supply chain only looks like a chain when considering a specific customer or consumer (Cooper, Ellram, Gardner, & Hanks, 1997). This is often how supply chains are portrayed in literature. However, in reality, it is a massive set of tangled branches (Cooper et al., 1997). A supply chain also can be viewed as consisting of different tiers or layers, where horizontally there is a certain number of tiers and vertically there is a certain number of businesses per tier (Min & Zhou, 2002). The horizontal structure determines the length of the supply chain (Lambert & Cooper, 2000). An example of a short supply chain is that of bulk cement, as it only involves obtaining the raw materials, combining these materials, and moving them to the appropriate location where they are used for constructing buildings (Lambert & Cooper, 2000). A longer supply would have more tiers and involve, for example, different businesses that further process the raw materials and use them to produce different types of goods. The vertical structure determines the wideness of the supply chain, where a supply chain with many businesses per tier is considered wide and a supply chain with a few businesses per tier is considered narrow (Lambert & Cooper, 2000).

Some of the work on supply chains have a broad view that includes the producers of raw materials as well as the end-consumers of goods (see e.g. (Lambert, Cooper, & Pagh, 1998; Min & Zhou, 2002; Yu, Yan, & Cheng, 2001)). Other work has a more narrow view on supply chains and investigate them starting with a parts manufacturer and ending with a retailer, or starting with a seller and ending with a buyer (see e.g. (Jensen & Vratapu, 2015; H. L. Lee & Whang, 2000; Pruksasri, van den Berg, & Hofman, 2014)). Businesses in a supply chain can be classified as primary or secondary partners (Lambert et al., 1998). Primary partners perform operational or managerial activities and include manufacturers and retailers (Min & Zhou, 2002). Secondary partners support these activities by providing resources, knowledge and utility and include third-party logistics providers and IT service providers (Min & Zhou, 2002).

If we take the broad view, then a supply chain starts with the businesses that produce or harvest raw materials, such as miners and farmers. These can then be passed forward to manufacturers that make components from it and manufacture a product. When the raw goods are already suitable for consumption, such as flowers or fruit, then further processing might not be necessary and the supply chain will be shorter.

Several businesses are involved in transferring goods from one place to the other. The carrier is the party that is physically transporting the goods (van Stijn et al., 2011).

However, from a juridical point of view, the carrier is the party that is responsible for transporting the goods according to a contract of carriage (The European Commission Directorate-General Taxation and Customs Union, 2016). In the case of maritime transport, a maritime performing party can perform the carrier's obligations, and the owner of the ship on which the goods are transported and the carrier mentioned on the bill of lading are not necessarily the same (Smeele, 2010).

As we focus on international container shipping, the ocean carrier is an important party that is involved in the supply chain. In addition, the goods need to be transported from and to the seaports of departure and arrival. This can be done by truck or by inland shipping, for example, making a trucking company or inland carrier part of the supply chain as well.

At the ports, there are several businesses involved with shipping the goods. If a consignment is less than a container load, then a consolidator might be involved that consolidates different consignments in a container. In addition, port terminal operators and port services handle the containers at the terminal in the ports. Stevedores are responsible for loading and discharging containers. At the other end of the supply chain, there are wholesalers and retailers. These businesses sell the goods to other businesses and eventually to the end-consumer.

Parties might have various roles in the supply chain. The goods within a supply chain might be sold and bought by several parties. The producers of the raw materials and the manufacturers will usually be sellers of the goods. The manufacturers might be buyers of the raw materials or components for their products as well. The buyers of the goods are usually the retailers and the end consumers. In between these ends of the supply chain, various traders might buy the goods only with the aim of reselling them.

The consignor is the party that sends the goods and thus sometimes is also referred to as the sender or the shipper (Hesketh, 2010; van Stijn et al., 2011). The consignee is the party that receives goods. Some work does not make a clear distinction between the consignor and the seller, as well as the consignee and the buyer, or they consider the consignee to be the customer of the consignor (see e.g., (Harris, Wang, & Wang, 2015; Overbeek, Janssen, & Tan, 2012)). However, the consignor does not always correspond to a seller and the consignee does not always correspond to a buyer (Hesketh, 2010; van Stijn et al., 2011). For example, a seller might employ a freight forwarder in one country who sends the goods to a freight forwarder in another country (Hesketh, 2010).

Transport documents often only mention agents instead of the true buyer and seller or the true consignor and consignee (Hesketh, 2010; Jensen, 2017; Zomer, de Putter, et al., 2014). For example, in the ENS freight forwarders might be identified as consignor and consignee (Hesketh, 2010; Jensen, 2017). Not being able to identify the true consignor and true consignee can interfere with the effectiveness of risk assessment performed by customs (Zomer, de Putter, et al., 2014).

In addition, there are exporters and importers of goods. In this work, we focus on import into the EU. The importer of goods is the party that makes an import declaration with customs or they can be viewed as the party that places the goods in the market in the

EU (The European Commission Directorate-General Taxation and Customs Union, 2016).

Freight forwarders are responsible for planning, arranging and optimising the transport of the goods on behalf of another party (Chow, Choy, & Lee, 2007; The European Commission Directorate-General Taxation and Customs Union, 2016). They can provide additional services such as storage, packing and unpacking of goods, insurance, tracking and tracing and perform customs formalities (Verhagen, 2017). In some cases, the freight forwarder assumes responsibility for the carriage of goods and is considered a carrier from a juridical point of view (Smeele, 2016).

Different kinds of brokers and shipping agents that act on behalf of other parties can be involved in the supply chain. Some of the additional services a freight forwarder can provide correspond with the services provided by a customs broker, namely preparing and submitting the necessary documents to get containers cleared. In addition, an information or message broker might be involved, that can be responsible for systems such as a Port Community System (PCS) or a Business Community System (BCS) that provide connectivity to businesses and supports the reuse of data (van Baalen et al., 2009). A service broker is a trusted party that forces service providers to meet certain information practices (Papazoglou & van den Heuvel, 2007). A shipbroker is an intermediary that brings together charterers of ships and ship owners (Pisanias & Willcocks, 1999).

A third-party logistics provider is an external party that manages, controls and provides logistics activities on behalf of a shipper (Hertz & Alfredsson, 2003). According to Hertz and Alfredsson (2003, p. 140), typical services provided by third-party logistics providers include “*transport, warehousing, inventory, value-added services, information services and design, and reengineering of the chain*”. Third parties might also be involved in inspecting or surveying cargo. Accordingly, several of the parties mentioned above, such as a terminal operator, can be classified as third party logistics providers in some cases.

In addition, there are trusted and neutral third parties that provide additional services. These trusted third parties can, for example, be tasked with auditing or coordinating the auditing of businesses (Hofman, 2011). In addition, such parties can govern inter-organisational information systems and determine what parties and their systems can connect to it (Pruksasri, van den Berg, Hofman, & Tan, 2016; van Baalen et al., 2009). A PCS can be a trusted third party in some cases as well (van Stijn et al., 2011).

It is important to note that not all types of parties are involved in each supply chain. Furthermore, the same party might perform different functions within the same supply chain. For example, a business acting as a freight forwarder might also act as a customs broker and the seller and the exporter of goods are often the same parties.

Businesses can have different types of relationships with each other. Dabholkar and Neeley (1998) classify interdependencies between businesses according to their temporal perspective (i.e., short term or long term), goal orientation (i.e., individual gain or joint gain), and power balance (i.e., balanced or unbalanced). Based on this framework, they find the following types of interdependencies between businesses: coercive, competitive, cooperative, supportive, command, divergent, coordinative and keiretsu (Dabholkar & Neeley, 1998).

Businesses also might have very different technical capabilities. For instance, a small farming-business might not have the resources to hire technical staff or even have internet access. They might gather only basic information and use a paper-based system to store their information. On the other hand, a large carrier might have the capability to equip containers with GPS-sensors or other IoT sensors and they might have a technical staff that develops and maintains their own information systems and perform data analysis.

In international container shipping, businesses might be involved from a variety of countries in different continents. This means that the goods and previously their components are shipped from a variety of countries as well. Furthermore, it means that different parts of the supply chain might be governed by different sources of law.

To summarise, supply chains can be highly complex. They involve a high variety of businesses that are involved in the shipping of goods directly or indirectly. These businesses might fulfil several different roles in the supply chain and might act on behalf of other businesses. Furthermore, these businesses might have various properties, such as technical capabilities. Furthermore, they might have different relationships with each other.

7.2.2 Types of data and documents shared

Because of the high variety of businesses that can be involved in supply chains, various types of data are gathered and shared between businesses and with customs. There are several papers that provide overviews of the many different types of information businesses gather and share (see e.g., (Jensen & Vatrapu, 2015; H. L. Lee & Whang, 2000; Lucassen, Klievink, Griffioen, & Commission, 2010; van Baalen et al., 2009; Zomer, de Putter, et al., 2014)). For a more complete overview, we would like to refer the reader to these papers. In this section, we provide a description of some of the more common data types and documents.

Lee and Whang (2000) describe four different types of information that are shared in a supply chain between businesses, namely information on inventory, sales, demand forecast, order status for tracking and tracing and product schedule. In addition, businesses might share data on performance measures, such as product quality, and on capacity (H. L. Lee & Whang, 2000). These data types can also involve data generated by IoT devices.

IoT has a significant impact on supply chains and how they are managed. In fact, one of the first application domains discussed for IoT was supply chain management (Gubbi et al., 2013). Even though since then a variety of applications of IoT and variety of IoT devices have been developed, the technology is still considered to be in the early stages (Ben-daya, Hassini, & Bahroun, 2017). This means that this variety could grow in the upcoming years. Ben Daya et al. (2017), provide an extensive and recent overview of the current state of affairs concerning supply chains and IoT.

Two important technologies in IoT are Near Field Communication (NFC) and Radio Frequency Identification (RFID). These technologies can be used to monitor goods in real-time at almost every link in a supply chain (Atzori et al., 2010; Ben-daya et al., 2017). Monitoring of things like the temperature or humidity in which goods are

transported can be especially useful in the case of perishable goods to avoid uncertainty about their quality (Atzori et al., 2010). Monitoring of perishable goods also can lead to higher efficiency and reduced emission levels (Ilic, Staake, & Fleisch, 2009). In addition, IoT devices can be used to provide high-quality product information that businesses can use to respond to changes in the market fast (Atzori et al., 2010). This product information can also be used to provide additional information to the end-consumer (Atzori et al., 2010).

Some of the other literature describes additional types of data that is shared and categorises them differently. In the Cassandra project, for example, the types of data are categorised according to their subject, viz. goods information, party information, transport information, monetary information and data required by law for customs (Lucassen et al., 2010).

When focusing more on documents that are shared between businesses and with customs, we can observe that many of them contain different data types. Several documents are related to agreements made by different parties in the supply chain. First, there is the purchase or sales order. The purchase order is often issued by a retailer and is sent to the business that they choose for supplying their goods (Matthias, Stephen, Jan, & Johanna, 2005; Papazoglou & van den Heuvel, 2007). This document typically contains order details and can be used to identify the buyer of the goods, *inter alia* (Lucassen et al., 2010). A related document is the commercial invoice that the seller sends to the buyer.

Another related document is the international contract of sale. This is a contract between the seller of the goods and the buyer of the goods. This document contains a full description of the goods, unit price, payment details, insurance, and other terms about the planned movement of the goods (van Stijn et al., 2011). The international contract of sale also describes the Incoterms under which goods are shipped. Incoterms are internationally accepted rules related to the rights and obligations between the consignor and consignee (van Baalen et al., 2009). The commercial invoice is a statement for payment to the buyer and contains detailed information on the goods as well (Zomer, de Putter, et al., 2014).

The contract of carriage is a contract between the carrier and the consignor (or shipper) and consignee (Hofman, 2011; Karampetsou, 2016; Klievink, van Stijn, et al., 2012). The rights, liabilities and duties of parties are defined in this contract (Hofman, 2011). A waybill is a receipt for the goods that refers to the contract of carriage (Hofman, 2011). The shipping note describes what is shipped (Lucassen et al., 2010). The ship manifest is a list of all cargo on a ship and is based on the bills of lading associated with the cargo (Hesketh, 2010; Klievink, Aldewereld, & Tan, 2014; Veenstra, Hulstijn, Christiaanse, & Tan, 2013).

A bill of lading is the receipt that the carrier gives to the shipper stating that it has received the goods, it confirms that they agree to transport the goods against a certain tariff, and it shows the details of these goods (Hesketh, 2010; Jensen, Tan, & Bjørn-Andersen, 2014; Levi, 2005; Zomer, de Putter, et al., 2014). Different bills of lading might be issued for different parts of the transport, for example, over land and over sea (Zomer, de Putter, et al., 2014). In that case, the different carriers transmit the information in the bill of lading as well (Zomer, de Putter, et al., 2014). A party that wants to obtain the goods from the carrier needs to show an original copy of the bill of lading (Smeele, 2009).

The packing list is a list of goods that are packed in a container and it is used to stuff the container. It is an important source for information in several other documents (Hesketh, 2010; Jensen & Vatrapu, 2015). For example, the bill of lading and shipping manifest can be based on information in the packing list (Hesketh, 2010).

Jensen and Vatrapu (2015) provide an overview of the ten most important documents that government authorities require in their case study of a trade lane in which flowers are shipped from Kenya to the EU. Some of these documents are shared with government authorities other than customs (e.g., the phytosanitary certificate). The documents that are shared with customs are the export declaration, the pro forma invoice, the certificate of origin, the EUR1 Movement Certificate, the bill of lading, the entry summary declaration (ENS), the notification of arrival, and the import declaration (Jensen & Vatrapu, 2015).

The shipper or consignor should make an export declaration to customs at the country of export before goods leave the country (Hesketh, 2010; Jensen & Vatrapu, 2015). Typically, the export declaration contains a list of goods descriptions and other information, such as certificates (Jensen, Bjørn-Andersen, & Vatrapu, 2014). This document might provide relevant data for customs of the importing country to reuse.

The proforma invoice is generated by the seller of goods. It contains information on the goods and on their price. This document is used by customs to calculate tariffs (Jensen & Vatrapu, 2015).

The certificate of origin describes where the goods that are shipped and their components were made (Brenton & Imagawa, 2005). Proving the origin of goods and obtaining this certificate can require sophisticating accounting procedures and generating and providing other documents, which might be costly (Brenton & Imagawa, 2005). The customs organisation of the exporting country can be involved in issuing the certificate of origin (Brenton & Imagawa, 2005). The customs organisation of the country of import checks the certificate for origin, as it is their responsibility to implement the rules of origin (Brenton & Imagawa, 2005). Following these rules is required for applying trade policy measures (Brenton & Imagawa, 2005). The movement certificate is a certificate that is used to prove the origin of goods (European Commission Directorate-General Taxation and Customs Union, 2016).

A carrier that transports containers over the sea is required by European law to provide customs with an ENS describing the goods it is carrying and to do so 24 hours before loading at the port of departure (The Commission Of The European Communities, 2006b). Customs uses the information in the ENS for risk assessment (The European Parliament and the Council of the European Union, 2013; Zomer, 2011). The data in the ENS should provide customs with the means to do this and it thus contains data elements like a goods description, number of items, number of packages, consignee and gross mass (The Commission Of The European Communities, 2006a). The information in the ENS is based on ship manifests and data from the bill of lading (Klievink et al., 2014; Zomer, 2011).

The import declaration should be submitted to customs by the importer before the arrival of the goods (Jensen, Bjørn-Andersen, & Vatrapu, 2014). It should contain similar data elements as the export declaration and the ENS, but it also might include

additional documentation such as certificates (Jensen, Bjørn-Andersen, & Vatrapu, 2014). The information in the import declaration is often provided to the importer by the exporter (Jensen, Bjørn-Andersen, & Vatrapu, 2014). When a carrier that operates a ship enters the European Union, they should notify customs of their arrival at the customs office of first entry with a notification of arrival (The European Parliament and the Council of the European Union, 2013).

In supply chains, there thus is a high variety of data shared between businesses and with customs. This ranges from different types of data generated by IoT devices to documents that businesses share with each other and with customs. This variety of data already is or could be shared with customs and used by them to monitor the goods flow.

7.2.3 Information sharing systems in the container-shipping domain

The complexity of B2G information sharing in the container-shipping domain is not only affected by the variety of businesses and data types. In addition, the variety of systems involved can contribute to a higher complexity as well.

The technology that businesses use to share information can be quite different. Each party has their own system in which they gather their own data and from which they might share it with others. As the variety of businesses can be high, this makes the variety of systems involved in information sharing high in those cases as well. Jensen and Vatrapu (2015), for instance, describe a trade lane for roses from Kenya to the Netherlands in which more than twenty different information systems are involved.

Communication between businesses and with customs can be based on different mediums, such as electronic data interchange (EDI), phone, fax, text message, email and ordinary mail (Jensen, Bjørn-Andersen, & Vatrapu, 2014; Jensen & Vatrapu, 2015). The information received is often retyped by hand in the system of the party receiving the information (Jensen, Bjørn-Andersen, & Vatrapu, 2014; Jensen & Vatrapu, 2015). The variety of information systems involved, the variety in media used for communication and the manual retyping of information do not only add to the complexity of B2G information sharing, but might lower the quality of the information that is shared as well (Jensen, Bjørn-Andersen, & Vatrapu, 2014; Jensen & Vatrapu, 2015).

There are also parties that provide systems or services that facilitate B2B and B2G information sharing. For example, a business community system is a system that can be used by several businesses in a supply chain (Lucassen et al., 2010). It can be owned by its members or by a commercial party and provide them with different functionalities (Lucassen et al., 2010).

A system important for facilitating B2B and B2G information sharing is the Port Community System (PCS). A PCS is an inter-organisational information system (van Baalen et al., 2009). It provides their services to a community geographically bounded to a port (Klievink et al., 2014). PCSs serve as an electronic hub that facilitates and coordinates information sharing by standardising messages and centralising information sharing (Klievink et al., 2014; van Baalen et al., 2009). They might provide other services as well and often businesses can buy services in modules (van Baalen et al., 2009). PCSs are often used to share documents with customs, such as the export declaration (Klievink,

Janssen, et al., 2012). PCSs reduce the need to retype data in different systems mentioned above and thus might help avoid the related data quality issues (van Baalen et al., 2009).

7.3 Related initiatives to support B2G information sharing

Considering the potential benefits of B2G information sharing in international container shipping, there are several initiatives to develop other solutions to support it. In this section, we discuss the other initiatives to support B2G information sharing in the international container-shipping domain. In section 7.4, we discuss what our context-aware architecture adds to this work.

7.3.1 The single window

In one-stop government, the public services provided by several public authorities or private service providers are integrated (Wimmer & Tambouris, 2002). These integrated public services are then accessible to customers via a single window (Wimmer & Tambouris, 2002). Businesses in international container shipping might need to share information with various authorities, such as customs organisations, phytosanitary authorities, health authorities, and veterinary authorities (Jensen, 2017). One of the other initiatives to support B2G information sharing is to provide these businesses with a single window that they can use to share information. This type of information sharing in which a business has to supply information only once is based on the once-only principle (Wimmer, Tambouris, Krimmer, Gil-Garcia, & Chatfield, 2017).

The United Nations Centre for Trade Facilitation and Electronic Business (UN/CEFACT) (2005) defines a single window as *“a facility that allows parties involved in trade and transport to lodge standardized information and documents with a single entry point to fulfil all import, export, and transit-related regulatory requirements. If information is electronic, then individual data elements should only be submitted once.”* The single window for e-customs provides support for customs to coordinate all cross-border operations and the sharing of documentation with all the other border agencies that are involved in the transport of the goods (Zomer, 2011).

The single window concept has some benefits both for the businesses in the supply chain as well as customs organisations. The main benefit for the businesses is that they can submit all required documentation one time at one point to all customs agencies (UN/CEFACT, 2005). This can reduce delays and lower administrative burden, *inter alia* (UN/CEFACT, 2005). The most important benefit for the customs organisations is that they can use the data that is collected systematically to improve their risk management techniques (UN/CEFACT, 2005).

The UN/CEFACT (2005), mentions three basic models or architectures for the single window concept. For the first architecture, a single authority receives all information and distributes this information amongst all relevant customs organisations. The single authority prevents unnecessary hindrances to the logistics chain by coordination controls. The second model, according to UN/CEFACT (2005), involves a single automated system that collects and distributes the information. Either the information can be processed by a system that is integrated with the systems of the

customs organisations or the information is sent by a system interfacing with the systems of government organisations and processed there. A combination is possible as well. In the last model, an automated information transaction system is used which not only collects the information from the businesses and shares it with the customs organisation but also can be used by customs organisations to send approval of applications back to the businesses.

7.3.2 The shipping information pipeline

The idea of a Shipping Information Pipeline (SIP) was first proposed by UK and Dutch customs (Jensen & Tan, 2015; Pruksasri et al., 2014). It was developed to allow original information to be captured in real-time at the source to increase reliability (Klievink, van Stijn, et al., 2012). The information that is made available in the SIP is the raw and original information that companies have in their systems to base their own operations on (Klievink, van Stijn, et al., 2012). When this data is made available in the SIP, it could be reused for other purposes than that it was gathered for, according to the piggybacking principle (Hofman, 2011; Rukanova et al., 2011). According to Hesketh (2010), the information that is shared between the parties describes the transactional data that is captured by the parties in the supply chain, the physical data that is captured by tracing, tracking and monitoring IoT devices and relevant commercial risk management data such as quality and technical compliance tests. In the pipeline, data on goods and people are distinguished from data on carriages (Overbeek et al., 2011).

The SIP is based on a Service-Oriented Architecture (SOA), in which resources are made available as independent artefacts that can be accessed in a standardised way (Graham, 2006; Klievink, van Stijn, et al., 2012). SOAs are the de facto standard for data integration (Overbeek et al., 2011). In the SIP, each subsequent party in the supply chain makes its source data accessible as soon as it becomes available (Klievink, van Stijn, et al., 2012). With each step, the data is enriched with new data (Klievink, van Stijn, et al., 2012). By linking the data that becomes available in this manner, an integrated data view is created, providing a full view of the trade lane (Klievink, van Stijn, et al., 2012). The SIP is therefore referred to as an integrated data pipeline or seamless integrated data pipeline as well (Hesketh, 2010; Overbeek et al., 2011).

Other types of data pipelines exist as well. The main differences between the SIP and other kinds of data pipelines are that in the SIP data is shared between parties in a supply chain and with customs and that it only supports the sharing of shipping information. Furthermore, it allows for a transition from the current data push approach in which businesses push documents to customs, to a data pull approach in which customs pulls the data they require (Klievink, van Stijn, et al., 2012). Naturally, the access to data in the SIP is only allowed for parties that are authorised to do so by the owners of the data (Klievink, van Stijn, et al., 2012).

In the literature on its more practical design, usually, the SIP involves a single or limited number of central components that the information goes through (Klievink et al., 2014; Lucassen et al., 2010). Such a central component can be a port community system or business community system acting as a central hub, or an event repository (Klievink et al., 2014; Lucassen et al., 2010). An event repository is part of the SIP if it

has a thin information flow and users share events with links to shipping information instead of the shipping information itself (van Engelenburg, Janssen, Klievink, et al., 2017).

7.3.3 Global Trade Digitisation

Blockchain technology was originally developed by Nakamoto in 2008 to store transactions of the cryptocurrency Bitcoin. Following the development of Bitcoin, a high number and variety of other blockchain-based cryptocurrencies were developed, such as Ethereum (Wood, 2017). Blockchain technology has recently gained a lot of attention in the academic world and outside of that for sharing and storing other data than cryptocurrency transactions. Blockchain is investigated to support a variety of processes in e-government, supply chain management and business process management (Batubara et al., 2018; Korpela et al., 2017; López-Pintado et al., 2017; Mendling et al., 2017; Ølnes et al., 2017; Saveen & Monfared, 2016; Schweizer et al., 2017; Tian, 2016; van der Aalst et al., 2017; van Engelenburg, Janssen, & Klievink, 2017a; Weber et al., 2016). Examples of the use of blockchain technologies in this domain are tracing goods (Saveen & Monfared, 2016; Tian, 2016), conflict resolution (Weber et al., 2016), crowdlending (Schweizer et al., 2017) and supply chain integration (Korpela et al., 2017).

An initiative that specifically focuses on using blockchain technology to support information sharing in supply chains and with customs is that of the global trade digitisation (GTD). The GTD is developed by Maersk and IBM and also is referred to as 'Tradelens' (Jensen, 2017; Tan et al., n.d.). The GTD can be viewed as succeeding the shipping information pipeline (Tan et al., n.d.). In fact, the GTD consists of a combination of a data pipeline and blockchain technology (Jensen, 2017). The data pipeline of the GTD is cloud-based and thin and is used to share events (IBM Corporation & Maersk GTD, 2018).

We would like to refer the reader that is not familiar with blockchain technology to section 11.3.1 where we discuss its basics. Here we only discuss how blockchain technology is used in the case of the GTD. In the GTD, blockchain technology is used to store document filings, relevant supply chain events, authority approval status, and audit history (IBM Corporation & Maersk GTD, 2018).

The GTD relies on the Hyperledger fabric for the blockchain which is developed by IBM (IBM Corporation & Maersk GTD, 2018). The Hyperledger fabric consists of multiple permissioned blockchains called channels (Androulaki et al., 2018; Cachin, 2016). In the GTD, there is a channel for each carrier (IBM Corporation & Maersk GTD, 2018). Furthermore, documents are stored on single nodes (IBM Corporation & Maersk GTD, 2018). Only the other nodes in a channel that have permission can have access to the documents (IBM Corporation & Maersk GTD, 2018). The blockchains in Hyperledger store smart contracts written in the language Go (Cachin, 2016). In GTD these are used to govern what information can be written to the ledger (IBM Corporation & Maersk GTD, 2018).

7.4 The need for a context-aware B2G information sharing architecture

In the previous sections, we have discussed the motivation for supporting B2G information sharing in addition to obligated information sharing in international container shipping, as this could improve compliance (section 7.1.2). Furthermore, we have established that there are incentives for businesses to participate in such information sharing (section 7.1.3). However, we also established that to support B2G information sharing in this domain, there are some risks that need to be dealt with (section 7.1.4). Furthermore, we have established that the international container-shipping domain is a highly complex environment (section 7.2). In this section, we discuss how dealing with this complexity requires supporting information sharing with a context-aware architecture. We have also discussed some of the related initiatives to support B2G information sharing (section 7.3). We will discuss what our work contributes to this related work as well.

It is important to note that our line of reasoning mirrors that in section 1.3, but that it is not the same. In section 1.3, we specify the problem statement and objective of the overall research, viz. *“There is a lack of knowledge on what the design of context-aware architectures that support business-to-government information sharing in complex environments should look like”*. As we show in this section, the international container-shipping domain is an instantiation of a complex environment. Developing the context-aware architecture in this domain thus means working on an instantiation of the overall research problem and contributing to solving it.

In chapter 1, we stated that a complex environment is characterised by a high variety of elements with a high variety of relationships and properties. From the discussion of the complexity of the international container-shipping domain in section 7.2, we can derive that this domain is complex. Many businesses are involved that are very dissimilar in many aspects (see section 7.2.1). Furthermore, the data that is or needs to be shared can vary considerably (see section 7.2.2). In addition, the systems that they will use for the sharing of the information can vary as well (see section 7.2.3).

These varieties all could affect what the information flow according to which information is shared should look like. For example, what data can be shared with parties might depend on the type of party. Relationships between parties will be important for who can have access to data and whether sharing is lawful, for instance. Furthermore, if an information sharing system is involved in the information flow for which security was not arranged properly, and the information is sensitive, then there might be additional issues. Here it is also important to mention that our list of things that vary in this domain is not exhaustive and that many other elements in the environment exist for which their properties and relationships could have an impact.

Due to the complexity of the environment of international container shipping, there thus is a need to support different information flows in different situations. As we discussed in chapter 1, to accommodate this, we require a context-aware architecture that adapts the information flow to the situation in which information is shared. However, we still do not know what such an architecture should look like. The problem we want to solve by developing the architecture is thus that there is no knowledge on what the design

of a context-aware architecture that supports business-to-government sharing of information in addition to obligated documents in the international container-shipping domain should look like.

In section 7.3, we discussed related initiatives to support B2G information sharing in international container shipping. Each of these initiatives offers support to information sharing in a limited set of situations. The single window is aimed at supporting the sharing of documents, for example, while the shipping information pipeline can be used to share additional information. For the shipping information pipeline, there are various designs with different scopes and with information flows that are thick or thin that are used in different situations (see e.g., (Rukanova, Henningsson, Henriksen, & Tan, 2018)). This indicates that there is a need for these different systems providing different information flows in different situations. The context-aware architecture will support the sharing according to these different information flows in different situations.

The GTD and SIPs with a thin flow can be viewed as having a context-aware architecture to some extent. In a SIP with a thin flow, the links to shipping information are shared via the pipeline, while the shipping information is shared directly between the systems of businesses and customs. In the GTD, events are shared via a data pipeline. Documents, however, are stored on blockchain nodes and can only be accessed by parties that have permission to do so (IBM Corporation & Maersk GTD, 2018).

Different information is thus shared in different information flows already in SIPs with a thin flow, as well as the GTD. The context-aware architecture would go a step further than these existing systems, however, and it would take a variety of other elements in the context into account. Furthermore, the context-aware architecture would be more autonomous and involve a decision component that decides what the best flow for sharing information is.

The existing initiatives to supporting B2G information sharing in international container shipping offer functionality that can be useful. For example, the shipping information pipeline links information on a container and provides an overview of the supply chain. Using blockchain technology to store documents makes it difficult to tamper with them, which might be useful in case of disputes. As they are useful, the context-aware architecture should not replace these systems. Instead, it should overarch them and include the appropriate system in the flow of information depending on the situation. This work, therefore, does not seek to improve or replace the systems in the existing work. Instead, it seeks to unite them, preferably in such a way that it is easy for any systems that are developed to offer new functionality in the future to connect to the architecture as well.

7.5 Goals and foci for the context-aware B2G information sharing architecture

Based on the problem specification in the last section, we specify two goals for the context-aware architecture in section 7.5.1. The overall architecture should meet these goals. In section 7.5.2, we derive foci for the context-aware architecture from these goals,

according to the steps specified in the new method for designing context-aware systems. These foci are then used to derive sensors, adaptors and context rules for the architecture.

7.5.1 Goals

To address the problem posed in section 7.4, the context-aware architecture should support B2G sharing of information in addition to the obligated documents in the international container-shipping domain. In section 7.1.4, we discussed some of the risks associated with such B2G information sharing in the international container-shipping domain. Especially for businesses, information sharing might entail risks. This might make them unwilling to share. However, the businesses supplying information need to be willing to share, as customs does not want to obligate businesses to do so (see section 7.1.3). To support B2G information sharing, these risks should, therefore, be avoided. One goal for the architecture thus is to provide for information flows in which businesses that supply information are willing to participate.

Information sharing should be lawful as well. This would avoid the risks of businesses sharing information unlawfully and for customs to receive information unlawfully. Moreover, the laws and regulations governing information sharing are meant to protect society against unfair business practices, for example. The second goal for the architecture is thus to provide for information flows that are lawful.

7.5.2 Foci

In this section, we report the results of step 1.1 of the method presented in chapter 6, namely the determining of the foci of the context that we study in this research. As discussed in section 3.3.1, this is part of the activity of defining objectives for a solution. The foci are derived from the problem statement (section 7.4) and the goals of the artefact (section 7.5.1), which we discussed already in this section.

According to the method, identifying the focus should start with identifying the problem we want to solve. We have identified and described the problem for the overall research in section 1.3 and further specified the problem in the previous sections of chapter 7 for the domain of international container shipping. The problem we want to solve by developing the architecture is that there is no knowledge on what the design of a context-aware architecture that supports business-to-government sharing of information in addition to obligated documents in international container shipping should look like. Next, we need to specify the goals of the design, which we did in section 7.5.1. In this section, we perform the last part of step 1.1 and specify what the world should look like when these goals are met, and we formulate the foci based on that.

The first goal of the architecture is reached when businesses that supply information are willing to participate in the flow of information provided by the architecture. Based on this, we can formulate focus 1.

Focus 1:

The willingness of businesses supplying information to participate in the information flow provided by the architecture

The second goal of the architecture is reached when the architecture provides for lawful information flows. Hence, we can formulate focus 2.

Focus 2:

The lawfulness of the information flow provided by the architecture

8 Design process for the context-aware architecture

In section 2.2, we discuss our choice for a design science approach. We also provided an introduction to the activities for performing design science as proposed by Peffers et al. (2007), viz.: 1) problem identification and motivation, 2) define the objectives for a solution, 3) design and development, 4) demonstration, 5) evaluation and 6) communication. In this section, we describe our choice of methods for each of these activities for developing the context-aware architecture. In the sections where the results of the activities are presented (for an overview see table 2, p. 54), we discuss how we applied the methods in detail.

Designing and performing design science research is usually not a neat and clean linear process of going through all the design activities sequentially. In fact, usually, the process consists of several small and big iterations (Hevner et al., 2004; W. Kuechler, Vaishnavi, & Petter, 2005). This includes, for example, reconsidering the problem and make changes to the design based on new insights. While we present our research design in a linear fashion to enhance clarity, it is important to note that we made some larger and many small iterations between different stages as well. The design of the architecture presented in this dissertation was preceded by two previous designs, published in van Engelenburg, Janssen and Klievink (2015) and in van Engelenburg, Janssen and Klievink (2017a). We start by providing a summary of these architectures. In the rest of this section, we refer back to the work on these previous versions when they have contributed to the development of the context-aware architecture presented in this dissertation.

8.1 Previous designs of the architecture

For both previous versions of the architecture, we focused on different aspects of the design problem and we provided different solutions for them in the different designs. These previous versions were not yet fully context-aware and we did not apply the new method. However, they do contain elements that support context-awareness in the architecture presented in this dissertation, such as distributed information sharing and access control using business rules. In this section, we provide a general overview of the two previous versions of the architectures. We also discuss how they were investigated.

The first design of the architecture, which was published in van Engelenburg, Janssen and Klievink (2015), focused on two things, 1) the requirements that an architecture needs to meet in order for businesses to be willing to participate in information sharing, and 2) the possibility of allowing businesses to use a combination of business rules and encryption and decryption to control their data. For this design, we started out by studying literature on information sharing in the container-shipping domain to establish the requirements that an architecture should meet for businesses to be willing to participate in sharing information using the architecture. We derive three requirements, viz. 1) keeping information confidential when needed, 2) ensuring there is no obstruction for information sharing from the possible increase of liability when businesses receive information, and 3) ensuring the sharing of information and its use complies with legislation.

The first version of the architecture we developed consists of a decision component and a component that allows access to data according to the decision. Metadata, business rules, global rules and context information on the requester of access to data is used as input for the decision component to reach a decision. Access to data is prevented or granted by respectively encrypting parts of data and decrypting parts of data using private keys.

Business rules could be specified by businesses that send information. These rules are applicable even when information is not received directly from its original source or when it is combined or enriched. This empowers business by providing them with control over their information sharing. In addition, this makes it easier for businesses to share data that they have received from others, as they do not have to worry about unintentionally revealing sensitive data from others. In addition, global rules are used to make sure that access to data complies with legislation.

The evaluation of this architecture relied on analysing the extent to which the requirements were met by the architecture. We concluded that overall, an architecture incorporating business rules, global rules, a decision component and encrypted data has enough potential to merit further investigation.

The first design shared an important principle with blockchain technology, namely distributed information sharing. Furthermore, in both cases, there is an important, albeit different, role for encryption and decryption of data using public and private keys. Therefore, for the second design, we focused on the possibilities of using blockchain technology to support B2G information sharing. Furthermore, we focused on data sharing to improve public safety and security, as this is one of the important uses of the data for Customs. This second design was published in van Engelenburg, Janssen and Klievink (2017a).

For the design of the second version of the architecture, we performed the same activities described by Peffers et al. (2007) as we followed for the design of the architecture in this dissertation. The requirements for improving the willingness of businesses to participate in information sharing that we generated for the first version of the architecture were further worked out based on the data we collected using the interviews we did as part of studying the information flows in Cassandra (see section 8.3.2.1). Furthermore, we used transcripts and minutes of two workshops with several staff members at Maersk Line, which is a large sea carrier of containers. The staff members had expertise in the juridical domain and the domain of IT innovation. The requirements we established were similar. However, they were confirmed by the new data and were able to add more detail to their description.

The second version of the architecture we developed consisted of five main components; 1) blockchain for recording events, 2) business rules for setting the conditions to share information, 3) access control to ensure only authorised access, 4) metadata and context information to understand whether the context enables information to be shared, and 5) encryption and decryption. Blockchain technology was used to create a general ledger of events that is accessible to customs and that enables the secure sharing of information that is signed by businesses to indicate that they believe the information to be true. Business rules are used to set parameters about under what conditions to share

information. Metadata and context information about the requester of access to data, together with the business rules, are used as input for the decision component to reach a decision about whether to share data. In this way, data sharing is context-dependent and is controlled by businesses. Access to data is prevented or granted by respectively encrypting data or providing a key to decrypt data elements.

We demonstrated the architecture using a typical user activity. The evaluation of the architecture was partially based on analysis. In addition, we presented and discussed parts of the architecture and the principles on which it was based during workshops at Maersk Line. We concluded that blockchain technology seems suitable to ensure the trustworthiness of information by letting businesses that can know whether the information is true ‘sign’ it. Furthermore, specifying business rules enables businesses to keep data confidential whenever they need to. In addition, in the architecture, the sharing of data that are received by others, that are enriched or combined, is made easier by making it possible for the business rules of multiple parties to be applicable. Furthermore, it seems that the use of generic business rules to ensure that data access complies with legislation is also a means to increase willingness to share information. The way the generic business rules and the decision component are governed is important for this.

In both of the previous architectures, we already incorporated the piggybacking principle that is applied in other initiatives to support B2G information sharing in international container shipping (e.g., the SIP in section 7.3.2). This means that the architectures supported B2B as well as B2G information sharing to make it easy for customs to piggyback on the B2B information flow. Furthermore, the previous architectures supported the sharing of obligated documents as well as additional information. This also makes it easier to share additional information as the same systems can be used to do so. We applied the same principles for the architecture presented in this dissertation.

8.2 Activity 1: Problem identification and motivation

The research problem and motivation are described on a high level in section 1.3. The overall research problem is that *“there is a lack of knowledge on what the design of context-aware architectures that support business-to-government information sharing in complex environments should look like”*. Based on this, we formulated the following objective: *“Create a design for context-aware architectures that support business-to-government information sharing in complex environments.”*

However, to get deeper insight into the problem, we have to make it more tangible. We can do so, by studying a specific instance of the problem. B2G information sharing in international container shipping provides an opportunity to do so, as the international container-shipping domain is a complex environment. Information sharing in international container shipping has been investigated quite thoroughly in other work. This means that there are ample sources to use as a basis for our study of this domain.

We analysed the relevant literature to establish the need for a context-aware architecture in this domain. Based on the literature, we describe in section 7.1 that B2G information sharing in this domain can be beneficial and that there are incentives for business to share additional information, providing a motivation for solving the problem.

We also describe that such information sharing in this domain is associated with risks that need to be dealt with. In section 7.2, we use the literature to give an overview of the international container-shipping environment and the variety of elements in that environment. In section 7.3, we describe some of the related initiatives. In section 7.4, we synthesise all the information from the literature to show that for this domain, we need to solve an instantiation of the overall research problem.

8.3 Activity 2: Define the objective of a solution

Defining the objectives of a solution entails determining the objectives of the context-aware architecture. As we argued in section 3.3.1, in the case of designing context-aware systems and architectures in complex environments, an additional connection to the environment needs to be made. The reason for this is that what flow of information is appropriate will depend on the situation in which information is shared. The relationship between different situations and the reaching of the goal of the architecture thus needs to be established. In chapter 6, we provide a method for doing so. The first step of this method, presented in section 6.1 can be used to get the required insight into relationships.

In this section, we explain how we performed step 1 of the new method. Step 1 consists of three sub-steps, namely 1.1) determine the focus, 1.2) gather data and 1.3) analyse the data (section 6.1). In each subsection of this section, we explain how we performed each of the sub-steps.

8.3.1 Step 1.1: Determine the focus

We want to get insight into context and decide what belongs to the relevant context that we should take into account in our design. Step 1.1 of the method provides a basis for this, by establishing the focus of the context that will be investigated. In other words, what belongs to the relevant context depends on what has an impact on the foci of the architecture.

The foci for the context-aware architecture need to be derived from the problem specification and goals of the architecture. In section 8.2, we discussed how we derived the problem specification. We derived the goals from the problem specification. For our line of reasoning, we relied on work performed in previous design cycles.

Initially, for the architectures presented in van Engelenburg et al. (2015) and van Engelenburg et al. (2017a), we derived three requirements from the literature that need to be met in order for businesses to be willing to share information using an information sharing architecture. These requirements were 1) keeping information confidential when needed, 2) ensuring there is no obstruction for information sharing from the possible increase of liability when businesses receive information, and 3) ensuring that the sharing of information and its use complies with legislation. We presented them during workshops at a large sea carrier with several staff members who have expertise in the juridical domain and the domain of technical innovation.

For deriving the goals of the architecture, we use the same line of reasoning as we did for deriving these requirements. We merged requirement 1 and 2, as the need to keep information confidential and avoiding liability for businesses both are of importance

because it affects whether businesses are willing to share. In accordance with the method, we then derived two foci from these goals by making explicit what the world should look like for these goals to be met.

In section 7.5.2, we present the results of step 1.1. The foci we derived are the following: 1) the willingness of businesses supplying information to participate in the information flow provided by the architecture, and 2) the lawfulness of the information flow provided by the architecture. In the next section, we discuss how and what data we gathered on the situations that impact these foci.

8.3.2 Step 1.2 and 1.3: Gathering and analysing data

In step 1.2, we want to determine what situations impact the foci of the context-aware architecture identified in step 1.1. In this step, high-level descriptions of these situations are generated. In step 1.3 these are further refined and the context elements are derived from the description of these situations. As we discuss in section 6.1, getting insight into context involves alternating step 1.2 and step 1.3, as refinement might lead to finding new gaps in understanding for which additional data needs to be gathered.

The purpose of step 1.2 and step 1.3 is to get insight into what belongs to the context of each of the foci. The insight into context is expressed as a list of context elements and context relationships. We will use this output of the steps to derive the sensors and adaptors and the context rules in step 2 and step 3 of the method (see section 8.4.1).

In the rest of this subsection, we discuss how we performed step 1.2 and step 1.3 for each of the foci.

8.3.2.1 Focus 1: The willingness of businesses supplying information to participate in the information flow provided by the architecture

The focus of the willingness of businesses supplying information to participate in the information flow provided by the architecture is very broad. Willingness is affected by the benefits and the risks businesses experience from information sharing. This research is limited to determining the context elements that impact willingness by reducing risks. The rationale for this is that the benefits of information sharing are often external and not something that the system can manipulate. This is, for example, the case when customs decides to reward businesses that share additional data with them, or when a business gets a fee for providing access to their data. The risks of sharing data can usually be reduced more effectively by adaptation of the architecture, for example, by taking certain measures to filter or secure data. Therefore, investigating context elements that impact the risks has priority.

We used a case study approach to gather data on the focus of the willingness of businesses supplying information to participate in the information flow provided by the architecture. In section 4.4, we already discussed what we mean by ‘case study’ in this research and we discussed for what kind of research the case study method is suitable.

We also discussed some criteria for determining when the case study method is appropriate based on the work of Benbasat et al. (1987) and Yin (1994).

According to Benbasat et al. (1987) and Yin (1994), the case study method is suitable when a phenomenon should be investigated in its natural context. In our case, the phenomenon we are investigating is the impact of context on the foci of the context-aware architecture. This means that we do not only need to investigate something in its context, but the context of the architecture itself is the subject of our investigation.

The second criterion that Benbasat et al. (1987) and Yin (1994) mention is that there should be a focus on contemporary events. This is also true in our case. We expect that the willingness of businesses to share information depends on things like the type of data that is to be shared and the systems that are involved. These things are influenced by technological developments. A contemporary view on willingness thus is suitable.

A third criterion is that the type of knowledge should be suitable for the case study method (Yin, 1994). In section 6.1.2 we stated that investigating the situations that impact the foci of a context-aware system or architecture is exploratory research, at least in the early stages. Namely, the designer is exploring the situations for which there is this impact. The case study method is suitable for explorative research (Yin, 1994). In later stages of the research, the findings are further refined and a generalisation from specific situations to high-level situation descriptions takes place (see section 6.1.3). For this, a replication logic is followed. This replication logic is the same logic used to generalise from cases as described in the work of Yin (1994) and Eisenhardt (1989), for example.

A final criterion posed by Benbasat et al. (1987) and Yin (1994), is that no manipulation or control over variables is necessary or possible. We believe that this is true for this research to a certain extent. We want to know what impacts the willingness of businesses to participate in an information flow. We can do so by investigating cases in which businesses share information. This will provide information on what impacts their willingness to share, without manipulation of variables.

However, it might also introduce a blind spot in the research. Namely, only information flows are included in which businesses are willing to participate and this might mean that information flows that were rejected by the businesses will not be identified, nor the reasons for rejecting them. However, on the other hand, we cannot observe information flows in practice in which businesses are not willing to participate, as they will not exist for that precise reason. Furthermore, we want to test some of our thoughts on new solutions and new flows of information that the context-aware system could provide for.

A solution to this is to not only gather information on the willingness of businesses to participate in actual information flows in the cases we selected, but also on some scenarios of information flows that do not actually exist in the cases. This means that these scenarios include systems, data and parties from the case, but in a different flow. This has the advantage of still having a tie to reality for the scenarios.

The case study design to investigate the context of willingness is presented in section 9.1.1. The results are presented in section 9.1.2.

8.3.2.2 Focus 2: the lawfulness of the information flow provided by the architecture

Investigating the context of the focus of the lawfulness of information flows was conducted in two phases. In the first phase, an initial set of situations restricting the focus was identified. In the second phase, the description of these situations was tested and refined.

8.3.2.2.1 Phase 1: Scoping and Exploration

As the juridical domain is very broad, it is not feasible to collect data from every part of it. Therefore, we first made a selection of juridical fields to include in our study. This selection of relevant fields was made based on the extent to which juridical considerations from the fields are likely to impact the lawfulness of flows of information in our domain. We took into account several considerations and sources of evidence to make the selection, amongst which workshops, interviews with juridical experts and documentation from related projects. Based on this, we determined that the fields of competition law, Intellectual Property (IP) law and customs law are most important to take into account, as they are most likely to impact the lawfulness of information flows.

After establishing what juridical fields are most important for this research, we tried to generate a first set of context relationships that impact the lawfulness of flows of information. The legislator is the party that determines what is lawful and what is not. It thus makes sense to start with collecting data from the legislator and we used the new method to derive some initial context relationships directly from juridical texts. In addition, we added some context relationships after discussion with a member of our project with a juridical background.

8.3.2.2.2 Phase 2: Testing and refining of the context relationships

Determining whether something is lawful cannot be done by simply reading an applying a law, or as one of the juridical experts we interviewed stated: *“the law is not a cookbook”*. Instead, to determine whether something is lawful, legal texts should be interpreted and the intention of their author should be established (Barak, 2007). Such an interpretation can involve solving contradictions within the legal text and also between legal texts (Barak, 2007).

Considering this, the context relationships generated in the exploration phase are of limited value to base on directly the design of the context-aware architecture. They are directly derived from the law, without further interpretation. Furthermore, some of them are based on conversations involving a juridical expert with expertise in fields different from those in this study. Therefore, they require additional testing and refinement. Additional context relationships might be found with further investigation as well.

These context relationships found in the exploration phase can be viewed as providing a general overview of the juridical issues that play a role. This provided us with the structure necessary for further investigation. Furthermore, this provided us with the opportunity to generate already some possible ways in which the architecture can adapt

so that these issues are dealt with and so that information sharing is lawful. These possible solutions can be formulated as positive context relationships and these can be tested.

Interpreting the law requires a high level of juridical knowledge and expertise. To refine and test the context relationships and to add to them, we thus obtained information from juridical experts. To do so, we performed expert interviews. The expert interviews and the way they were performed are described in further detail in section 9.2.1.2.

Phase 2 thus leads to two results. The first result is a list of context elements and context relationships. In addition, we already crossed the line towards activity 3 and collected information on what the sensors and adaptors for the different context elements should look like. The results of this are presented in chapter 10.

8.4 Activity 3: Design and development

The context-aware architecture needs three types of components to perform its functions. The first type of components already has been discussed extensively. These are the components that depend directly on what belongs to the relevant context of the architecture, namely sensors, adaptors and context rules. These components are necessary for the context-aware architecture to sense context and adapt to it autonomously.

The second type of component that the context-aware architecture needs to incorporate is information sharing systems that provide functionality to their users, for example, the systems we discuss in section 7.3. As the developments of new technologies in ICT continue, we expect continued development of new systems that provide different types of functionalities to the user. Proposing a static set of systems would make the context-aware architecture unnecessarily inflexible. Flexibility can be important as businesses might not know with whom they will collaborate and share information next (Shishkov, van Sinderen, & Verbraeck, 2009).

Therefore, we do not consider these type of components as part of the design and development activity of this research. Instead, we provided insight to designers of such systems into the impact of different design choices for their system on the willingness of businesses to use it in van Engelenburg, Janssen and Klievink et al. (2017, 2018). This is necessary, as whether their system will be part of the flow of information in different situations will partially depend on this willingness. They should thus ensure the balance between risks and benefits that their system provides ensures that businesses are willing to use their system in situations where they want to provide their functionality.

For context-awareness, not only sensors, adaptors and rules are required. In addition, the architecture needs some basic components to store context information and to make decisions based on the rules. Furthermore, the architecture needs to provide for the flow of information that is decided upon. Part of the design process is to determine what these components should look like as well. The three types of components discussed should be connected to each other to form the context-aware architecture.

8.4.1 Step 2 and 3: Determining the sensors, adaptors and rules

The sensors and adaptors that the context-aware architecture requires can be derived from the insight in the context that we gained in the previous activity. Deriving the sensors and adaptors can be done in step 2 of the method described in section 6.2. Following this method, we first performed step 2.1 (section 6.2.1) and we determined for each of the context relationships which of its context elements would be an appropriate adaptor.

Typically, the decision for what context element to manipulate using an adaptor is based on common sense. Many context elements, such as what businesses are competitors, can simply not be manipulated. In addition, in many cases solutions were already found and tested in the documents of projects from which we gathered information to get insight into context concerning willingness. For example, from these documents, we derived that businesses do not want to share data in a data pipeline that is global and thick, and that therefore a thin global pipeline was provided. It thus makes sense to include an adaptor in our system that makes information flows thin as well. In addition, we tested some initial ideas for possible adaptors during the expert interviews with the juridical experts to check whether they are viable. We based our decision on these discussions as well.

In step 1.2 (section 6.2.2), the sensors are determined. In accordance with the method, we focused on what we needed to observe and how to observe it to get the appropriate context information. However, again, in many cases deriving the sensors could be based on common sense. The results of performing step 2 are presented in section 10.2.

We obtained additional information on what requirements the sensors and adaptors should meet from a juridical point of view to ensure that information sharing is lawful. We obtained this information during the expert interviews with juridical experts. The way in which these interviews were performed and the relevant information was derived from them is discussed in detail in section 9.1.1.

The context rules were derived in step 3 (section 6.3). At this stage, we already determined for different context relationships what context elements will be sensed and manipulated. A context rule can be derived from each context relationship by simply putting the predicate (or its negation for negative context relationships) for the element that is manipulated in the header and the other elements in the body. The results of performing step 2 are presented in chapter 10.

8.4.2 Basic components to support context-awareness

In this section, we describe the methods for identifying and designing the basic components that are necessary for the context-aware architecture to be context-aware. What basic elements the context-aware architecture requires and what they should look like is derived from what the context-aware architecture should do to be context-aware.

To identify the basic components that the architecture needs in addition to the sensors and adaptors, we first identified in the literature an existing architectural pattern that fits with what we need for the context-aware architecture. Next, we determined what tasks should be performed according to that pattern. Then we derived what components the architecture requires to perform those tasks.

The design of the basic components was guided by the goals for the architecture we defined in section 7.5. For fulfilling these goals, we relied on the insight we gained in previous cycles in the research about the willingness of businesses to participate in information sharing and its lawfulness. Based on this insight, we identified blockchain technology as a useful technology to apply in our design.

To get more insight into how this technology could be used, we review the literature to derive its basics and its typical characteristics. This provides insight into the advantages of blockchain technology that we could use, as well as what risks we should avoid. The basic components of the architecture are described in chapter 11.

8.4.3 Overall design of the architecture

When we have determined what the components of the architecture should look like, the overall architecture can simply be established by connecting each of the components. This connection is determined by the functionality that they provide. For example, the sensors generate context information and should thus be connected to a component that stores context information, and so on. The combining of the components into the overall architecture is thus mainly based on common sense. The overall design of the context-aware architecture is presented in chapter 12.

8.5 Activity 4: Demonstration

The artefact that is designed should be demonstrated to show how it can be used to solve an instance of the design problem (Peffer et al., 2007). We demonstrated the architecture in one easy scenario for explanation purposes and two typical scenarios in which information is shared between businesses and with customs. We generated the scenarios based on an interview with an expert at customs. Purpose of this interview was to determine what information sharing would be useful to customs and that would be difficult to do in the same system, hence requiring the context-aware architecture to adapt and support information sharing in both scenarios. The context-aware architecture based on these scenarios is demonstrated in section 12.2.

8.6 Activity 5: Evaluation

Evaluating an artefact involves measuring how well the artefact supports a solution to the problem (Peffer et al., 2007; Verschuren & Hartog, 2005). In chapter 7, we established that for the context-aware architecture to support additional B2G information sharing in international container shipping, information sharing using the architecture should be lawful and businesses should be willing to share. To determine the extent to which the context-aware architecture provides a solution to the problem, we, therefore, need to measure the extent to which the architecture meets these goals.

To evaluate the architecture, we should determine the extent to which the flows of information provided for by the architecture in different situations, in fact, are lawful and whether businesses, in fact, are willing to participate in them. However, the number of situations and possible flows of information that the architecture could provide are numerous, if not infinite. It is not feasible to list them and check them one-by-one.

Another approach could have been to compare the context-aware architecture with existing solutions that support B2G information sharing. However, the context-aware architecture incorporates these existing solutions, such as data pipelines and single windows. The context-aware architecture thus will provide an appropriate flow of information in the same, or a higher variety of situations as these architectures, except in two cases: 1) the context-rules are incorrect and do not lead to the selection of an appropriate flow of information, or 2) businesses are not willing to use the context rules, sensors, adaptors, or the basic components necessary to support context-awareness, or this is not lawful.

This means that to evaluate the architecture, we should do two things 1) validate the context model, and 2) evaluate the architecture itself. The context model is validated by interviewing an expert with expertise in international container shipping and an interviewee with juridical expertise. We asked them to verify whether the context relationships that we derived are correct. The validation of the context model also contributes to the evaluation of the new method. The results are thus presented in section 14.2.1.

For the evaluation of the architecture itself, we need to establish the answer to three questions:

1. Is information sharing using the architecture lawful?
2. Are businesses willing to use the architecture?
3. Is the architecture useful to customs for compliance monitoring?

Questions 1 and 2 are directly related to the lawfulness and willingness in the goals of the architecture. Question 3 was added, as the initial motivation of the research is the need of customs for data to base their compliance monitoring on. To support such information sharing we needed to ensure willingness and lawfulness. However, we should ensure that the resulting architecture is useful to customs.

Answering these questions calls for different perspectives on the architecture. To answer question 1, a juridical perspective is required. To answer question 2, a business perspective is required and answering question 3 requires the perspective of customs.

Implementing an overarching context-aware architecture involving a variety of parties was not possible. This would have been very costly and would require a variety of parties and systems (e.g., data pipelines, customs) to be connected to the architecture. This is something that could take years and requires much more resources than available. This means that we performed an artificial evaluation instead of a naturalistic one. We gathered data from three sources: 1) workshops at Maersk Line, 2) interviews with an expert in formal law at Dutch customs and an expert from academia in trade law, and 3) an interview with an expert in IT and governance. Together, they cover all perspectives.

How	When	What
First four workshops Maersk	Ex ante	Juridical: -Is information sharing using the architecture lawful?
		Business - Are businesses willing to use the architecture?
Final workshop Maersk	Ex post	Juridical: -Is information sharing using the architecture lawful?
		Business - Are businesses willing to use the architecture?
Interview expert in formal law Dutch Customs	Ex post	Juridical -Is information sharing using the architecture lawful?
		Customs - Is the architecture useful to customs for compliance monitoring?
Interview with expert in trade law from academia	Ex post	Juridical: -Is information sharing using the architecture lawful?
Expert on IT and governance	Ex post	Business - Are businesses willing to use the architecture?

Table 11: Overview of the methods used to evaluate the context-aware architecture based on the framework of Pries-Heje et al. (2008)

Maersk Line is a partner in the project of which this research is part. We presented our work at several workshops at Maersk Line, which were attended by some of their juridical experts as well as their experts on IT and innovation. Furthermore, the academics from Erasmus School of Law that are included in the project of which this research is part were present as well, as were academics with expertise in IT and governance and information sharing in international container shipping. There were five workshops in total. At the last workshop, the final version of the architecture was presented and feedback was requested from participants. In the previous workshops, data feedback was obtained on previous versions of the architecture. These workshops contributed to answering question 1 and 2.

We added an additional interview with an expert in IT and governance with an academic background and experience in industry in higher management positions. This interview also contributes to answering question 2. It was added because the architecture is complex and abstract. Including an additional expert with an academic background reduces the possibility that things are overlooked due to this complexity and abstractness.

Last, but not least, the interviews with the formal law expert at Dutch customs and the expert in trade law from academia contributed to answering questions 1 and 3.

In the section on the evaluation of the method (see section 4.4), we already discussed the choices that need to be made on different dimensions in order to evaluate an artefact. We will use the same framework of Pries-Heje et al. (2008) to discuss the choices we made in the evaluation of the architecture. Table 11 shows an overview.

8.7 Activity 6: Communication

The last activity that Peffers et al. (2007) describe is the communication of the results of the research. This dissertation is one way to communicate the results. In addition, we have written several papers presenting the results of this research. The following papers have been published:

- van Engelenburg, S., Janssen, M., & Klievink, B. (2015). Design of a Business-to-Government Information Sharing Architecture Using Business Rules. In *Software Engineering and Formal Methods* (Vol. 9509, pp. 124–138). Springer. <http://doi.org/10.1007/978-3-319-15201-1>
- van Engelenburg, S., Janssen, M., & Klievink, B. (2017a). Design of a software architecture supporting business-to-government information sharing to improve public safety and security. *Journal of Intelligent Information Systems*. <http://doi.org/10.1007/s10844-017-0478-z>
- van Engelenburg, S., Janssen, M., Klievink, B., & Tan, Y. (2017). Comparing a Shipping Information Pipeline with a Thick Flow and a Thin Flow. In *Electronic Government, 16th IFIP WG 8.5 International Conference, EGOV2017, St. Petersburg, Russia, September 4-7, 2017, Proceedings* (Vol. 10428, pp. 228–239). Springer International Publishing AG. <http://doi.org/10.1007/978-3-319-64677-0>
- van Engelenburg, S., Janssen, M., Klievink, B., Tan, Y., & Rukanova, B. (2018). Comparing the Openness of Archetypical Business-to-Government Information Sharing Architectures. In A. Zuiderwijk & C. C. Hinnant (Eds.), *Proceedings of 19th Annual International Conference on Digital Government Research (dg.o'18)*. ACM, New York, NY, USA. <http://doi.org/10.1145/3209281.3209350>
- van Engelenburg, S., Janssen, M., & Klievink, B. (2018). A Blockchain Architecture for Reducing the Bullwhip Effect. In B. Shishkov (Ed.), *BMSD 2018, LNBIP 319* (pp. 69–82). Springer International Publishing AG. <http://doi.org/10.1007/978-3-319-94214-8>
- Klievink, B., Janssen, M., van der Voort, H., & van Engelenburg, S. (2018, September). Regulatory Compliance and Over-Compliant Information Sharing—Changes in the B2G Landscape. In *International Conference on Electronic Government* (pp. 249–260). Springer, Cham.

The first two papers are previous versions of the architecture presented in this thesis. Some of the other papers present analyses of different design choices and their effects on

the willingness of businesses to use an information sharing architecture. The last paper provides an overview of the changes in the B2G information sharing landscape.

In addition, we shared the results with the public in- and outside of academia by performing the following activities:

- presenting intermediate results and end results of the research at workshops at Maersk Line attended by juridical experts and experts on technological innovation (14 January 2016, 7 December 2016, 11 May 2017, 15 May 2018, Copenhagen, Denmark),
- presenting and participating in 8th meeting of the European Commission customs 2020 Project Group to study a possible framework to develop the EU Single Window environment for customs (EU-SW) including the legal context (12-14 March 2018, Rotterdam/the Hague, the Netherlands),
- presenting and participating in European Commission workshop on “Blockchain and Distributed Ledger Technology for Taxation and customs IT Systems” (29-30 May 2018, Valetta, Malta),
- presenting and participating in a discussion about blockchain technology and the context-aware architecture at Open University (18 September 2018, Utrecht, the Netherlands), and
- presenting and participating in a discussion about blockchain technology and the context-aware architecture at Regioraad Zuid-West of EvoFenedex (19 September 2018, Naaldwijk, the Netherlands).

9 The context of B2G information sharing in the container-shipping domain

In this section, we describe our findings on what belongs to the relevant context of B2G information sharing in the container-shipping domain that should be taken into account in the design of the architecture. We also describe the instruments and the methods we used for getting this insight. This section is divided into two parts. In section 9.1, we present the methods and the results concerning the first focus of the willingness of businesses supplying information to participate in the information flow provided by the architecture. In section 9.2, we present the methods and the results concerning the lawfulness of the information flow provided by the architecture. In the last subsection, we discuss the validity and reliability of this part of the research.

9.1 Focus 1: The willingness of businesses to participate in the information flow

In this section, we present the design of the case study we performed to investigate the context of the focus of willingness. In addition, we provide an overview of the results and discuss what belongs to the context of this focus.

9.1.1 Case study design

According to Yin (1994), there are several important components of a case study design. We will describe each of them in this section for our case study. The first important component is the case study questions. The goal of this case study is to collect data on what impacts the focus of willingness. More specifically, we want to know what restricts the willingness of businesses to participate in an information flow.

The second component mentioned by Yin (1994), namely the propositions, we discuss in section 9.1.1.1. The third component, the units of analysis, we discuss in section 9.1.1.2. As we also used scenarios for the research, we discuss these scenarios here as well. In section 9.1.1.4, we discuss the logic linking the data to the propositions and the criteria for interpreting the findings. In addition to the components mentioned by Yin (1994), we discuss the methods we used for collecting the data in section 9.1.1.3.

9.1.1.1 Using a focus instead of propositions

According to Yin (1994), propositions are used to direct attention to something that should be examined and to help to identify sources of relevant evidence. In this research, we do not have such propositions. However, we can use the focus for the exact same purpose. It provides, quite literally, something for the researcher to focus on. If something does not impact the focus, it is not relevant.

9.1.1.2 Cases, unit of analysis and scenarios

According to Darke, Shanks and Broadbent (1998), a unit of analysis identifies what is considered a ‘case’ in the study. We want to gather information on what situations restrict the willingness of businesses to participate in an information flow. Taking into account

the scope of this research, information flows in international container shipping are a suitable unit of analysis in the case study. For these flows of information, we want to know in what situations businesses are and are not willing to participate in them.

For exploratory case study research, both single and multiple case study designs are a possibility (Darke et al., 1998). In this research, we study multiple cases. According to Yin (1994), a multiple case study has the advantage of being considered more robust on the one hand, but it requires more resources on the other. It should follow a replication logic, in which an attempt is made to replicate the results from one case in the other (Yin, 1994). As we describe in the next section, we will follow such a logic in this research.

In this research, including multiple cases has the additional advantage of being broader and allowing us to uncover more context elements and context relationships that impact the focus. If we only consider a single information flow between businesses and customs, then there might be things specific to that information flow that lets us collect information only on certain context elements. For example, if in the case commercially sensitive data is shared. However, it does not allow us to learn about what impacts willingness when the data shared is not commercially sensitive. Including multiple cases might help to deal with this. As we discussed in our motivation for using the case study method (see section 8.3.2.1), for the same reason we include scenarios in our study as well.

A disadvantage of doing a multiple case study is that it requires more resources (Yin, 1994). To accommodate the requirements of the research with the resources available, we did a secondary study of the data from two projects, namely the Cassandra project and the CORE project. In these projects, architectures supporting information sharing in the container-shipping domain have been designed, implemented and tested. To design, implement and test such architectures, the willingness to share data of the businesses in the information flows provided for by the architectures has to be taken into account. These flows of information can thus be used as cases in this research in which businesses are willing to share. In addition to this, we developed scenarios to investigate information flows in which businesses might or might not be willing to share.

In the remainder of this section, we provide background information on the Cassandra project and the CORE project and the cases that we included in this study.

9.1.1.2.1 Case and scenarios from the Cassandra project

The aim of the Cassandra project is *“to make container logistics more efficient and effective by enabling and facilitating the combination of existing information sources in supply chains into new and better visibility that allows the assessment of risks by both business and government”* (Lucassen et al., 2010, p. 3). To reach this aim, the concept of a data pipeline was detailed and demonstrated (Lucassen et al., 2010). Part of the project was using several trade lanes between Asia and Europe as a living lab for demonstrating different configurations of the data pipeline (Lucassen et al., 2010). The cases we studied were one actual flow of information between businesses in one of these trade lanes with customs and two scenarios in which information was shared by the same businesses, using the same, or similar systems, but in a scenario with a flow of information that was not actually implemented in Cassandra.

The research in Cassandra on these trade lanes provides detailed descriptions of the trade lanes themselves, as well as the information shared by parties in them. This provides us with the opportunity to obtain a description of the flows of information, as well as to obtain information on their contexts. In the project, the situation concerning the sharing of information as-is is described and analysed in detail. In addition, the concepts for the data pipeline are demonstrated and evaluated. Since the data pipeline provides for a different flow of information than in the situation as-is, it is reasonable to expect that the Cassandra researchers had to put effort into determining whether businesses were willing to participate in them and that they collected information on this.

One of the architectures implemented and tested in the Cassandra project was a data pipeline for information sharing in a supply chain from Yantian in China and the port of Felixstowe in the UK. Via this data pipeline, a packing list that is verified by a tallyman could be shared with other parties, including customs. Businesses are not obligated to share this verified packing list, however, the information in it might be more reliable than the information that customs received that is based on a planned packing list. In this flow of information, businesses share additional information with customs that they can use for risk assessment. As this is a situation in which we would like to support information sharing this information flow is a suitable case for this research.

In practice, UK Customs did not use the Customs dashboard by which they could access the information in the verified packing list in the Cassandra project (Lucassen et al., 2010). They made clear that they would have liked the pipeline to directly link to their declaration system so they can have the high-quality information in their risk assessment systems (Lucassen et al., 2010). This was not true for the first case with the actual flow of the packing list. Therefore, they found the customs dashboard via which they received the verified packing list of limited value and they did not use it in daily practice (Lucassen et al., 2010).

However, there is a flow of information between businesses and the declaration system of UK customs in the same trade lane, namely the flow of information in the ENS. We were curious why the verified packing list was not shared via the same information flow and expected that willingness could play a role. The second flow of information that we studied was one in which the verified packing list was shared via the same flow of information as the information in the ENS in the Yantian-Felixstowe trade lane. This second flow is a scenario and not a case, as the sharing of the verified packing list did not happen in this way in reality.

During performing this part of the research, some new ideas were developed on information sharing using events or thin information flows (van Engelenburg, Janssen, Klievink, et al., 2017). We were thinking about including such flows in our architecture as well and therefore we were interested in whether the thickness of an information flow would impact the willingness of businesses. We, therefore, included a third information flow in which the same verified packing list is shared using an event-based shipping information pipeline.

9.1.1.2.2 Cases from the CORE project

When the Cassandra project finished in 2014, the CORE project built further upon its findings (Zomer, Tan, & Hofman, 2014). In the CORE project, the operationalisation of a trusted trade lane supervision model was further investigated (Malenstein et al., 2014; Zomer, Tan, et al., 2014). The CORE project is broader than Cassandra and includes more than 70 partners, such as shippers, freight forwarders, shipping lines, customs, universities, IT vendors and consultants (Jensen, 2017). In the CORE-project, three supply chains are studied in which information is shared between businesses and with customs using different information sharing architectures.

In the case of Cassandra, we made a selection of the flows of information we were going to study on beforehand. This was useful, as it was at the start of investigating the context and we had to decide whether the method was suitable for investigating context. For this, we needed to get a somewhat deeper insight into the nature of the context.

However, the CORE project has a broader scope than the Cassandra project and, in the end, we wanted to get a broad view of what belongs to the context of B2G information sharing. The different architectures studied in CORE each support a variety of flows of information that could serve as cases in our research. Fully describing each of them is not feasible, considering their number. Secondly, the information to do so was not fully available. Third, this would risk us first investigating what a variety of information flows look like and later finding out that there is no information concerning willingness for some of them.

Furthermore, each of the architectures in CORE involved B2B information sharing as well as B2G information sharing in addition to required documents. The information flows in CORE were thus almost all in the scope of this research and therefore it was neither necessary nor efficient to select on beforehand. There were some G2G information flows included in CORE as well, and information flows involving other government organisations than Customs organisations within the European Union. These were out of scope for this research. However, we found that in practice, this was only a small portion of the information flows in CORE and it was easy to filter them out of the results during the analysis of the data.

To get a more broad perspective, in the end, we thus did not make an explicit selection and description of the information flows on beforehand. Instead, we chose to study the data gathered on the different architectures in CORE and select the parts that could say something about the willingness of businesses to participate in information flows (see section 9.1.1.3). In this way, we could work more efficiently and fully make use of the data that is useful for our research.

In the CORE project, different cases were studied. The unit of analysis in CORE was different from ours. In CORE, architectures were studied. In our study, the unit of analysis is at the level of the flows of information supported by the architectures. To discern the two, we will refer to the cases of CORE (so at the architectural level) as CORE cases. As some of the data in CORE is confidential, we cannot describe the CORE cases and the information flows they support in detail here. However, we can describe the

CORE cases at a higher level based on public information. An overview of the CORE cases can also be found in Rukanova, Henningsson, Henriksen and Tan (2018).

The first CORE case is the shipping information pipeline (SIP) case. In this case, a SIP developed by the sea-carrier Maersk and IBM is investigated. This SIP is a thin pipeline, including only events and links to shipping information instead of the shipping information itself (Rukanova et al., 2018). The scope of the SIP is global (Rukanova et al., 2018). The idea of this thin pipeline is that it connects with other systems, such as national hubs and thick pipelines (Rukanova et al., 2018).

The data on this case was collected by researchers in CORE that were involved in the coordination of the development of the SIP (Rukanova et al., 2018). Data gathering included communication via phone, email and face-to-face (Rukanova et al., 2018). As these researchers were actively involved in these developments, they collected data relevant to the willingness of businesses to participate in information flows in the SIP as well.

The second CORE case is the FloraHolland case. The focus of this CORE case is a thick information pipeline in which the actual shipping information is shared (Rukanova et al., 2018). The scope of this architecture is a specific trade lane (Rukanova et al., 2018). The trade lane studied in this CORE case is that of flowers that are shipped between Kenya and the Netherlands (Rukanova et al., 2018). FloraHolland is a flower auction house that is responsible for several of those trade lanes (Hulstijn, Hofman, Zomer, & Tan, 2016). They represent the growers of the flowers in Kenya in the trade lane and were involved in the information sharing process as well as the project (Rukanova, Huiden, & Tan, 2017).

The data collected on the FloraHolland CORE case was also performed by researchers who were involved in coordinating the development of the pipeline (Rukanova et al., 2018). This included continuous communication via phone, email and face-to-face (Rukanova et al., 2018). In addition, these researchers participated in key meetings and events and they used primary and secondary data on the CORE case (Rukanova et al., 2018).

The third CORE case is the Felixstowe case. This case involves four supply chains connecting to the port of Felixstowe, including the same supply chain as the Yantian-Felixstowe case in Cassandra (Hulstijn et al., 2016; Lucassen et al., 2010). The architecture that is studied in this CORE case is the same as the data pipeline in Cassandra and this can be viewed as further evaluation of this pipeline in the CORE project (Hulstijn et al., 2016; Lucassen et al., 2010). This data pipeline is thick and its scope is international (Rukanova et al., 2018). The thick data pipeline links to two national community hubs in the UK (a member of the European Union Customs Union at the time), namely the private hub Destin8 and the public hub OneGov (Rukanova et al., 2018).

The data on the Felixstowe CORE case was collected based on documentation, interviews and communication (i.e., e-mails, face-to-face meetings, conference calls) with the partners involved in the development of the pipeline (Rukanova et al., 2018).

9.1.1.3 Data collection

The appropriate sources of data on the willingness of businesses to participate in a flow of information can be identified by determining what sources could provide information on what impacts the focus of the willingness of businesses to participate in a flow of information (see section 6.1.2). Following the method, we selected two different types of parties that we could get relevant information from, namely businesses using information sharing architectures and designers of information sharing architectures.

The businesses themselves have deep insight into in which situations they are willing to share and in which situations they are not willing to share. Such deep insight into the motivation of businesses is not required to find context elements and relationships, as we merely need to know what impacts willingness (context elements) and how (context relationships). We do not need to know exactly why. However, to check whether nothing has been missed, gathering some data from businesses directly is necessary.

The insight of the designers into what impacts the willingness of businesses might be less deep, but broader as they have to work with a variety of businesses and consider their requirements. To consider these requirements, they need to have the exact same knowledge as we need, viz. what impacts willingness and how. Therefore, the bulk of data should be collected from these designers.

To collect data from businesses directly, we interviewed staff at a business involved in the Cassandra project. To collect data from the researchers and designers of the architectures, we interviewed one of the designers in the Cassandra project. In addition, we studied the documentation of the CORE project generated by the researchers in this project. We will discuss how we performed the interviews and studied the documents in the remainder of this section. In addition, we discuss how we analysed the data from the interviews and documents to derive the context relationships and context elements from them.

9.1.1.3.1 Interviews

The interview is an important source of case study information (Tellis, 1997; Walsham, 1995a; Yin, 1994). In concordance with the suggestions of Darke et al. (1998) on using interviews in case study research, we started out preparing for the interviews by studying the documentation on the Cassandra project. Based on knowledge of the case, we developed interview questions and made descriptions of information flows to ask the questions about. The interviews were conducted at the beginning of the research and therefore their purpose was mainly to explore what impacts the willingness of businesses to participate in an information flow. Therefore, we relied mainly on open questions, as these allow for a broad set of answers by the interviewees (Runeson & Höst, 2009).

The interviews were semi-structured. Semi-structured interviews allow the interviewer to ensure that all questions are asked, but at the same time allows for improvisation and exploration (Runeson & Höst, 2009). In the case of this research, this provided for the opportunity to explore further the different ways in which willingness might be impacted that might not have been expected on beforehand. The structure of the interviews was provided by describing an information flow and showing an image of it

to the interviewee. Subsequently, questions were asked to attain information about the willingness of businesses to participate in the information flow presented. In the first interview, we also included questions about the ability of businesses to participate in information sharing. However, due to using the new method for the second interview, the research had a more strict scope for the second interview and these questions were left out for that interview.

The interviews were conducted on the basis of an interview protocol. The interview protocol was tested in a pilot interview. Such a pilot test can be used to find weaknesses and further refine research questions (Turner III, 2010). This pilot interview was conducted with a colleague with extensive knowledge on the Cassandra project, as they participated in it as a researcher. Based on the pilot interview, the scenarios were further worked out and some of the questions were adapted.

First, the interviewees were asked for permission to record the interview and they were informed about the way in which such a recording and the information gathered would be used and how confidentiality would be protected. Next, the background of the research project was explained to the interviewees as well as the objective of the interview. To avoid the risk of misunderstandings, some key concepts (e.g., 'information flows') were defined as well. The first questions asked to the interviewees were their background and daily work activities to confirm that they have the knowledge relevant for this research.

The first interview was performed with a researcher involved in implementing the architecture in the Cassandra case. In her function in the project, she had to deal with the requirements of the businesses and thus has knowledge about what impacts their willingness. She could thus provide information on this. In this interview, we first presented the flow of information involving a verified packing list in Cassandra and asked questions about that. The second flow of information that the interviewee was questioned about was a scenario in which the verified packing list is shared via the same flow of information as the ENS.

The first interview was conducted in Dutch, as this is the mother tongue of both the interviewer and the interviewee. The setting of the interview was a communal area in the building where the interviewee currently worked. This setting was chosen to make it as convenient as possible for the interviewee to participate in the interview.

The second interview was conducted with a systems project analyst and a senior manager at a business involved in the Cassandra project. The systems project analyst could provide insight into what impacts willingness from a technical point of view, while the senior manager could do so from the interests of the business as a whole. The first flow of information presented to these interviewees was the same as for the first interview, namely the actual flow of information of the verified packing list. The second information flow, however, was different for the second interview. Namely, the verified packing list is shared in a thin information flow in this scenario. The reason for this change is that the answers of the first interviewee on the scenario were already quite clear and we wanted to know about the impact of using a thin information flow. We were not able to ask questions about both scenarios, due to time limitations. We also used the first interview to generate some hypotheses on things that would impact willingness. We added

questions to verify these hypotheses. However, we only asked about them if they were not brought up by the interviewees themselves.

The second interview was conducted in English, as the interviewer and the interviewees can speak this language. The interview was conducted via phone as the interviewer and interviewees were in other countries. While researchers often prefer face-to-face interviews, there is no evidence that interviews over a phone lead to lower quality data (Novick, 2008).

An important consideration in doing interviews is the level to which the interviewer directs the interviewee. Too much directing does not allow an interviewee to express their views and reduces the richness of the data collected (Walsham, 1995a). However, too little direction of the interviewer might lead the interviewee to believe that the interviewer is not really interested or that they are incompetent (Walsham, 1995a). The strategy for the interviews performed in this research was to find a good compromise. First, the interviewer let the interviewee respond to each question uninterruptedly and without further direction by the interviewer. The interviewer then tried to summarise the answer of the interviewee to determine whether she fully understood. Then, she asked follow up questions. Prior to performing the interviews, the interviewer followed a course on doing interviews that included learning how to provide different levels of direction and showing interest verbally as well as using body language.

We recorded the interviews. Recording interviews allows for having a full description of the responses of the interviewees and it does not require the interviewer to determine what is important during the interview, which might be difficult (Runeson & Höst, 2009; Walsham, 1995a). However, it might inhibit the interviewee and it might make the interviewer not participate fully in the interview process (Darke et al., 1998; Walsham, 1995a). To counter this, the interviewer promised the interviewees anonymity and to not share the recording with people outside of the research project. To ensure full participation of the interviewer, she tried to summarise the answers of the interviewee and check with them whether she fully understood their answer. Another disadvantage of recording interviews is that transcribing them and analysing the transcripts takes time (Walsham, 1995a). This was not an issue in this research, as only a limited number of interviews were performed.

9.1.1.3.2 Documentation

Another source of data in case studies are documents (Yin, 1994). In this research, we analysed the existing documents generated by the researchers in the CORE project. After obtaining agreement from different parties in the CORE project, we received the documents directly from the researchers in the CORE project. The documentation we studied of the CORE project was confidential for a large part. This means that here we can only provide a high-level overview of the documents we analysed as part of our case study.

We studied a broad set of documentation generated in CORE. For background information, we had access to the original research proposal of CORE. To collect data on the cases, we used three other types of documents from the CORE project. The first type of document and the most important source of information were reports on the progress

and the findings of the research at different stages for the different trade lanes. These reports are comprehensive and describe the progress of the research in detail. This includes descriptions of issues that the researchers came across and how they were solved and other considerations when making decisions on the design of the architectures in the different trade lanes. Therefore, these reports were a rich source of information on the way in which the researchers needed to consider different factors that influence what the flow of information supported by the architectures in the different trade lanes should look like, including considerations having to do with the willingness of businesses and reducing the risks of information sharing for them.

Another type of document that proved to be useful was a report of an ethics committee that investigated the relevant ethical and legal issues and proposed ways to address them in the projects. This analysis included subjects relevant to this research as well, such as the need for data confidentiality and the protection of trade secrets. In addition, we had access to a document containing notes on the problems the researchers encountered with information sharing. This list included issues that have to do with willingness. In these notes, solutions to the problems were proposed as well.

Bowen (2009) provides several ways in which the documents should be evaluated for the study. The researcher should establish that the documents are relevant for the purposes of the research and that they fit the conceptual framework (Bowen, 2009). We describe the relevance of the documents above per document type. In addition, as we describe in section 9.1.1.2, the cases in CORE are architectures in a trade lane, and that the cases in our case study are information flows. While the units of analysis are different, we view architectures as supporting information flows and therefore a study of these architectures can be expected to provide data on the information flows that these architectures support as well.

In addition to this, the general quality of the data in the documents needs to be assessed, such as authenticity, credibility and accuracy (Bowen, 2009). We had access to the original project proposal with a detailed description of the methods used in the CORE project. In addition, we had access to progress reports and thus could assess how these methods were applied. Furthermore, the CORE project is funded by the European Commission. Therefore, the project needs to meet their quality standards.

There are several benefits to including documents as a source of data in the research. Document analysis is efficient and cost-effective, and documents can provide a broad range of information on different cases (Bowen, 2009; Yin, 1994). This is an advantage in our case, as it allows us to get a broad view of the context that impacts the willingness of businesses to participate in an information flow. This is important, as our objective is to support information sharing in a variety of situations.

The documents were not generated for the purpose of this case study, and the analysis of the documents is thus a secondary analysis (Glass, 1976). There are limitations to using documents that were not generated for the purpose of the research. As the original purpose of the document might have required their writer to emphasise other things than required for answering the research question, details needed for the new research might be missing (Bowen, 2009; Runeson & Höst, 2009). We were in regular context with these

researchers in CORE and had the opportunity to ask them questions about the documents or request them for further information when we found such missing details.

9.1.1.4 Data analysis

There are various strategies for interpreting data from interviews as well as documents. In our case, we used the new method proposed in chapter 6 for analysing the data. More specifically, we wanted to determine what situations impact the willingness of businesses to share information (see section 6.1.2) and from the descriptions, we wanted to derive context elements (see section 6.1.3).

According to Yin (1994) data analysis in a case study consists of linking the data to the propositions and establishing the criteria for interpreting the findings. As discussed in section 9.1.1.1, the focus fulfils the role of propositions in the case study. Instead of linking the data to propositions, we thus had to link it to the focus of the willingness of businesses supplying information to participate in the information flow provided by the architecture.

The method provides a criterion for deciding whether something belongs to the relevant context or not (see p. 91). The data can be linked to the focus if the data describes a situation that meets the criterion. To test whether the criterion is met, a designer can try to fill out a table shell (table 6, p. 94) based on the information collected. If they succeed, then they have found a situation that restricts the focus.

We analysed the transcripts and the documents based on this procedure. We went through the data and each time we found a piece of text that could say something about the willingness of businesses to participate in a flow of information, we attempted to fill out the table shell. When we found a new context relationship, we entered its information in an Excel file. The Excel file contained the same columns as the table shell, so, a description of the restriction to the focus, a description of the situation restricting the focus and the support of the context relationship. For the latter, we referred back to the document or interview transcript from which we derived the data.

The statements in the data varied according to their specificity. In some cases, they describe very specific situations in a specific case. To derive a context relationship from such a specific statement, the first step was to describe the situation and restriction for the specific case in the Excel file. The next step was to generalise. We did this by replacing terms referring to specific objects (e.g., a name of a business) by terms referring to the group that the object belongs in (e.g., businesses), as described in section 6.1.3. To find the appropriate group, we determined why the objects are relevant to the focus.

For example, if a transport company is not a business in the information flow, clearly, their willingness does not impact the focus of whether businesses in the flow are willing to participate. Therefore, the transport company is relevant, because it is part of the flow of information. Thus, we replaced ‘transport company’ by the more general term ‘business involved in the information flow’. Furthermore, according to the data, a combination of certain data elements impacts the willingness of the transport company because sharing it would hurt their interests. This combination thus belongs to the group of sets data elements that are sensitive to the business.

After making generalisations in this manner for all the specific statements in the data, we compared the situations and their impact with each other. When these were very similar, or even the same, they were combined into one context relationship. Both statements then act as support for that relationship. When situations were very similar but did not lead to the same restriction to the focus, additional information could be gathered to find out why this is the case and the situations could be further refined. However, in practice, this did not happen.

It is important to note that if a restriction to the focus has more citations in its support, this does not necessarily mean that it is better substantiated. It could be the case, for instance, that the researchers in the documents refer to the same situation twice. It was sometimes hard to determine whether this is the case.

From the description of the generalised situations in the Excel file, we derived the context elements according to step 1.3 of the new method (see section 6.1.3). This means that we made a list of everything that we could state about objects in the situations. Then, we expressed this information using predicates. We added the predicates for the context elements to the Excel file as well.

9.1.2 The context of the willingness of businesses supplying information to participate in the information flow provided by the architecture

The data that was analysed for the focus of willingness contained 35 statements describing a situation in which the focus was restricted. 27 of these statements had to do with reducing the risks of data sharing in one way or the other. In some cases, these described several situations in which the focus is restricted.

As we describe in section 9.1.1.3, we generalised the situations and grouped the situations that are the same when generalised. The grouping of the statements resulted in the identification of 12 context relationships. They are presented below. The data used as support for the context relationships was confidential and therefore cannot be published. To protect the confidentiality of the data, we did not include the specific situations from which we generalised. Furthermore, we left out the support for the same reason. In the case study database, the situations below are recorded with the specific descriptions and their support in order to keep track of the evidence.

In this section, we provided an overview of the context relationships we found that impact the focus of willingness. We also already discuss what context elements are sensor elements and what context elements are adaptor elements. Identifying the sensor elements and adaptor elements is actually part of step 2 of the method. In this section, we will thus only identify them. In chapter 10, we will describe the sensors and adaptors that the systems require based on them, as well as the context rules that can be derived from the context relationships.

If we strictly follow the method, we also need to add predicates for the context relationships to state that there is data, there is a business, this business supplied data to the information flow, etc. While this is useful to derive the context elements from a situation description, it is very repetitive and can harm readability when including and

repeating these predicates for each of the context relationships. Instead, we named the arguments of the predicates in such a way that it is clear to what kind of object they refer.

In addition, many of the context relationships share context elements with each other. If a context element is expressed using the same predicate in different context relationships, then it is the same context element. The meaning of such a predicate is explained the first time the context element is mentioned. After this, each time the predicate is used it has the same meaning.

9.1.2.1 Not sharing sensitive data

The first context relationship we found seems quite obvious at first sight. We found that businesses do not want to participate in a flow of information if certain other parties can access their sensitive data. The reasons for the information being sensitive only seems to impact who cannot have access to the data (e.g., fear of being bypassed in the chain or customers seeing mistakes).

We can view data elements as the smallest unit of data that is still meaningful, such as a container number or a goods description. What is interesting is that sets or combinations of data elements are sensitive rather than individual data elements by themselves. This means that for the sensitivity predicate the argument concerning the sensitive data elements should be a set of one or more data elements.

We use a logic-programming paradigm to express the context elements. For example, Prolog allows for lists as arguments of predicates (Sterling & Shapiro, 1999). We can use these to represent sets of data elements. The only difference is that the lists in Prolog are ordered, while the set of data elements can be unordered. We need to ensure that the set of data elements is treated as an unordered list when it is included in a context rule. We will use `{...}` for unordered lists (i.e., sets) and `[...]` for ordered lists.

Another interesting result is that in all specific situations that are described by the data it is mentioned what other parties a business does not want to have access to their data, for example, their customers, a specific party, or competitors. Sensitivity thus seems to need to be represented by a ternary predicate with as arguments the set of data elements, the business that believes the set of data elements is sensitive and the party they do not want to have access to the data because it is sensitive. This is different from how ‘sensitivity’ is used in everyday natural language, which seems more natural to represent by a binary predicate.

sensitiveTo(SetofDataElements,Business,Party) expresses that set of data elements *SetofDataElements* is sensitive for business *Business* such that they do not want party *Party* to have access to these data elements. For example, consider a case in which we have a freight forwarder named ‘*A*’ and a carrier named ‘*B*’. *sensitiveTo({customername,containernumber},freightforwarderA,carrierB)* expresses that the set of data elements containing a customer name and a container number is sensitive according to freight forwarder *A* such that they do not want carrier *B* to have access. We cannot influence what data elements are sensitive to a business and this context element is therefore a sensor element.

The last context element of this context relationship is the access of a party to a set of data elements. *hasAccess(SetofDataElements,Party)* expresses that party

Party has access to the set of data elements *SetofDataElements*. As an example, $hasAccess(\{customernameFF, containernumber\}, carrierB)$ expresses that carrier *B* has access to a set of data elements containing the customer name of a freight forwarder *FF* and a container number. This last context element is an adaptor element, as it is the only context element of the context relationship that can be manipulated.

Restriction/ Situation	Context elements	Adaptor/ Sensor
Restriction: Business is not willing	$sensitiveTo\left(\begin{matrix} SetofDataElements, \\ Business, Party \end{matrix}\right)$	Sensor
A set of data elements is sensitive to the business from another party. The party that the data elements are sensitive from, gets access to them.	$hasAccess\left(\begin{matrix} SetofDataElements, \\ Party \end{matrix}\right)$	Adaptor

Table 12: The context relationship ‘Not sharing sensitive data’

9.1.2.2 Encrypt sensitive data

The second context relationship that we found is a positive one. According to this context relationship, businesses are willing to share their data, even if it is sensitive, when only parties that they believe are entitled to access the data can have access. This means that not only the party that the data is sensitive to should not be able to obtain a key but all parties that the businesses consider not entitled to access to the data.

$encrypted(SetofDataElements, Key)$ expresses that the sensitive data elements in *SetofDataElements* are encrypted and that *Key* is the key to decrypt the data. There are two possibilities here. The first is that the data elements are already encrypted before they are shared. In that case, this should be a sensor element. However, the architecture could also send the data elements via a component that encrypts the sensitive data elements. In that case, it is an adaptor element. There seems no clear objection to have a sensor as well as an adaptor for this context element, so this context element will be both.

$\neg entitled(access, SetofDataElements, Business, Party)$ expresses that party *Party* is not entitled to have access to the data elements in *SetofDataElements*, according to business *Business*. The data from our case study suggests that this is always the case unless there is a clear benefit for the business if the party can have access to the data, for example, when the party can use the data to improve the logistics process, or if they pay a fee to access the data. This context element is a sensor element, as it cannot be manipulated.

A key can be shared just like any other data. Moreover, just like for other sets of data elements, it is possible to manipulate whether a certain party has access to a set of data elements with only a key.

Restriction/ Situation	Context elements	Adaptor/ Sensor
Restriction: Business is willing A set of data elements is sensitive to the business from another party. The sensitive data elements are encrypted. Only parties that are entitled to access can decrypt the data elements.	$sensitiveTo\left(\begin{matrix} SetofDataElements, \\ Business, Party_1 \end{matrix}\right)$	Sensor
	$encrypted(SetofDataElements, Key)$	Sensor/Adaptor
	$\neg entitled\left(\begin{matrix} access, \\ SetofDataElements, \\ Business, Party_2 \end{matrix}\right)$	Sensor
	$\neg hasAccess(\{Key\}, Party_2)$	Adaptor

Table 13: The context relationship ‘Encrypt sensitive data’

9.1.2.3 No aggregating sensitive data

According to the case study, in some cases, data can be sensitive to businesses only when it is aggregated. If data is aggregated, it might be analysed using big data techniques and these results might be sensitive to a business. For example, one goods description of goods and their quantity shipped in one container might not be sensitive to a business. However, if this data is aggregated, then the volume that the business produces or ships might be derived from that. This can be sensitive as competitors might misuse this data.

The context element of $sensitiveToAggregate\left(\begin{matrix} SetofDataElements, \\ Business, Party \end{matrix}\right)$ is very similar to the context element expressing that a set of data elements is sensitive. In fact, aggregated data elements are a set of data elements as well. However, based on the analysis we found a difference in the impact of combinations of data elements that are always sensitive and those that are only sensitive when aggregated. Therefore, we identified them as separate context elements. $canAggregate\left(\begin{matrix} Party, \\ SetofDataElements \end{matrix}\right)$ in its turn, is similar to the context element of what data can be accessed by parties. However, in this case, it means that *Party* is not allowed to aggregate, or store, *SetofDataElements*.

Restriction/ Situation	Context elements	Adaptor/ Sensor
Restriction: Business is not willing Data elements are sensitive to the business from another party when aggregated. The party that the data elements are sensitive from, when aggregated, can aggregate them.	$sensitiveToAggregate\left(\begin{smallmatrix} SetofDataElements, \\ Business, Party \end{smallmatrix}\right)$	Sensor
	$canAggregate(Party, SetofDataElements)$	Adaptor

Table 14: The context relationship ‘No aggregating sensitive data’

9.1.2.4 No aggregating for customs

When a business considers a set of data elements sensitive when aggregated, they do not want government organisations to aggregate the data, if they do not provide a reason. This is because they fear the data falling in the wrong hands. In this case, it seems that the business decides what is a good reason and when customs is entitled to aggregate the data.

Restriction/ Situation	Context elements	Adaptor/ Sensor
Restriction: Business is not willing Data elements are sensitive to the business from another party when aggregated. Customs is not entitled to aggregate the data elements. Customs can aggregate the data elements.	$sensitiveToAggregate\left(\begin{smallmatrix} SetofDataElements, \\ Business, Party_1 \end{smallmatrix}\right)$	Sensor
	$partyType(customs, Party_2)$	Sensor
	$\neg entitled\left(\begin{smallmatrix} aggregate, SetofDataElements, \\ Business, Party_2 \end{smallmatrix}\right)$	Sensor
	$canAggregate(Party_2, SetofDataElements)$	Adaptor

Table 15: The context relationship ‘No aggregating for customs’

As we only found in the data that this is the case of customs, whether a party is a customs organisation is a context element as well. This context element is expressed by $partyType(customs, Party)$. It is a sensor element, as it cannot be manipulated.

9.1.2.5 Only viewing sensitive data for customs

In the case study, we found that businesses were willing to let customs view data that is sensitive to them when aggregated, as long as customs cannot aggregate the data. Viewing the data means that customs can view the data in a dashboard, for example. However, it means that they are cannot download the data store it.

$view(Party, SetofDataElements)$ expresses that $Party$ can view $SetofDataElements$. This is an adaptor element, as the context-aware architecture could share the data via a system that allows customs or another party to view the data, without storing it in their system.

Restriction/ Situation	Context elements	Adaptor/ Sensor
Restriction: Business is willing Data elements are sensitive to the business from another party when aggregated. Customs cannot aggregate the data elements. Customs can view the data elements.	$sensitiveToAggregate\left(\begin{matrix} SetofDataElements, \\ Business, Party_1 \end{matrix}\right)$	Sensor
	$partyType(customs, Party_2)$	Sensor
	$\neg canAggregate\left(\begin{matrix} Party_2, \\ SetofDataElements \end{matrix}\right)$	Adaptor
	$view(Party_2, SetofDataElements)$	Adaptor

Table 16: The context relationship ‘Only viewing sensitive data for customs’

9.1.2.6 Not broadcasting sensitive data

From the case study, we derived that businesses are not willing to share their sensitive data via a system that broadcasts the data. $broadcasts(System)$ expresses that system $System$ broadcasts the data it has access to in the information flows (e.g., some event-based data pipelines). Such a system could connect to the context-aware architecture. It is not possible to manipulate directly whether it broadcasts data, as this likely is part of the functionality the system provides. Therefore, this context element is a sensor element.

The predicate $\neg hasAccessSystem(SetofDataElements, System)$ is similar to $\neg hasAccess(SetofDataElements, Business)$. The difference is, however, that for the last context element it is the business that does not have access to $SetofDataElements$ and for the first context element, it is a system.

Restriction/ Situation	Context elements	Adaptor/ Sensor
Restriction: Business is not willing	$sensitiveTo\left(\begin{smallmatrix} SetofDataElements, \\ Business, Party \end{smallmatrix}\right)$	Sensor
Data elements are sensitive to the business from another party.	$broadcasts(System)$	Sensor
There is a system in the flow of information that broadcasts the data elements.	$hasAccessSystem\left(\begin{smallmatrix} SetofDataElements, \\ System \end{smallmatrix}\right)$	Adaptor

Table 17: The context relationship ‘Not broadcasting sensitive data’

9.1.2.7 No thick, global pipeline

According to the data from the case study, businesses are not willing to use a data pipeline when its geographical scope is global and the flow of information is thick. A thick information flow contains the actual documents and shipping information. It is a risk for businesses to share this in a global data pipeline that can be used by businesses from a variety of supply chains, as controlling access to such a global pipeline is too complex and difficult to arrange.

$systemType(System, datapipeline)$ expresses that system *System* is a data pipeline. This cannot be manipulated and therefore is a sensor element.

$thick(SetofDataElements)$ expresses that *SetofDataElements* and thus contains actual shipping information or information from documents. As we will discuss for context relationships presented later, it is possible to include a component in the architecture to make thick data sets thin. However, this is not possible the other way around. Thus, this context element is a sensor.

$geographicalScope(System, global)$ expresses that the geographical scope of system *System* is global.

Restriction/ Situation	Context elements	Adaptor/ Sensor
Restriction: Business is not willing	$hasAccessSystem\left(\begin{smallmatrix} SetofDataElements, \\ System \end{smallmatrix}\right)$	Adaptor
A data pipeline is part of the flow of information.	$systemType(System, datapipeline)$	Sensor
The flow of information is thick.	$thick(SetofDataElements)$	Sensor
The flow of information is global.	$geographicalScope(System, global)$	Sensor

Table 18: The context relationship ‘No thick, global pipeline’

9.1.2.8 Thick international pipeline

In contrast with the previous context relationship, businesses are willing to share their data in a thick information flow using a data pipeline with an international scope. However, this does mean that they need access to their data to be controlled in the data pipeline. Controlling access is also less complex and easier to arrange in a data pipeline with an international scope.

controlsAccess(System) expresses that system *System* controls access to the data that it is used for.

Restriction/ Situation	Context elements	Adaptor/ Sensor
Restriction: Business is willing A data pipeline is part of the flow of information. The flow of information is thick. The flow of information is international. Access to data is controlled in the data pipeline.	$hasAccessSystem\left(\begin{smallmatrix} SetofDataElements, \\ System \end{smallmatrix}\right)$	Adaptor
	$systemType(System, datapipeline)$	Sensor
	$thick(SetofDataElements)$	Sensor
	$geographicalScope\left(\begin{smallmatrix} System, \\ international \end{smallmatrix}\right)$	Sensor
	$controlsDataAccess(System)$	Sensor

Table 19: The context relationship ‘Thick international pipeline’

9.1.2.9 Thin global pipeline

A thin information flow does not contain the actual shipping data that is shared, but it contains links to this shipping information. Businesses are willing to share their data in a global pipeline in a thin information flow.

thin(SetofDataElements) expresses that *SetofDataElements* is thin. A set of data elements can be because the business that shares the data has made it thin before sharing it. In that case, this context element should be sensed. However, it is also possible to make a thick data set thin. In that case, this context element is an adaptor element. Just as for the context element of the encryption of data, there is no clear obstacle to include both.

Restriction/ Situation	Context elements	Adaptor/ Sensor
Restriction: Business is willing	$hasAccessSystem\left(\begin{smallmatrix} SetofDataElements, \\ System \end{smallmatrix}\right)$	Adaptor
A data pipeline is part of the flow of information.	$systemType(System, datapipeline)$	Sensor
The flow of information is thin.	$thin(SetofDataElements)$	Sensor/adaptor
The flow of information is global.	$geographicalScope(System, global)$	Sensor

Table 20: The context relationship ‘Thin global pipeline’

9.1.2.10 Check system security

In one of the cases, we found that a business was only willing to share data that was sensitive to them if the systems in the flow of information had been subject to security checks and were secure enough according to them. Otherwise, they were afraid that these systems could leak their sensitive data.

Restriction/ Situation	Context elements	Adaptor/ Sensor
Restriction: Business is willing	$sensitiveTo\left(\begin{smallmatrix} SetofDataElements, \\ Business, Party \end{smallmatrix}\right)$	Sensor
Data elements are sensitive to the business from another party.	$hasAccessSystem\left(\begin{smallmatrix} SetofDataElements, \\ System \end{smallmatrix}\right)$	Adaptor
The security of the systems in the flow of information has been checked to be according to the level required by the business.	$securityRequirement\left(\begin{smallmatrix} Business, \\ SetofDataElements, \\ Level_1 \end{smallmatrix}\right)$	Sensor
	$securityProvided(System, Level_2)$	Sensor
	$lower(Level_1, Level_2)$	Computation

Table 21: The context relationship ‘Check system security’

securityRequirement(Business, SetofDataElements, Level) expresses that business *Business* requires level of security *Level* for the systems with which *SetofDataElements* is shared. We cannot manipulate this, so this is a sensor element.

securityProvided(System, Level) expresses that system *System* has level *Level* of security. Of course, this information should be provided by a reliable source that has assessed the security of the system. We will discuss this in further detail when we discuss the sensor for sensing this context element.

lower(Level1, Level2) expresses that *Level1* is a lower level of security that *Level2*. When the levels of security are standardised, then this can be computed based on an ordered list with systems.

9.1.2.11 Connection to shipment

Businesses do not want to share any data with parties that are not connected to the shipment about which that data is.

subject(SetofDataElements, Shipment) expresses that the subject of *SetofDataElements* is shipment *Shipment*. This is a sensor element, as it cannot be manipulated.

\neg *connection(Party, Shipment)* expresses that *Party* is not connected to *Shipment*. This cannot be manipulated as well, and thus is a sensor element as well.

Restriction/ Situation	Context elements	Adaptor/ Sensor
Restriction: Business is not willing The data elements in the flow of information are about a shipment. A party has access to the data elements. This party is not connected to the shipment.	<i>subject(SetofDataElements, Shipment)</i>	Sensor
	\neg <i>connection(Party, Shipment)</i>	Sensor
	<i>hasAccess(SetofDataElements, Party)</i>	Adaptor

Table 22: The context relationship ‘Connection to shipment’

9.1.2.12 Filter data

If a certain combination of data elements is sensitive to a business, then they are willing to share it, if some of these data elements are filtered from the data set. Such filtering of data could lead the data to be anonymised. Anonymization of data falls under this context relationship as well.

filter(SetofDataElements, FilteredSetofDataElements, Business, Party) expresses that *FilteredSetofDataElements* is a subset of data elements *SetofDataElements* such that it is not sensitive for *Business* to *Party*. The

architecture can include a component that could filter the data and thus manipulate this context element. It is thus an adaptor element.

Restriction/ Situation	Context elements	Adaptor/ Sensor
Restriction: Business is willing Data elements are sensitive to the business from another party. The party that the data is sensitive from can only access this set of data elements after they are filtered.	$sensitiveTo \left(\begin{matrix} SetofDataElements, \\ Business, Party \end{matrix} \right)$	Sensor
	$filter \left(\begin{matrix} SetofDataElements, \\ FilteredSetofDataElements, Business \end{matrix} \right)$	Adaptor
	$hasAccess(FilteredSetofDataElements, Party)$	Adaptor

Table 23: The context relationship ‘Filter data’

9.2 Focus 2: The lawfulness of the information flow provided by the architecture

In this section, we first discuss the methods we used to collect and analyse data on the context of the focus of the lawfulness of information flows. Next, we discuss the different fields of law that we included in the study and their relevance to this focus. Then, we present the context elements and context relationships that we identified.

9.2.1 Methods

In section 8.3.2.2, we discussed our choice of methods to use for investigating the context of the lawfulness of information flows. In this section, we discuss in more detail how we applied those methods. We obtained information on the context in two phases. Each of the phases is discussed in a subsection of this section.

9.2.1.1 Phase 1: Scoping and exploration

Before investigating the context of the lawfulness of information flows, we performed the case study on the context of the willingness of businesses to participate in information flows. We found that protecting competitive and commercially sensitive data was a big concern for businesses. This means that such information is shared in supply chains, as otherwise, these concerns would not have come up in the cases we studied. Furthermore, it means that businesses will take measures to arrange such juridical protection (e.g., using contracts). This will make certain flows of information lawful or unlawful. For arranging

such protection, Intellectual Property (IP) law is important. Furthermore, competition law restricts with whom competitively sensitive data can be shared. Businesses are obligated to share certain information with customs. In some cases, these obligations overrule other juridical concerns that otherwise would make sharing unlawful. These obligations are established in customs law. This makes customs law relevant to include in our study as well.

We considered including data protection law in our analysis as well. However, during a workshop at Maersk Line, it was indicated by their juridical experts that the only personal data that is currently shared in supply chains where they are involved are the names of employees that sign documents. Data protection law is only applicable to personal data. This was also confirmed in an interview with a data protection expert employed by Maersk. Furthermore, in the CORE and Cassandra documents, no other personal data was mentioned. Considering the minor role of personal data in the overall information sharing process, the decision was made not to include it in our study.

To further confirm that the fields of customs law, competition law and IP law are indeed most important to consider in our study, in phase 2, we asked during expert interviews with experts from the fields these fields what other fields they considered most important for determining the lawfulness of the flows of information. These experts did not mention additional fields of law that they considered highly important as well.

To determine what flows of information are and are not lawful according to the legislator, it is possible to look directly at the law. We obtained advice on what articles in what laws to look at from a juridical expert that is involved in the project this research is a part of. She advised to look at the Union Customs Code art. 13 and 127 (The European Parliament and the Council of the European Union, 2013) to obtain information on customs law. For IP law, she advised looking at Directive (EU) 2016/943 of the European Parliament and of the Council of 8 June 2016 on the protection of undisclosed know-how and business information (trade secrets) against their unlawful acquisition, use and disclosure, art. 4.2 and 4.3 and 5 (The European Parliament and the Council of the European Union, 2016).

We used the new method again for deriving context relationships from the articles. This means that we tried to fill out the table shell on p. 91 again. If we succeeded, we added a description of the restriction to the focus, a description of the situation and the support of the context relationship to the Excel file. In addition, we added some situations and restrictions in direct cooperation with the juridical experts involved in our project. The result of the exploration phase was a list of 10 context relationships that we used as input for the second phase.

9.2.1.2 Phase 2: Testing and refining of the context relationships

In phase 2, we tested the context relationships identified in phase 1. Furthermore, we further refined them. To do so, we performed scenario-based expert interviews. In this section, we describe in detail how we performed these interviews and how we analysed the information we gained from them.

9.2.1.2.1 Experts to interview

According to Meuser and Nagel (2009), who is suitable as an expert in an expert interview is decided by the researcher based on their objective and the recognition of the experts as an expert within their own field. In this part of the research, we want to determine what impacts the lawfulness of B2G information sharing flows. At the beginning of the research (see section 8.3.2.2), we made a selection of juridical fields that we consider most important for answering this question. The fields that we selected were customs law, competition law and IP law. For the expert interviews, we thus require the participation from experts within these fields.

Someone can be recognised as an expert in a juridical field in several ways. First, usually, academics within a certain field are considered experts within their field. They could provide information on what impacts the lawfulness of information flows from a more theoretical perspective and taking into account the newest developments and insights. In addition, businesses and customs organisations often employ juridical experts to provide them with advice on juridical issues, including those concerning information sharing. These jurists typically are highly educated and specialised in a juridical field. Furthermore, they have experience with interpreting the law in practice and they know what considerations and circumstances to take into account when interpreting the law in practice. We thus interviewed juridical experts from academia, as well as juridical experts working in industry.

Within the EU, everybody has to adhere largely to the same laws. The law is supposed to be the same for everybody in the same situation. This means that in general, it will be clear when information flows are lawful and unlawful and there will not be many different opinions on that. Of course, there are exceptions when the law is unclear and there is not a lot of legislation available. However, as our research is not of a juridical nature, it is limited to finding situations in which the restriction to lawfulness is clear-cut. This means that it is not necessary to obtain information from a variety of experts per domain as we can assume that in general, they will agree on when a flow of information is lawful. The cases in which it is not clear to experts whether information sharing is lawful should be subject to research in the juridical domain. These outcomes can then be added as new context relationships.

While interviewing one expert per juridical domain might be enough to determine the lawfulness of information flows, we tried to consult multiple experts per domain. The main reason for this is that, in practice, it is hard to gather all data in the limited time that the experts have available (between 1 and 2 hours). Furthermore, as matters can be quite complex, analysis of an interview might be needed to point out things that are not yet clear enough and that should be discussed in further depth. Additional interviews provide the opportunity to do so.

Date of interview	Setting	Interviewee pseudonym	Juridical field	Background	Function
2-11-2017	Meeting room	Interviewee 1	Customs law and tax law	Academia	Professor
13-11-2017	Phone	Interviewee 2	IT law and IP law	Business	Senior Legal Counsel
21-11-2017	Phone	Interviewee 3	IP law	Academia	Professor
				Business	Lawyer
17-1-2018	Meeting room	Interviewee 4	Customs law	Academia	Teacher and program director
				Customs	Senior policy advisor
26-2-2018	Meeting room	Interviewee 5	Customs law and VAT law	Business	International policy director
		Interviewee 6	Transport law and commercial law	Academia	Scientific teacher
				Business	Jurist
		Interviewee 7	Expert in supply chain management (not a juridical expert)	Business	Project manager logistic cooperation
9-3-2018	Meeting room	Interviewee 8	Competition law	Business	Lawyer

Table 24: Interviews with different juridical experts

Table 24 provides an overview of the interviews and the experts we interviewed. Most of the experts were approached based on the recommendation of members of the juridical team of the project that this research is a part of, or by experts in previous interviews. All interviews were conducted in English. The settings for the interviews were chosen to be most convenient for the interviewee, usually at their place of employment and otherwise via phone. One of the interviews was conducted with several interviewees at the same time, as this was preferred by one of these interviewees. This actually lead to a more broad discussion that was useful for getting a more broad overview of the practical considerations that play a role in determining the lawfulness of information flows. Only one expert in competition law was interviewed. However, this expert had expertise precisely in information sharing between businesses in international supply chains and with customs. Furthermore, they had allocated an extensive amount of time to participate in the interview. Therefore, this interview provided us with sufficient information to answer our questions.

9.2.1.2.2 Scenarios

An issue with interviewing an expert that can arise when the expert and the interviewer have different backgrounds is that it might be hard to understand each other and this might interfere with the quality of the data collected. Understanding goes two ways. First, the interviewer should ensure that they understand the information the expert provides to them. For the interviewer, it is important to be informed in advance about the important rules and principles that play a role in the context of the expert (Meuser & Nagel, 2009). In this research, we ensured this in several ways. The interviewer participated in a course at the bachelor's level in which the basics of technology and law were taught. Furthermore, she participated in a workshop on normware in which the relationship between the law and technology was subject to in-depth discussions. In addition, she prepared the interviews by studying relevant sources of law on beforehand in the exploration phase. To help bridge the gap between fields during the interviews, most of them were done in the presence of a juridical expert involved in the project and thus familiar with the research.

The challenge the other way around is to ensure that the juridical experts understand the technical details and concepts involved with different flows of information that might impact their lawfulness. A strategy for facilitating communication of possible solutions and concerns with a variety of parties from the design field is by using scenarios (Rosson & Carroll, 2002). Scenarios are stories involving a setting and situations and actors, and various tools and objects that can be manipulated by the actors (Rosson & Carroll, 2002). In our case, the scenarios are flows of information between businesses and with customs in the container-shipping domain.

A major advantage of using scenarios in our case is that it provides the experts with concrete information flows for which they can say something about their lawfulness based on their expertise. Furthermore, it provides examples of how different technologies that could be incorporated in the context-aware architecture work, such as data pipelines or blockchain technology. This might make it easier for the experts to understand these technologies and to determine the effect of using them on the lawfulness of information flows. The scenarios thus provide concrete material and such material is often interpreted more easily and thorough than abstract materials (Rosson & Carroll, 2002).

We generated the scenarios for the interviews based on the context relationships found in the exploration phase. We established two sets of two scenarios. Each set contains a problem scenario and a solution scenario. The problem scenarios were designed such that we expect them to be unlawful according to the context relationships found during the exploration. In this way, in the two problem scenarios, all context relationships that might make flows of information unlawful were tested.

The solution scenarios were each an adaptation of the problem scenario. The adaptations were such that we would expect the sharing of the same information as in the problem scenario to be lawful in the solution scenario according to the context relationships. In this way, positive context relationships that might make flows of information lawful were tested. Furthermore, this also allowed to test already some ways in which context elements might be manipulated and to test some options for the design of the new architecture.

In our previous research on the willingness of businesses to share information, including the design of previous versions of the architecture (see section 8.1), we found an effect of an information flow being open or closed and thick or thin (van Engelenburg, Janssen, Klievink, et al., 2017, 2018). This means that the architecture should support information flows that vary on these dimensions. Furthermore, varying these dimensions could be a way for the architecture to adapt and ensure that information sharing is lawful. For example, encrypting data and making the information flow less open could ensure that competitors do not have access to each other's competitively sensitive information. Another basic variable for information sharing architectures seems to be whether they are distributed or centralised, which should be considered as well. Taking into account all these considerations, we generated four scenarios. An overview of these scenarios is presented in table 25.

Scenario	Expected lawfulness	Type of information flow
1.1	Unlawful	Distributed, open, thick
1.2	Lawful	Distributed, closed, thick
2.1	Unlawful	Centralised, open, thick
2.2	Lawful	Centralised, open, thin

Table 25: Overview of scenarios used in the expert interviews

9.2.1.2.3 Type of interview and procedure

The procedure for conducting the interviews with the juridical experts was similar in many ways to that of the interviews that are part of the case study into the context of the willingness of businesses to share information (see section 9.1.1.3). This is not surprising, as in both cases we want to get insight into context concerning in the same domain for designing the same architecture. This resulted in often making similar choices for how to conduct the interview based on similar considerations.

The expert interviews were semi-structured for the same reason as the previous interviews for the focus concerning willingness. Namely, we wanted to ensure that all questions are asked and all areas are covered, while at the same time allowing for improvisation and further exploration. The latter was necessary here as well, as we expected to find context relationships in addition to those generated in the exploration phase. Furthermore, we wanted to use the expert interviews to check for possible ways for the architecture to adapt and to sense context information. This means that we needed to be able to improvise and ask additional questions about newly discovered context relationships.

The interviews were conducted based on an interview protocol. The interview protocol was presented and discussed with a juridical expert in our research project several times to ensure the quality of the protocol. In addition, we asked her to assess the intelligibility of the scenarios and questions to juridical experts without a technical background.

Similar to the previous interviews in this research, first, the interviewees were asked for permission to record the interview. They were also informed about the way in which such a recording and the information gathered would be used and how

confidentiality would be protected. Then the interviewer explained the background of the research project to the experts and the objective of the interview. Some key concepts (e.g., ‘information flow’) were defined as well.

The experts were asked to describe their background and area of expertise. This helped us to confirm the extent of their expertise and its relevance to answering the research question. The rest of the interview was structured by presenting scenarios and asking questions about this.

The scenarios were described in detail to each of the experts. Furthermore, they were shown images of the scenarios and these were used to discuss each step of information sharing the flow. The interviewees were asked whether they fully understood the scenarios and encouraged to ask questions if something was not clear.

After explaining the scenarios, the interviewees were asked questions about them. For a non-expert, it is sometimes hard to determine where the expertise of an expert begins and ends. We thus used the same interview protocol for each of the experts containing the same questions. We asked the experts to assess whether they could answer each question from their expertise. If they indicated that they could not, then we moved on to the next question.

There were two sets of questions for each of the scenarios. The first set was open and general questions about the scenario. First, we asked the expert to determine what fields of law would be relevant to assess the lawfulness of the information flow. The purpose of this question was to confirm that we did not miss areas of the law in the research that could have a major impact on the lawfulness of the information flows. We then asked the expert whether they believe the flow of information to be lawful and, if not, how they would solve this. The purpose of these questions is to allow the experts to discuss openly the lawfulness of the flow of information and their considerations for assessing this lawfulness. In this way, new context relationships could be identified. In addition, some questions were asked about the properties of objects in the flow of information, such as parties, systems and the goods that the information is about. The purpose of these questions was to stimulate the expert to further discuss what they take into account in their considerations and why.

The next set of questions was meant to check whether the lawfulness of the information flow corresponded to what was expected based on the context relationships from the exploration phase. When the lawfulness of the information flow corresponded with what was expected according to a context relationship, we tried to confirm that this was actually due to the relationship with the situation described in the context relationship. When the lawfulness of the flow of information did not correspond with what was expected based on the context relationships, we tried to explain the difference and use this to refine the context relationships. In addition, we asked questions on how and where to measure different context elements. The interview protocol contained an extensive overview of the context relationships, the expected lawfulness based on them and why, and questions to ask about them so that the reviewer could keep track of everything and ensure that all questions were answered.

We ended the interviews by asking the experts for additional general advice and recommendations. Furthermore, we asked them suggestions for other experts to contact

and interview. Of course, we expressed our gratitude for cooperating in the interviews as well.

For these expert interviews, we provided the same level of direction as in the previous interviews in the same way. First, the interviewer let the interviewee respond to each question uninterruptedly and without interference. The interviewer then summarised the answer of the expert to determine whether she fully understood. Then, she asked follow-up questions if she considered this useful.

We recorded the interviews with the experts. We did so for the same reasons as the previous interviews. Namely, recording interviews allows for having a full description of the responses of the interviewees and it does not require the interviewer to determine what is important during the interview, which might be impossible (Runeson & Höst, 2009; Walsham, 1995a). However, it might inhibit the interviewee and it might make the interviewer not participate fully in the interview process (Darke et al., 1998; Walsham, 1995a). To counter the possible effect of inhibiting the expert in their answers, interviewer promised the interviewee anonymity and to not share the recording with people outside of the research project (Darke et al., 1998; Walsham, 1995a). To ensure full participation of the interviewer, she tried to summarise the answers of the interviewee and check with them whether she fully understood their answer (Darke et al., 1998; Walsham, 1995a). Another disadvantage of recording interviews already mentioned previously is that transcribing them and analysing the transcripts takes time (Walsham, 1995a). As we only performed a limited number of interviews, it was feasible to transcribe them.

9.2.1.2.4 Analysis of the transcripts

We used the new method to analyse the transcripts of the expert interviews in a similar way as we did for the previous interviews (see section 9.1.1.4). This means that we tried to fill out the table shells for new context relationships and if we succeeded, we added the new context relationship and its support to an Excel file. We generalised in the same manner as well. Thus, we tried to see what situation descriptions from the different interviews were the same when generalised and then added matches as support to a context relationship. In this way, we identified new context relationships.

In addition, we assessed for each of the context relationships from the exploration phase whether they are confirmed or should be refined based on the new information. If necessary, we further refined these context relationships. We kept track of these refinements and their support in the Excel file as well.

In addition, we searched the transcripts for information provided by the experts on how context elements could be sensed or adapted to. We made an overview of all statements by the experts on different possible adaptors and sensors. For the sensors, we made a distinction between the way in which different context elements are defined and statements on how they can be measured. For the statements on adaptors, we kept track of statements on to what juridical issues they could help deal with in different situations and the requirements that need to be met for doing so. This information was input for the next activity in the research in which the sensors and adaptors for the context-aware architecture are generated. It was input as well for the evaluation of the architecture, as it provides insight into the extent to which the architecture can support lawful information

sharing by making an adaptation. We discuss the use of this data in further detail in section 8.4 and section 8.6.

9.2.2 Lawfulness and different fields of law

According to an expert we interviewed from our project, a party can be unlawful, negligent or in breach of a contract. In the first case, they directly violate a rule in the law. In the second case, a party is liable for something because they are negligent. In the last case, the party entered a contract with another party and they did not follow a promise in the contract. While this distinction is important to determine who is liable in case something goes wrong in the information sharing process, in our case we are concerned with how to support information sharing in a way to prohibit this from happening. Establishing who is liable is of a juridical nature. This should be left for experts in the juridical field. For this research, this is out of scope. For getting the insight into context we require to design the architecture, we thus do not need this distinction and we will refer to all three cases as being ‘unlawful’ to enhance simplicity.

We start with a general discussion of the fields of law for which we interviewed the experts. We asked the experts about what other fields of law were applicable according to them for determining the lawfulness of information flows and in some cases, we discussed the relationships between the field of their juridical expertise with other fields. Such a general discussion could support a better understanding of the context relationships.

9.2.2.1 Data protection law

At the beginning of the research, we considered to include data protection law in our investigations of context. However, in the end, we decided not to include it, as the impact of data protection law on the lawfulness of information flows in international container shipping is very limited. We conducted an interview with an expert in data protection law to confirm this. Furthermore, we asked the other experts about the relevance of other fields of law and the subject of protecting personal data came up several times.

According to the experts we interviewed, data protection law is only applicable to personal data, which is data that can be linked to a natural person. Data is considered personal even if it takes some effort to make this link. Certain documents can contain personal data only in certain circumstances. For example, in general, a packing list will not be considered to contain personal data unless it involves goods from a natural person instead of a business, or a trader involved is a natural person instead of a business.

We interviewed an expert in data protection law with a background in industry. This interview is not included in section 9.2.1.2.1, as we did not use it to derive context relationships. However, the interview followed the same procedure as the other interviews. According to this expert, the cases in which personal data is shared in international container shipping are very limited. This is why we did not derive context relationships from this interview or conduct additional interviews concerning data protection law.

Some of the other juridical experts mentioned data protection law as well. According to the expert in IP law with a background in international container shipping, there is some personal data shared, such as names, contact information and IP addresses, but this is limited. Usually, this data is shared by the person himself or herself and therefore consent is usually not an issue. In addition, usually, such personal data is shared by the person in their capacity as an employee, which means that sharing is subject to the agreements between the companies. Furthermore, this data usually is not sensitive. The expert in IP law from academia also stated that, at least for documents such as the ENS, things like addresses of the shipper could be personal data. However, usually, this will not be the case as these are businesses.

The competition law expert suggested that there might be privacy issues with Customs seeing customer information if this is not necessary for their task. However, he also states that he is not really aware of such privacy issues in practice.

The customs law expert at customs did view data protection law as a concern. According to this expert, Authorised Economic Operators (AEOs) need to be able to show which employees they have in order to be licensed as an AEO. This data is personal. He states that if customs wants this data, then they will ask the businesses for it. When they do so, then it is to determine whether the business has met their obligations as an AEO. This provides customs with a legal basis to ask the information and it provides the business with a legal basis to share the information with customs. In addition, the juridical expert in our project mentioned that sharing personal data seems to become more important for reasons of security. For example, information on truck drivers might be shared to prevent the stealing of goods.

This last result shows that even though the sharing of personal data is limited in the international container-shipping domain, still some personal data is shared and might need to be shared. For us, the limited sharing of such data means that it is out of scope. However, future research could focus on data protection law and add sensors, adaptors and context rules based on this field of law as well.

9.2.2.2 Customs law

Customs law is important, as it determines the obligations of businesses to share information with customs, *inter alia*. According to the expert we interviewed from our project, if information is shared to meet such an obligation, it is lawful. This means that it can ‘overrule’ the other context relationships that might make information sharing unlawful.

According to the customs law expert from academia, customs law regulates exchange with customs authorities. There is no provision on whether data, such as the data in the ENS, must or cannot be shared between businesses and with third parties. Customs law is public law. It is only about sharing between businesses and customs, and the obligations and limits of this. Private law regulates the sharing of data between private parties (i.e., businesses).

The Union Customs Code (UCC) is “*the legal framework for customs rules and procedures in the EU customs territory*” which entered into force on 1 May 2016 (The European Commission, 2018). The academic customs law expert indicated that there is

still not a lot known about how the UCC should be interpreted considering the coverage of additional information sharing. The UCC is relatively new, there simply is not a lot of legislation available about it, and there are no books about it yet.

According to the customs law expert from customs, there is customs law and there are specific laws, such as for possession of weapons. Customs law is related to all these specific laws. When you follow the obligation specified in customs law, for example on what declarations to submit, then you also oblige to the specific law. He views customs law as the most important law to get information.

9.2.2.3 Intellectual property law

Part of IP law is the protection of trade secrets. IP law and especially trade secrets are relevant according to the IP law experts and the competition law expert. According to the IP law expert from academia, IP law is a very broad field that ranges from copyright to trademarks and know-how protection. It revolves on the one hand around exclusive rights to data and on the other hand to fair competition. IP law thus touches competition law as well.

9.2.2.4 Competition law

According to the IP law expert from academia and the transport law and commercial law expert, competition law is important. According to an expert from our project, competition law is concerned with prohibiting competing businesses to adjust their behaviour to the disadvantage of consumers. For example, if a business knows about the pricing of another business, they can adjust their own pricing to that. In general, it is not lawful for competitors to have the possibility to access data that could disturb the relationships in the market and make it possible to predict the future behaviour of competitors.

Competition law might lead to juridical issues when there are competitors in a system. The competition law expert confirms that competition law is relevant when competitors use the same system and he believes that competition law is relevant in the scenarios presented during the interview. According to the expert in competition law, the basis is that the sharing of commercially sensitive information between competitors either directly or indirectly is not lawful.

The IP law expert from industry states that competition law is difficult to deal with. This difficulty stems from that even the access to data already can play a role in the lawfulness of an information flow, regardless of whether the data was actually used by a business. The competition law expert confirmed this.

9.2.3 The context of the lawfulness of the information flow provided by the architecture

Only three of the context relationships from the exploration phase were also found in the testing and refining phase. These three were thoroughly refined as well. Furthermore, various new context relationships were found.

We will discuss each of the context relationships of the lawfulness of the information flow below. For each of them, we also present their context elements, represented by predicates. In addition, we already make a selection on which context elements will be sensed and adapted to and we discuss this choice. In chapter 10, the sensors and adaptors will be further worked out. We present the context elements in the same way as in section 9.1.2. Predicates for context elements that are the same in this section as the previous section have the same meaning, as do any other symbols.

9.2.3.1 Submit documents

The first context relationship is based on customs law. According to this context relationship, a flow of information is lawful when a business has an obligation to share data with customs and they share this data with them.

The information shared in a flow of information is a set of data elements. We can view documents as a set of data elements as well. For example, the ENS can be viewed as the set of data elements that it contains. According to the interviews, the obligation of a business to submit a set of data elements stems from their role in the shipment about which the data is. So, a business can be obligated to share the ENS with customs because they have the role of a carrier for a certain shipment.

The predicate *obligation* $\left(\begin{smallmatrix} submit, SetofDataElements, \\ GovernmentAgency, Shipment, Role \end{smallmatrix}\right)$ expresses that a business with role *Role* for shipment *Shipment* has the obligation to submit *SetofDataElements* that are about shipment *Shipment* to government agency *Government agency*. For example, *obligation* $\left(\begin{smallmatrix} submit, ens, \\ KZDU3401208, customs, carrier \end{smallmatrix}\right)$ expresses that the business with the role of the carrier for the shipment in the container with number *KZDU3401208* has the obligation to submit the ENS for this shipment to customs. This context element is a sensor element, as it is not possible to manipulate.

The predicate *hasRole* $(Party, Shipment, Role)$ expresses that party *Party* has the role of *Role* for shipment *Shipment*. In this example, we can use *hasRole* $(maersk, KZDU3401208, carrier)$ to express that Maersk is the carrier for the shipment in the container with number *KZDU3401208*. This is also a sensor, as it cannot be manipulated.

For businesses to meet their obligation, they should actually share the set of data elements with customs. Here, it is important that it is exactly the set of data elements that they are obligated to share, and not a subset of it, for instance. According to the interviewees, customs only considers an obligation to be met when it is the business that has the obligation that shares the data with them (or a party authorised by them, see next context relationship), and if it is the full set of data as once. This is why in the situation description it says “*there is a route containing these data elements*”. It is important to note that this is not a requirement that is specified in a law, but something that customs imposes.

To express this, we need to use several predicates. The first predicate *systemOf* $(System, Party)$, expresses that *System* is a system of *Party*. For example, *systemOf* $(23.198.24.203, maersk)$ expresses that the system with IP address

23.198.24.203 is the system of Maersk. This is a sensor element as well because we cannot manipulate it.

The second predicate we need concerns the route of the information in a flow, namely *routeinFlow(ListofSystems,SetofDataElements)*, where *ListofSystems* is an ordered list of systems and *SetofDataElements* is a set of data elements. This predicate expresses that in the flow of information *SetofDataElements* is shared from one system to another in the sequence of *ListofSystems*. For example, *routeinFlow([23.198.24.203,84.45.69.135,85.159.98.33],ens)* expresses that in the flow of information, the ENS is sent from the system with IP address 23.198.24.203, to the system with IP address 84.45.69.135,85 and then to the system with IP address 85.159.98.33. This is something that can be manipulated. This context element is thus an adaptor for the context-aware architecture.

The third predicate that we need is to express a constriction on the flow of information. This constriction is that the order of the systems in the route of the information should be in a certain way for the data elements. Namely, such that the system of the party that has the obligation to share data is before the system of the government organisation with which they are obligated to share. Otherwise, it would not be that business sharing the data elements with the government organisation. *index(System,ListofSystems)* is a function that has as output the index of *System* in list *ListofSystems*. The predicate $<$ has its usual meaning.

Using a combination of these three predicates, we can express that the system of the business should be in the flow of information of the set of data elements before the system of the government organisation. In other words, it expresses that the business shares the set of data elements with the government organisation.

What is interesting to mention here as well, is that according to the experts in customs law, businesses do not need to share the information directly with customs. This means that, for example, a carrier could share the ENS via a data pipeline or another system. However, if something goes wrong in the other system, the business is still held responsible for submitting the documents. Businesses can thus share via another system at their own risk; the business will be liable if something goes wrong. However, in some cases, the business could hold the party responsible for the system that they used liable as well, for example when they have neglected to properly secure the system.

Another requirement that needs to be met is that customs can directly read the data. Customs will not accept it if they receive data that businesses are obligated to share that is encrypted and they need to take steps to obtain a key and decrypt it. This means that we need to add one last context element. $\neg\text{encrypted}(\text{SetofDataElements})$ expresses that the set of data elements *SetofDataElements* is not encrypted. This last context element could be manipulated by the architecture. This is, for example, required for the context elements for willingness. However, in this case, this is not the most obvious choice for an adaptor for this context element, as it says that a manipulation should not be performed, instead of saying that a manipulation should be performed.

Restriction/ Situation	Context elements	Adaptor/ Sensor
Restriction: Lawful There is an obligation to share the data elements in the flow with a government agency by a party that has a certain role in the shipment. There is a route containing these data elements between this party and the government agency. Customs can directly access the data elements without having to decrypt.	$obligation \left(\begin{array}{c} submit, \\ SetofDataElements, \\ Shipment \\ GovernmentAgency, Role \end{array} \right)$	Sensor
	$hasRole(Party, Shipment, Role)$	Sensor
	$routeinFlow \left(\begin{array}{c} ListofSystems, \\ SetofDataElements \end{array} \right)$	Adaptor
	$systemOf(System_1, Party)$	Sensor
	$systemOf \left(\begin{array}{c} System_2, \\ GovernmentAgency \end{array} \right)$	Sensor
	$index(System_1, ListofSystems) < index(System_2, ListofSystems)$	Computation
	$\neg encrypted(SetofDataElements)$	Sensor

Table 26: The context relationship ‘Submit documents’

9.2.3.2 Authorised submitting documents

Based on the expert interviews, we found another way in which data sharing meets an obligation and thus is lawful. For the last context relationship, we discussed that a business can have an obligation to share data with customs. However, such a business can also authorise another party to submit these documents on their behalf.

According to the experts, a business (e.g., customs broker), can be authorised to act on behalf of another business, (e.g., a carrier), by representing them directly or indirectly. In direct representation, the business acting on behalf of another is liable. In indirect representation, the business acting on behalf of another is not liable. In our case, this is not directly important and this distinction is left out of the context relationship.

This context relationship another context element is added to the previous one to express that a business authorises another business to submit documents on their behalf.

The predicate $authorized \left(\begin{array}{c} Party2, Party1, submit, SetofDataElements, \\ Shipment, GovernmentAgency \end{array} \right)$ seems quite complex. However, the last four arguments are just referring to the first context element of the obligation that businesses have to share data with a government organisation. The predicate expresses that party *Party2* authorises party *Party1* to meet this obligation and submit the required document on their behalf.

One of the interviewees mentioned that multi-filing will be possible in the future. However, as this is currently not possible, we left it out here. This can be added as a context element later.

Restriction/ Situation	Context elements	Adaptor/ Sensor
<p>Restriction: Lawful</p> <p>There is an obligation to share the data elements in the flow with a government agency by a certain party.</p> <p>This party authorises another party to submit the documents.</p> <p>There is a route of the information between the authorised party and the government agency.</p> <p>Customs can directly access data elements without decrypting.</p>	$\text{obligation} \left(\begin{array}{c} \text{submit,} \\ \text{SetofDataElements,} \\ \text{Shipment} \\ \text{GovernmentAgency, Role} \end{array} \right)$	Sensor
	$\text{hasRole}(\text{Party}_2, \text{Shipment}, \text{Role})$	Sensor
	$\text{routeinFlow} \left(\begin{array}{c} \text{ListofSystems,} \\ \text{SetofDataElements} \end{array} \right)$	Adaptor
	$\text{authorized} \left(\begin{array}{c} \text{Party}_2, \text{Party}_1, \text{submit,} \\ \text{SetofDataElements,} \\ \text{Shipment,} \\ \text{GovernmentAgency} \end{array} \right)$	Sensor
	$\text{systemOf}(\text{System}_1, \text{Party}_1)$	Sensor
	$\text{systemOf} \left(\begin{array}{c} \text{System}_2, \\ \text{GovernmentAgency} \end{array} \right)$	Sensor
	$\text{index}(\text{System}_1, \text{ListofSystems}) < \text{index}(\text{System}_2, \text{ListofSystems})$	Computation
	$\neg \text{encrypted}(\text{SetofDataElements})$	Sensor

Table 27: The context relationship ‘Authorised submitting documents’

9.2.3.3 No Sharing against agreement

The next context relationship is from the field of IP law. According to the juridical expert we interviewed, a flow of information is unlawful if a party is supplying data elements to the information flow and they have signed a contract or agreement that they cannot do so.

An important discussion is whether there is an owner of data. According to the experts in IP law, common misunderstanding is that data can be owned. Who has control over data depends on the agreements between parties. This means that if a party has received data from another party and they have not signed a confidentiality agreement, then they are allowed to share the data (as far as this area in IP law is concerned).

$\neg \text{public}(\text{SetofDataElements})$ expresses that *SetofDataElements* is not already public. Of course, this could be manipulated by a component that makes the set of data elements public. However, it is highly likely that this will be unlawful as well in the situation as there are agreements that some parties cannot share the data. This context element is therefore a sensor element.

A party is supplying a set of data elements as they have a system that is part of the route of the set of data elements in the flow of information and if this system is not the last system in the route. This is what is expressed by the second, third and fourth predicates.

$\text{agreement}(\text{Party}_2, \text{Party}_1, \text{notshare}, \text{SetofDataElements})$ expresses that *Party1* has an agreement with *Party2* that they are not allowed to share

SetofDataElements . This is a sensor element, as it cannot be manipulated automatically.

Restriction/ Situation	Context elements	Adaptor/ Sensor
Restriction: Unlawful	$\neg public(SetofDataElements)$	Sensor
The flow of information contains a set of data elements that are not public. A party is supplying the set of data elements to the flow. This party has a contract or agreement saying that they cannot share the data.	$systemOf(System, Party_1)$	Sensor
	$routeinFlow(ListofSystems, SetofDataElements)$	Adaptor
	$index(System, ListofSystems) < length(ListofSystems)$	Computation
	$agreement(Party_2, Party_1, notshare, SetofDataElements)$	Sensor

Table 28: The context relationship ‘No Sharing against agreement’

9.2.3.4 Protect trade secrets

The protection of trade secrets was mentioned by several juridical experts as being relevant to determining the lawfulness of information flows. According to one of the experts in IP law, the sharing of trade secrets is lawful when all parties that can access the trade secret are in agreements binding them to confidentiality. According to the juridical expert involved in the project, the trade secret holder should give others the right to access the trade secret.

The literal $tradesecret(SetofDataElements, Business)$ expresses that *SetofDataElements* is a trade secret of *Business*. This means that *Business* is the trade secret holder that controls access to the trade secret that is confidential otherwise. This is a sensor element, it is undesirable to reveal businesses’ trade secrets and therefore it is undesirable to manipulate it.

$agreements(Business, SetofParties, notshare, SetofDataElements)$ is similar to the context element of an agreement between two parties in the previous context relationships. The difference is that this predicate expresses that *Business* has an agreement with all parties in *SetofParties* that they are not allowed to share *SetofDataElements*. This is a sensor element.

$elementsinSet(ListofSystems, SetofParties)$ expresses that each element in *ListofSystems* is an element *SetofParties*. It is thus a form of the subset relationship, however in this case between a list and a set.

Restriction/ Situation	Context elements	Adaptor/ Sensor
Restriction: Lawful The flow of information contains a set of data elements that are a trade secret. All parties that get access or that potentially get access to this information are bound by confidentiality obligations.	$tradesecret \left(\begin{matrix} SetofDataElements, \\ Business \end{matrix} \right)$	Sensor
	$agreements \left(\begin{matrix} Business, \\ SetofParties, notshare, \\ SetofDataElements \end{matrix} \right)$	Sensor
	$routeinFlow \left(\begin{matrix} ListofSystems, \\ SetofDataElements \end{matrix} \right)$	Adaptor
	$elementsinSet \left(\begin{matrix} ListofSystems, \\ SetofParties \end{matrix} \right)$	Computation

Table 29: The context relationship ‘Protect trade secrets’

9.2.3.5 Not share competitively sensitive data with competitor

According to competition law, it is not lawful to share competitively sensitive data between competitors if this data is not public. According to the experts, even providing a competitor with the possibility to access the data is already unlawful.

$comerciallySensitive(SetofDataElements, Business, Competitor)$ expresses that $SetofDataElements$ is commercially sensitive for $Business$ to $Competitor$. This is a sensor element.

For a business to provide the possibility of access to data to a competitor there needs to be a route of this data between their system and a system the competitor can access. The literal $possibleAccess(System, SetofDataElements, Competitor)$ expresses that $Competitor$ has the possibility to access $SetofDataElements$ in $System$.

9.2.3.6 Shield data from competitors

According to the competition law expert we interviewed, one way to prohibit the competitor of a business to have possible access to the data is by encrypting the data and not providing them with a key. According to the competition law expert, the party that decides who gets a key could be an independent third party. We will discuss this in more detail in the section on the adaptors, as it is a requirement on the related adaptor.

Restriction/ Situation	Context elements	Adaptor/ Sensor
<p>Restriction: Unlawful</p> <p>A set of data elements is competitively sensitive for a business from their competitor. This set of data elements is not public. The competitor has the possibility to access these data elements.</p>	$commerciallySensitive \left(\begin{matrix} SetofDataElements, \\ Business, Competitor \end{matrix} \right)$	Sensor
	$\neg public(SetofDataElements)$	Sensor
	$systemOf(System_1, Business)$	Sensor
	$routeinFlow \left(\begin{matrix} ListofSystems, \\ SetofDataElements \end{matrix} \right)$	Adaptor
	$possibleAccess \left(\begin{matrix} System_2, \\ SetofDataElements, \\ Competitor \end{matrix} \right)$	Sensor
	$index(System_1, ListofSystems) < index(System_2, ListofSystems)$	Compu- tation

Table 30: The context relationship ‘Not share competitively sensitive data with competitor’

Restriction/ Situation	Context elements	Adaptor/ Sensor
<p>Restriction: Lawful</p> <p>A set of data elements is competitively sensitive for a business from their competitor. The data elements are encrypted in the flow of information. The competitor cannot obtain a key to decrypt.</p>	$commerciallySensitive \left(\begin{matrix} SetofDataElements, \\ Business, \\ Competitor \end{matrix} \right)$	Sensor
	$encrypted(SetofDataElements, Key)$	Adaptor
	$\neg hasAccess(\{Key\}, Competitor)$	Adaptor

Table 31: The context relationship ‘Shield data from competitors’

9.2.3.7 Not make public competitively sensitive data

If making a set of data elements public disturbs the relevant market, then this is unlawful according to the juridical experts.

disturbMarket(SetofDataElements) expresses that *SetofDataElements* disturbs the relevant market when it is made public. This is a sensor element, as it cannot be manipulated.

makesPublic(System, SetofDataElements) expresses that when *SetofDataElements* is shared with *System*, it will make it public. This is also a sensor element, as it cannot be manipulated.

Restriction/ Situation	Context elements	Adaptor/ Sensor
Restriction: Unlawful	<i>disturbMarket(SetofDataElements)</i>	Sensor
A set of data elements disturbs the market when made public. The data elements are made public in the flow of information.	<i>routeinFlow</i> $\left(\begin{array}{c} ListofSystems, \\ SetofDataElements \end{array} \right)$	Adaptor
	<i>elementOf(System, ListofSystems)</i>	Computation
	<i>makesPublic</i> $\left(\begin{array}{c} System, \\ SetofDataElements \end{array} \right)$	Sensor

Table 32: The context relationship ‘Not make public competitively sensitive data’

9.2.3.8 Sharing public data

The last context relationship is very simple, albeit important. According to the experts, sets of data elements that are public can be shared freely.

inFlow(SetofDataElements) expresses that *SetofDataElements* is shared in the flow of information under consideration. This can be manipulated by the architecture and therefore is an adaptor element.

Restriction/ Situation	Context elements	Adaptor/ Sensor
Restriction: Lawful	<i>public(SetofDataElements)</i>	Sensor
A set of data elements is public. This set of data elements is shared in the flow of information.	<i>routeinFlow</i> $\left(\begin{array}{c} ListofSystems, \\ SetofDataElements \end{array} \right)$	Adaptor

Table 33: The context relationship ‘Sharing public data’

9.3 Validity and reliability

According to Yin (1994), the quality of a case study can be determined according to several criteria. Other authors pose similar measurements for research quality as well. In

the previous subsections, we have already described and provided arguments for several decisions we made in the way in which we collected and analysed the data. In this section, we will summarise those and describe how this ensures the overall quality of the research into the context of the context-aware architecture.

Construct validity refers to “*establishing correct operational measures for the concepts being studied*” (Yin, 1994, p. 33). Yin (1994) provides three strategies for improving construct validity. The first is using multiple sources of evidence. For each of the foci, we used multiple sources of evidence. For the focus of willingness, we performed an interview with a researcher and staff at a business involved in a project and we studied documents on another project. For the focus of the lawfulness of information flows, we interviewed juridical experts with different areas of the law as their expertise. Furthermore, we interviewed experts from academia as well as experts from industry and customs.

The second strategy recommended by Yin (1994) is to establish a chain of evidence. We followed this strategy in this study as well. Namely, for each of the context relationships we derived from the data, we kept track exactly of its evidence and referred back to the transcript or document providing this evidence. In addition, Yin (1994) suggests that the draft case study report is reviewed by key informants. We did not do this, however, we presented our results at several workshops involving the other researchers in our project, and involving technical and juridical experts at a large sea carrier.

Internal validity is about establishing causal relationships (Yin, 1994). The purpose of step 1 in the design of a context-aware system is to determine what belongs to the relevant context to take into account for the design of the context-aware architecture. According to our definition of context (see chapter 5), something is part of the relevant context if it impacts a focus of the architecture. Such an impact is a relationship between the context and a focus. The method makes explicit what it means to impact a focus by providing a criterion for this (see p. 91). Furthermore, it provides a procedure for determining the foci of an architecture (step 1.1 of the method, section 6.1.1) and for testing whether the criterion is met (step 1.2 of the method, section 6.1.2). In this way, the new method thus provides for a systematic procedure to establish whether there is an impact and it thus supports ensuring the internal validity of the research.

The external validity concerns the extent to which results can be generalised (Yin, 1994). We did so by replacing elements in a situation description that refer to something specific, such as ‘freight forwarder A’, by the group that the element belongs to, such as ‘businesses supplying data in the information flow’ (see sections 9.1.1.4 and 9.2.1.2.4). We chose the groups based on insight into the context relationships and our understanding of why the situation impacts the focus in the context relationship.

According to Yin (1994), to show the reliability of the research, it needs to be demonstrated that the process of the research can be repeated with the same results. To ensure reliability, all interviews were conducted based on an interview protocol. In addition, we took the weaknesses of the different data collection methods into account and took measures to deal with them (see section 9.1.1.3 and section 8.3.2.2.1). Furthermore, for the case study, we specified a case study protocol on beforehand and we

kept a case study database. The analysis of the data was performed according to the new method and thus followed clearly documented steps as well.

10 Sensors, adaptors and context rules for the context-aware architecture

In chapter 9, we presented the results of our investigation into the context of the foci of the context-aware architecture. In section 12.1, we provide an overview of the overall architecture we present in this dissertation. The goal of the architecture is to provide for information flows in which businesses that supply information are willing to participate and that are lawful. The system of the architecture thus falls in the category of systems that maximise the efficiency of delivered services (Shishkov, Larsen, Warnier, & Janssen, 2018)

For these systems, context information is highly important to determine what services to deliver and how to adapt. In this chapter, we describe the part of the architecture that we can derive from insight into context, namely its sensors, adaptors and context rules. Adaptors and sensors are derived from insight into context, expressed as context relationships and context elements, in step 2 of the new method (see section 6.2).

We already presented part of the results of step 2, namely what context elements are sensor elements and what context elements are adaptor elements in chapter 9. The reason for this is that the decision for what context elements to sense and to manipulate is decided, in part, based on the context relationships. It seems thus more natural to discuss this at the same place where the context relationships are discussed. In this section, we first discuss what components will manipulate the different adaptor elements (section 10.1), or sense the different sensor elements (section 10.2). In appendix B, we provide an overview of the context elements, the sensors and adaptors that sense or manipulate them, and the context relationships of which they are a part.

Some of the components have been proposed during the interviews with the juridical experts already to determine whether they are suitable to sense or manipulate the context elements. We also discussed some of their requirements with these experts. If we derived a choice or requirement from the interviews, we refer to that. The way in we conducted the expert interviews is explained in section 9.2.1.

The context rules are derived from the insight into context as well. This is done in step 3 of the new method we propose (see section 6.3). We present the context rules we derived for the foci in section 10.3.

Parts of this chapter have been published in van Engelenburg, Janssen and Klievink (n.d.).

10.1 Adaptors

In this section, we provide an overview of the adaptors that the context-aware architecture should include for manipulating different context elements. There are two types of adaptors needed in the architecture. In the first group, there is only one type of adaptor, namely information routers. These information routers ensure that data is shared in a flow of information in which businesses are willing to participate and that is lawful, according to the context rules. The other group of adaptors affect the data itself in some way, by encrypting it, making it thin, or making it only viewable.

10.1.1 Information routers

Definition 10 (p. 32) defines a context-aware architecture as follows: “*A context-aware architecture is an architecture of a context-aware system that adapts by changing its organisation to only use the required components to perform the required operations to meet the system goal.*” We discussed that in our case we need a context-aware architecture, as different situations will require different flows of information. This means that we need an adaptor that changes the organisation of the architecture itself in order to provide for the appropriate flow of information in a variety of situations. As such, an adaptor will ensure that information is shared according to the sequence of systems in a certain information flow, or in other words, according to a certain route, we refer to it as an information router.

Any context-aware system would need an information router to adapt the flow of information. This means that information routers are one of the basic components of a context-aware architecture that do not directly depend on the context. However, the flow of information according to which information is shared also can be used to manipulate some of the adaptor elements. This means that the purpose of using it in the architecture presented here is influenced by the context. Therefore, we discuss it in this section and not in chapter 11 where we present the other basic components.

What information flow is appropriate for sharing information is determined by another basic component, namely an information flow planner. The information routers ensure that the information is shared according to the flow that the information flow planner decides upon. The information flow planner obtains context information from the context information repository. Furthermore, it stores proposed information flows in the context information repository, from where it can be obtained by the information routers.

The proposed design of the context-aware architecture contains three types of systems as components that could be part of an information flow. The first are adaptors (e.g., an encryption component), the second are information sharing systems (e.g., business systems, data pipelines). Each of these systems is connected to the architecture using an information router. Information can be shared between systems using the information routers. All users agree on beforehand that they will only share information using the information routers.

There are three ways in an information sharing process using information routers can be initiated: 1) the information router receives new information from another information router in the architecture (depicted in figure 10), 2) the owner of the system that is connected to the architecture using the information router adds new information to share (push) (depicted in figure 11), and 3) the information router receives a new request to share information stored in its system (pull).

In the case of possibility 1 (figure 10), the information router sends the new information to its system so it can be accessed and used by its owner. The information router then requests the information flow planner for information flows according to which the information should be shared. The information flow planner decides based on the context rules and context information what the appropriate flows of information are for the context elements. It then shares these flows of information with the information

router. The information router checks what information routers are directly next to them in the sequence of these flows and sends the information to these information routers.

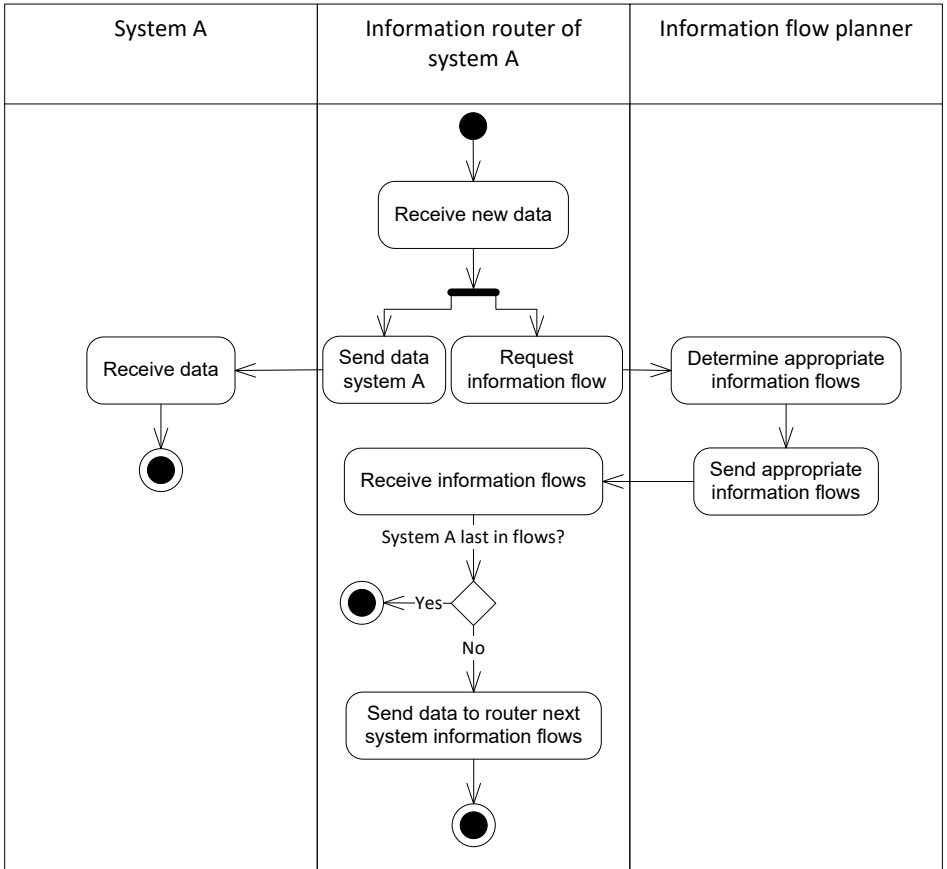


Figure 10: UML activity diagram for data sharing using an information router in the context-aware architecture (receive and send further)

There are several reasons for the owner of a system to push new data to be shared via its information router. The first possibility is that they have generated new data, for example, a packing list, and they want to share it with others. The second possibility is that they have received information previously and that they have adapted or processed it somehow and now want to push it. This could be the case, for example, if the system is an adaptor that encrypts sensitive data elements in data sets received and then shares the partially encrypted data set. In this case, the information sharing process follows the same steps as for the previous possibility, of course with the exception that the data router adds the information to its system. This process is depicted in figure 11.

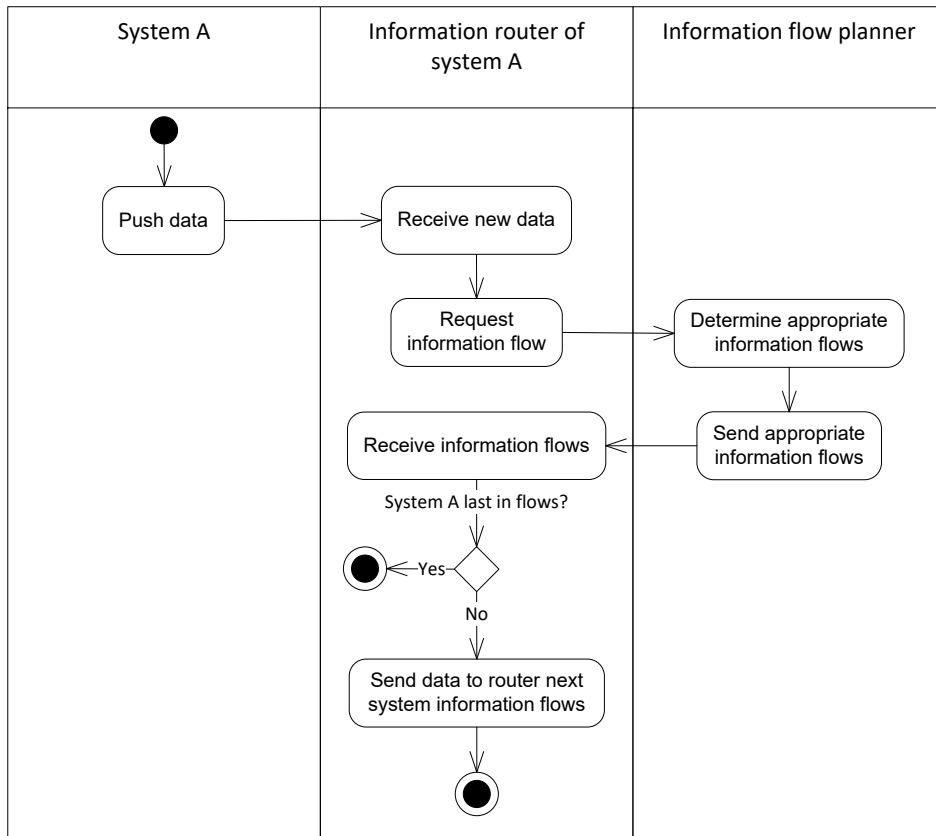


Figure 11: UML activity diagram for data sharing using an information router in the context-aware architecture (Push data)

The last possibility is that another party wants to pull information from the system of the information router. In that case, the information sharing process is initiated by a request for information received by the information router. Other than that, the process is the same.

A request from an information router to an information flow planner contains the following data elements:

1. Identifier of the information router sending the request
2. Identifiers of the data elements shared
3. Identifier of the information router of a party to share with

The first data element is used to identify the information router that sends the request to the information flow planner. The second data element is identifiers of the data elements that are shared. The information flow planner needs these identifiers in order to obtain the correct context information on them. The last data element identifies the information router with which the data should be shared.

A proposed information flow that the information router receives from the planner consists of the following data elements:

1. Identifiers of data elements
2. A sequence of identifiers of systems

The sequence of systems is the route that the data elements identified should follow from one system to the other, I the information flow.

The information routers only share information according to the information flows proposed by the information flow planner. All parties using the architecture should agree on beforehand that they only share information using the information flow planner. In this way, the information routers thus enforce access control. This means that the information routers manipulate all adaptor elements in the information flow that have to do with restricting or providing access to sets of data elements. These are the following adaptor elements:

- *hasAccess(SetofDataElements, Party)*
- *hasAccessSystem(SetofDataElements, System)*
- *canAggregate(Party, SetofDataElements)*
- *routeinFlow(ListofSystems, SetofDataElements)*
- *filter(SetofDataElements, FilteredSetofDataElements, Business)*

Whether the context-aware architecture supports lawful information sharing in which businesses are willing to participate thus depends on the information flow. Each context relationship impacts either the willingness of businesses supplying information to participate in the information flow provided by the architecture, or the lawfulness of the information flow provided by the architecture. This means that each of the context relationships contains one of the adaptor elements mentioned above.

Information routers can influence whether a party has access to information by sending or not sending (part of) it to the information router of a system in which they can access the information. The same is true for providing access to the information to a system. Whether a party can aggregate information depends on the combination of data sets that are shared with it in the same way. Furthermore, the information routers of the different systems together influence what route the information follows. Of course, the information routers will do all of this based on the decisions by the information flow planner. The information routers can also filter or anonymise information by only including certain data elements in the set that they share.

The information routers are not only adaptors and basic components; they are sensors as well. According to many of the context relationships, certain parties or their systems cannot have access to certain sets of data elements. As we will discuss in further detail when we explain the sensors, usually data elements by themselves are not sensitive, but combinations of data elements are. The same is the case for, for instance, competitive sensitivity of data according to competition law.

To ensure that there is no access to certain combinations of data elements, not only the data elements that are shared in the current flow of information needs to be assessed, but also the data elements received previously by parties. As the information routers send information to the next system in the flow, they can provide information on who has access to what data. This means that each time they send information to the next system in an information flow, they should add an access history statement to the

repository with context information. The access history statement added by information routers should contain the following data elements:

- Identifier of the information router sharing the data
- Identifiers of the data elements shared
- Identifier of the information router the data elements were shared with

The information flow planner, in this way, can make decisions based on the full access history of parties or their systems. The agreement of businesses to share their data only via the information routers is highly important for enabling this.

10.1.2 Other adaptors

Each of the other adaptors has an information router and, in general, functions like any other system that is connected to the architecture. That means that if new context elements are identified that cannot be manipulated by the current adaptors, new adaptors can easily be added to the architecture by providing them with an information router and adding new rules to the information flow planner.

The adaptors can only perform their functions and manipulate context elements when they receive information via their information routers. This means that if they need to manipulate a context element, and for example encrypt data elements, the information flow planner needs to include them in the planned information flow.

10.1.2.1 Encryption component

The first adaptor is an encryption component. This adaptor manipulates context element *encrypted(SetofDataElements,Key)*. This component receives information via its information router. The encryption component encrypts the data elements. It stores the key that can be used to decrypt the data. The encryption component then shares the following set of data elements via its information router:

- The encrypted data elements
- Identifiers of the original data elements
- Its own identifier

The identifiers of the original data elements are added so that it is possible for the receiver of the encrypted data elements to look up their metadata (if available via e.g., a data pipeline) and determine what kind of data they contain and whether it is interesting for them to decrypt it.

The set containing the encrypted data elements can just be shared like any other information in the architecture. This means that it is shared according to the flow planned by the information flow planner. The key to decrypt the data can be shared in the same manner as well. However, the key will only be shared when a request is made to share it. When a business cannot have access to the key according to the context rules, then the information flow planner will not provide an information flow including this business. The information router then does not have an information flow according to which to share the key and therefore the key will not be shared with them.

From the point of view of the willingness to share information, the encryption should be of a quality that is high enough to ensure businesses that it cannot simply be

decrypted. Furthermore, they should trust the party that provides the encryption component and that they store the key securely enough. The exact algorithms to encrypt the data elements, the governance of the encryption system and the security of this system should be subject to further research. When these requirements vary, there is no obvious reason to have multiple encryption components, or for businesses to have and use their own encryption system.

From a juridical point of view, we can also say something about the requirements on the encryption component and the quality of encryption, based on the interviews. First, several of the experts state that they do not view the sharing of encrypted data elements as actual sharing when no key is provided to decrypt the data.

It is possible according to the experts to prohibit competition law issues by controlling access to competitively sensitive information by encrypting it and not providing competitors with a key (see section 9.2.3.6). The competition authority really has to show that there has been an exchange and that the data is commercially sensitive. According to the experts, the latter is not the case if you do not have the key. This way of controlling access to data can be useful, for example, if data elements are shared via a blockchain architecture or data pipeline in which competitors are users.

From the point of view of IP law, if a trade secret is effectively encrypted, then it is an effective way to keep a trade secret a secret. This is important, because if a trade secret is shared with a party that has not signed a confidentiality agreement, then it is not protected. In this way, the data can be encrypted and still be shared via systems used by parties that have not signed such an agreement from a juridical point of view (see section 9.2.3.4).

The quality of encryption is important according to the experts. The obligation to use good enough encryption is similar to the obligation to take appropriate security measures for the system where data is stored. When the encryption of data is of very high quality and it is cracked anyway, then the party that encrypted and shared the data is not considered at fault for any issues arising from that. However, this party is considered at fault if anybody could crack the encryption.

What quality of encryption is considered ‘good enough’ from a legal point of view depends on the type of legislation and the origin of legislation. In some cases, this will be what the market deems fit. In other cases, it could be required to adapt the encryption every time new research is published with new options that are feasible.

10.1.2.2 Thin maker

Another adaptor included in the context-aware architecture is a ‘thin maker’. This component manipulates the context element *thin(SetofDataElements)*. It receives information via its information router. It then provides a link to the information in the form of an identifier of an information router. Next, it shares a set containing the following data elements via its information router:

- Link to the identifier of the information router of the system where the data is stored
- Identifiers of the data elements

If the thin maker receives a set of data that is already thin, then it will not make any changes and just share it further. The identifiers of the data elements can be used by a party to obtain information on the data and determine whether it is useful for them to try to pull it.

It is likely that the businesses that share the data want to store it in their own systems to keep control over it. The thin maker should thus add the identifier of the information router of that system to the set of data elements that it shares. This means that for this component to work, there should be a data element in the information that it receives that refers to who this party is.

10.1.2.3 Data viewer

The last adaptor, namely the data viewer, is a bit different from the other adaptors. It does not receive data and then sends it further, like the encryption component or the thin maker. Instead, it receives the data and provides a service using it. In that sense, it works more similar to the other systems that are incorporated in the architecture that provide a service, such as single windows or data pipelines.

The service provided by the data viewer is that it lets other parties view the data without them being able to store it. It thus manipulates the context element *view(Party, SetofDataElements)*. To do so, they need to receive the data elements, as well as the identifier of the party that should be able to view the data elements in their system.

From the juridical point of view, experts were sceptical about letting businesses only view data to prohibit them from aggregating it as a way to deal with juridical issues. They suggest that the businesses would just be able to view the data and enter it into an Excel sheet, for example, and aggregate it anyway. In addition, it would not help to deal with issues in the area of competition law. It would be similar to telling the competitively sensitive information in person, which can also break competition law.

Because of this, according to the context rules, viewing the data will only happen when businesses do not want customs to aggregate the data. The investigations on the context of the willingness of businesses indicated that businesses could view this as a solution if they do not trust customs with their aggregated data (see section 9.1.2.4). According to the customs law expert at customs, it would not be an issue for customs if they could only view data instead of storing it. According to him, they can just view the data and then make a note if they see something interesting that they want to investigate.

10.2 Sensors

The sensors provide context information, which is stored in a context information repository. It is important to note that ‘sensor’ here should be interpreted broader than usual and as anything that can provide context information. The information flow planner uses this context information and the context rules to decide what the appropriate flow of information is for the information shared. This information flow will then be provided by the architecture using the adaptors.

We have identified three sources of context information that can act as sensors for sensing context elements, namely businesses involved in information sharing, independent third parties and customs. These sensors are different from what often is considered sensors traditionally, such as GPSs or thermometers. For most, if not all, of the context elements we found, it is impossible to determine their value automatically. A good example of this is the competitive sensitivity of data elements, which depends on the market, the position in the market of different businesses and the goods that the data is about, *inter alia*. Competitive sensitivity of data is thus not an attribute of the data itself but arises from multiple factors. It would therefore currently be impossible to let a system just view the data and determine whether it is competitively sensitive based on that.

A solution could be to add things like the market situation and on what goods businesses are competitors in the market as well as context elements. However, according to the experts we interviewed, this is subject to change and needs to be updated regularly. Currently, human insight is still necessary to know when and how this changes. Therefore, it is easier to let a third party just add what data elements are competitively sensitive and update this regularly and reduce the complexity of adding information on the relevant market and such. The different sensors are discussed in the rest of this subsection.

10.2.1 Businesses involved in information sharing

The first source of context information is the businesses involved in the information sharing whose systems are part of the architecture. The first type of context information that businesses provide is on the agreements that they are in with other businesses and the relationships that they have with others. The second type of context information that they provide is about the properties of the data they share. The last type of context information is on what data are sensitive to them and their rules for who can have access to their data.

10.2.1.1 Agreements between businesses

Businesses that are involved in the same supply chain and that share information with each other have different kinds of agreements with each other. First, there are the agreements concerning the goods in a supply chain and their transport (see section 7.2.2). From these agreements and documents, we can derive what businesses are connected to the shipment of goods and what their role is. For example, the contract of carriage could be used to identify who is the shipper and who is the carrier for a shipment.

The second type of agreement is about the data that is shared between parties. First, parties could authorise others to act on their behalf. In addition, they could make agreements on confidentiality of data and on what other parties can and cannot do with the data shared.

Information on agreements about shipments that are added to the context information repository contains the following data elements:

1. Identifier of the shipment
2. Identifier of the first party (public key)
3. Role of the first party according to the agreement (e.g., shipper)

4. Identifier of the second party (public key)
5. Role of the second party according to the agreement (e.g., carrier)
6. Hash of the agreement
7. Data elements 1-6 signed with the private key of the first party
8. Data elements 1-6 signed with the private key of the second party

From these data elements, the value for the following context elements can be derived: *connection(Party, Shipment)* and *hasRole(Party, Shipment, Role)*.

The shipment can be identified using common identifiers, such as a global shipment identification number (GSIN) (GS1, 2013). Each party involved in the context-aware architecture can be assigned a public key by a certification authority that can be used to identify them (see section 10.2.2.1). Their role can be specified using a standard.

The hash of the agreement needs to be added for security reasons. Whether a party gets access to information depends on the role in the shipment they have according to the agreements that they have with others. This means that it is important to ensure that the information on this is reliable. One way to do so is by connecting the data to the actual agreement upon which it is based. This makes it possible, in case of suspicion or dispute, to check whether the business actually had this role in the shipment according to an actual agreement.

A hash of data is a string of symbols which is assigned to it. In general, unique data has a unique hash. This means that a hash of an agreement can be used to identify that particular agreement. Agreements can contain sensitive information that should be kept confidential. However, in contrast with encryption, calculating a hash of data means losing part of the information. This means that the hash cannot just be translated back into the data. If an agreement is standardised, someone wanting to know what is in it could calculate the hash of different possibilities and check whether it conforms to the hash of the agreement added with the context information. Such an attack could be countered by simply adding a random number (salt) behind the agreement and hash it together with the agreement.

Furthermore, both businesses should sign this set of data elements using their private key to confirm that they believe the data is correct. Their public key can be used to check that they have signed the data. If for a shipment double roles are added, such as a second shipper, and one of those is not in an agreement with any other party connected to the shipment, then this could give rise for concern. This is something that could be checked for automatically.

The second type of agreement is about the data elements. Usually, businesses will not make agreements on the level of sets of specific data elements. For example, they will not make an agreement to keep the price of specific goods in a specific container confidential. Instead, they agree to keep certain types of information confidential (such as prices). However, the information flow planner needs to make decisions on how to share specific sets of data elements. It thus needs input on this level. If the confidentiality agreements are standardised, then it could be possible for businesses to derive automatically from these confidentiality agreements what the agreement is on the level of specific data elements. For example, if according to an agreement, prices could be kept

confidential and according to the metadata of a data element, it is a price, it could be derived that the data element is subject to the agreement.

The set of data elements added to the context information repository derived from agreements on data sharing are the following for agreements restricting the sharing and use of data:

1. Identifier for the party that discloses the data (public key)
2. Identifier for the party that receives the data (public key)
3. Action that the receiving party cannot perform on the data (e.g., sharing)
4. Set of identifiers of data elements for which the receiving party cannot perform the action
5. Hash of the confidentiality agreement
6. Data elements 1-5 signed with the private key of the disclosing party
7. Data elements 1-5 signed with the private key of the receiving party

From these data elements, the value for the following context elements can be derived: *agreement(Party₁, Party₂, Action, SetofDataElements)* and *agreements(Business, SetofParties, Action, SetofDataElements)*.

The same public keys can be used as identifiers for parties as previously. The actions can be standardised as well, just like the roles. To improve the reliability of the context information, a hash of the confidentiality agreement is added and the data elements are signed by the parties in the agreement.

The identifiers for the data elements could be their hashes, as they can be used to identify uniquely the data elements, without compromising their confidentiality. Furthermore, the first disclosing party that adds the context information shows in this way that they actually have the data. Otherwise, they would not have been able to calculate its hash. This prohibits other parties adding context information that they control the data that you could have when just assigning identifiers to data elements sequentially, for example.

Who controls the data relates to the discussion of who owns data. Ownership over data was extensively discussed during the interviews. According to the expert in IP law, it is a common misunderstanding data can be owned in the way physical goods can be owned. Parties have access to data, or they do not have access to data. According to the expert, when data is sold, this is on the basis of the other party not having access. A party having data and providing another party with access does not mean that the data is owned.

If data is supplied, then a party can get into a contractual relationship with the party that they are supplying the data to that they will keep it confidential. This is often part of the participation contract of a system as well. This ensures that everybody has to come to them to get access to the data. However, that does not imply that the data is owned by the party. If someone else figures out the same data, they can use it. Having access to data by itself can provide a party with power, but not with rights just by having access to it. Having a patent could provide a party with rights, but this is out of the scope of the research. This also means that, concerning IP law, a party is free to share data if they are not bound to a contract prohibiting them from doing so.

The last type of agreement is about the data as well. However, instead of restricting what can be done with the data, in this type of agreement one party authorises another party to act on their behalf and, for example, submit documents on their behalf to customs. The set of data elements that should be added to the context information repository is the following:

1. Identifier for the party that authorises (public key)
2. Identifier for the party that is authorised (public key)
3. Action that the authorised party is authorised to do (e.g., submit data)
4. Set of identifiers of data elements for which the receiving party is authorised to perform the action
5. Hash of the authorisation agreement
6. Data elements 1-5 signed with the private key of the party that authorises
7. Data elements 1-5 signed with the private key of the party that is authorised

This information can be used to provide context information on the following context element: *authorized* $\left(\begin{matrix} Party_1, Party_2, Action, SetofDataElements, \\ Shipment, GovernmentAgency \end{matrix} \right)$.

10.2.1.2 Properties of data elements

Several context elements are properties of sets of data elements. Some of them can easily be established by the party that created them. The party that creates the data elements will thus add those to the context information repository. These are the following:

- *encrypted(SetofDataElements)*
- *thick(SetofDataElements)*
- *subject(SetofDataElements, Shipment)*

The data elements in a set of data elements can be identified, again, by their hashes and the shipment using the identifier used before as well.

In addition, there are properties of sets of data elements that are more complex to establish, for example, whether making them public would disturb the relevant market. Context information on these context elements can be added to the context information repository by a third party that can oversee the market as a whole (see section 10.2.2).

10.2.1.3 Data sensitivity and access control

The last type of context information provided by businesses is on the data elements that they consider sensitive and on how they want to control access to their data. Sets of data elements can be sensitive to businesses for a variety of reasons such that they do not want some other parties to have access. On the other hand, they want certain data to be accessible only to parties that they view as entitled to access.

The following context elements are about whom businesses would and would not like to have access to their data:

- *sensitiveTo(SetofDataElements, Business, Party)*
- *sensitiveToAggregate(SetofDataElements, Business, Party)*
- *entitled(Action, SetofDataElements, Business, Party)*

Businesses might not want others to have access to data for a variety of reasons. In addition, they might believe that others are entitled to access for a variety of reasons as well. What data elements others should be able to access according to a business, might be in itself context-dependent, it might change over time and depend on individual characteristics of the business.

It would be very laborious, if not impossible, for businesses to specify for every set of data elements to whom they are sensitive according to them explicitly. Furthermore, they might not know that new data elements exist before they arrive at their information routers, meaning that they are not even able to specify this before that time.

A promising solution for providing businesses with control over their data is to let them specify business rules according to which they want their data to be shared (van Engelenburg et al., 2015; van Engelenburg, Janssen, & Klievink, 2017a). In these business rules, either they could specify conditions under which they believe certain types of data elements are sensitive or the conditions under which they believe others are entitled to certain types of data. The information flow planner will use these rules to derive for specific sets of data elements whether they are sensitive according to the business to a specific party, or whether a specific other party is entitled to access according to them. This can then be used for further reasoning with the context rules and deriving an appropriate flow of information accordingly. It is important to note that depending on the business rules that the businesses add, additional context information might be required to determine whether the conditions in the rules are met.

If these business rules are going to be used together with the context rules, they should have the same format based on the logic-programming paradigm (see section 5.2). The only difference is that the head of these rules does not specify an action that needs to be performed. Instead, the head of the rule should be one of the schematic predicates above. Reasoning with these rules then can be done as usual to try to derive whether any of their ground instances is true (see (Lifschitz, 1996)).

An example of a business rule that could be specified by a business identified by public key “18XsP”¹ are the following:

```
sensitiveTo([GoodsDescription, Price], 18XsP, Party) ←  
competitor(18XsP, Party), subject(GoodsDescription, Shipment),  
subject(Price, Shipment), dataType(GoodsDescription, goodsDescription),  
dataType(Price, goodsUnitPrice).
```

This business rule describes that this business thinks that the combination of a data element describing the goods in a shipment and its price is sensitive from their competitor.

The business rules do not need to be specified in this format initially. The businesses could be provided with a more user-friendly interface instead. After the appropriate information has been entered by the businesses, the business rules could be translated into a format that can be used to reason with by the information flow planner. The business rules are added to the repository like any other context information. To

¹ This public key is shorter than a public key normally would be to enhance readability.

ensure that businesses only specify business rules for themselves, they should also add a version of the business rule encrypted with their private key.

The information flow planner of course also needs information to determine whether the conditions specified in the business rules are true. Much information, such as the roles of parties in a shipment, will already be accessible to the information flow planner as context information. Other information might need to be added by the businesses that are involved in sharing these data elements.

If information is shared according to the business rules, then this can be viewed as consent for that information sharing by these businesses. The business rules could have a juridical status. In this way, the juridical and the practical protection of information go hand in hand and it is based upon the exact same set of rules. The experts in IP law disagreed on whether this should be specified in the participation agreement that all businesses that want to use the context-aware architecture should sign.

In addition to the business rules, the businesses involved in the architecture should provide information on two more context elements. They should add context information on who they consider their competitor and provide information on *competitor(Business, SetofDataElements, Party)*. They should also specify the level of security they require for a system to share sets of data elements with it and provide information on *securityRequirement(Business, SetofDataElements, Level)*. If it is difficult to add information on individual data elements here as well, then in the same way as for specifying what data is sensitive and what parties are entitled to access data rules can be added that can be used by the information flow planner to derive it.

The last context element that concerns sensitivity of data for businesses is *tradesecrets(SetofDataElements, Business)*. Single data elements can be a trade secret as well as combinations of data elements or aggregated data elements, according to the experts. Data can become a trade secret once it is combined with other data. Public data is not a trade secret according to the experts in IP law. Namely, for something to be a trade secret, there need to be measures in place to keep it secret, such as confidentiality agreements. In addition, there should be commercial value in the data because it is secret. The interviewees were asked who would be in the best position to provide information on what data elements are trade secrets. According to the experts, the trade secret holders, so the businesses themselves will know.

10.2.2 Independent third parties

Some of the context information should be generated by other parties than the businesses themselves. This could be the case when access control relies on the identity of businesses, or when context information is too difficult to establish by them or requires an overview of the market. Therefore, we need two types of independent third parties to provide context information as well, namely identity managers and trusted parties.

10.2.2.1 Identity managers

The identity managers are appointed by the governance body of the context-aware architecture. They verify the identity of businesses and government organisations that

want to use the context-aware architecture and provide them and their system with identifiers and a certificate. Only such verified parties for which there is a certificate can use the context-aware architecture.

Making the architecture closed instead of open helps to deal with several issues. First, it can be ensured that all parties sign an agreement that they will follow certain rules, such as only sharing data using the information routers. This is important for the architecture to decide what flow of information is appropriate.

In addition, if the businesses using the architecture can be identified, even if only by a third party, there is less chance for them to behave in a way that might harm others. If they, for example, provide misleading information, they can be identified and called out on it. Other issues, such as digital vandalism can be reduced as well if not just anybody can become part of the architecture.

There could be a single independent third party. However, if the context-aware architecture is used at a large scale, it might be difficult for one party to do all the work of verifying the identities of all parties that want to connect to the context-aware architecture. Furthermore, depending on a single party might lead to issues with balancing power and it means putting all trust in a single party. In addition, the party issuing the certificates could have a great distance geographically and culturally (including language) from the parties that they should verify the identity of. Alternatively, there can thus be multiple identity managers and identity management can be federated (van Engelenburg, Janssen, Klievink, et al., 2018).

A party that verifies the identity of businesses and provides identifiers for them and their systems is also in a position to obtain information on these parties and their systems. When businesses apply for using the context-aware architecture, they should provide this information to the third party. It could be part of the participation agreement that the businesses will not change some of the properties of their system unless notifying the third party first so that the context information can be changed.

The identity managers could provide context information on the following context elements:

- *partyType(PartyType, Party)*
- *broadcasts(System)*
- *systemType(System, SystemType)*
- *geographicalScope(System, Scope)*
- *controlsDataAccess(System)*
- *securityProvided(System, Level)*
- *systemOf(System, Party)*
- *makesPublic(System, SetofDataElements)*

The identity managers might also perform audits. They could, for example, establish that a system actually provides the level of security that it is said to provide by its owner or establish whether a system for which there is a guarantee of confidentiality actually appropriate measures are taken to do so. This could help to improve the reliability of this context information.

10.2.2.2 Trusted parties

For some context elements, it is difficult to establish their value without having a broader overview. This is the case for determining what data sets are commercially sensitive for element *commerciallySensitive(SetofDataElements, Business, Competitor)* and for determining what data set could disturb the relevant market when shared for context element *disturbMarket(SetofDataElements)*.

The subject of what data could be viewed as competitively or commercially sensitive from a juridical point of view was explicitly subject to discussion during the interviews with the juridical experts. As discussed before, data elements often are not commercially sensitive by themselves, but only when aggregated or combined with other data elements. According to the juridical experts, it is highly difficult to determine whether data is commercially sensitive and this relies on many different variables.

For data to be commercially sensitive, it needs to have value. According to the experts, a set of data elements is commercially sensitive according to competition law if it can be used to predict the future market behaviour of competitors. The definition in the law is quite broad and everything that takes away uncertainty about future market behaviour is considered commercially sensitive. Furthermore, the businesses should be competitors for the services or goods with which the data elements are concerned.

Commercial sensitivity of data also depends on the market. For example, if a supply chain works on a small scale or only a few parties produce certain goods, then it might be easier to identify these parties in the data and derive information on them, such as their volumes. This is different in the case of large-scale supply chains or when many businesses produce the same type of goods.

Businesses will know what data has commercial value and is competitively sensitive. For the context rules related to the willingness of businesses, we rely on the businesses themselves to provide information on who they believe are their competitors (see section 10.2.1.3).

The expert advised involving a trusted party to do a market analysis and determine whether data is commercially sensitive when competition law is concerned. They suggest that such a trusted party could be an independent economist. They could take into account all these variables and based on their expert knowledge provide the appropriate context information. Instead of doing so at the level of individual data elements, they can add rules similar to the business rules that the information flow planner can use to derive what data elements are commercially sensitive. As the market seems of great influence, it might be suitable to assign a trusted party to each market that determines for that market what data is commercially sensitive.

What data disturbs the market when made public also depends on the market. Data is considered to disturb the market when making it public, if making it public would favour one party over the other. For example, because one party has the means to perform big data analysis on the data made available and the other does not. What data disturbs the market could be established by the same trusted party in each market.

In addition, there is the context element for the public nature of a set of data elements *public(SetofDataElements)*. The issue with this context element is that it might also be hard to determine what data is public and what data is not public.

Furthermore, correctly identifying data as public is important, as public data can be shared freely according to the law (see section 9.2.3.8). This context information should thus come from a reliable source.

Data elements that are part of the public record are considered public. Data elements that are public by themselves or in certain combinations might not be public anymore when they are combined with other data elements. For example, the route of a carrier can be public. However, combined with information that a specific shipment is carried via the route might not be. The route of the carrier, in that case, is not public anymore when connected to the goods.

10.2.3 Customs

Customs is the last party from which context information can be obtained. *obligation(Action, SetofDataElements, Shipment, GovernmentAgency, Role)* is the only remaining context element that needs to be sensed. Customs is the party who knows exactly what data businesses are obligated to share with them. They are thus the appropriate party to add context information on this to the context information repository.

10.3 Context rules

In the previous section, we derived the context relationships. As described in step 3 of the new method (section 6.3), deriving the context rules from these relationships does not require deeper analysis anymore. We merely need to put the context elements that will be manipulated by an adaptor in the head and the other context elements in the body of the context rules. Doing so creates the list of context rules in Table 34. When there are multiple (possible) adaptors for a context relationship, we have multiple rules for this context relationship. The information flow planner will use these context rules to determine the appropriate flow of information in different situations.

Context rule	Section
$\neg hasAccess(SetofDataElements, Party) \leftarrow sensitiveTo(SetofDataElements, Business, Party)$	9.1.2.1
$encrypted(SetofDataElements, Key) \leftarrow sensitiveTo(SetofDataElements, Business, Party_1), \neg entitled(access, SetofDataElements, Business, Party_2)$	9.1.2.2
$\neg hasAccess(\{Key\}, Party_2) \leftarrow sensitiveTo(SetofDataElements, Business, Party_1), \neg entitled(access, SetofDataElements, Business, Party_2)$	
$\neg canAggregate(Party, SetofDataElements) \leftarrow sensitiveToAggregate(SetofDataElements, Business, Party)$	9.1.2.3

Table 34: The context rules for the context-aware architecture and the section where we describe their context relationships

Context rule	Section
$\neg canAggregate(Party_2, SetofDataElements) \leftarrow$ <i>sensitiveToAggregate(SetofDataElements, Business, Party₁),</i> <i>partyType(customs, Party₂),</i> $\neg entitled(aggregate, SetofDataElements, Business, Party_2)$	9.1.2.4
$\neg canAggregate(Party_2, SetofDataElements) \leftarrow$ <i>partyType(customs, Party₂),</i> <i>sensitiveToAggregate(SetofDataElements, Business, Party₁)</i>	9.1.2.5
<i>view(Party₂, SetofDataElements) \leftarrow</i> <i>partyType(customs, Party₂),</i> <i>sensitiveToAggregate(SetofDataElements, Business, Party₁)</i>	
$\neg hasAccessSystem(SetofDataElements, System) \leftarrow$ <i>sensitiveTo(SetofDataElements, Business, Party),</i> <i>broadcasts(System)</i>	9.1.2.6
$\neg hasAccessSystem(SetofDataElements, System) \leftarrow$ <i>systemType(System, datapipeline),</i> <i>thick(SetofDataElements),</i> <i>geographicalScope(System, global)</i>	9.1.2.7
<i>hasAccessSystem(SetofDataElements, System) \leftarrow</i> <i>systemType(System, datapipeline),</i> <i>thick(SetofDataElements),</i> <i>geographicalScope(System, international),</i> <i>controlsDataAccess(System)</i>	9.1.2.8
<i>hasAccessSystem(SetofDataElements, System) \leftarrow</i> <i>systemType(System, datapipeline),</i> <i>geographicalScope(System, global)</i>	9.1.2.9
<i>thin(SetofDataElements) \leftarrow</i> <i>systemType(System, datapipeline),</i> <i>geographicalScope(System, global)</i>	
$\neg hasAccessSystem(SetofDataElements, System) \leftarrow$ <i>sensitiveTo(SetofDataElements, Business, Party),</i> <i>securityRequirement(Business, SetofDataElements, Level₁),</i> <i>securityProvided(System, Level₂),</i> <i>lower(Level₁, Level₂)</i>	9.1.2.10
$\neg hasAccess(SetofDataElements, Party) \leftarrow$ <i>subject(SetofDataElements, Shipment),</i> $\neg connection(Party, Shipment)$	9.1.2.11

Table 34 (continued): The context rules for the context-aware architecture and the section where we describe their context relationships

Context rule	Section
$\text{filter} \left(\begin{array}{c} \text{SetofDataElements}, \\ \text{FilteredSetofDataElements}, \text{Business} \end{array} \right) \leftarrow$ $\text{sensitiveTo}(\text{SetofDataElements}, \text{Business}, \text{Party})$	9.1.2.12
$\text{hasAccess}(\text{FilteredSetofDataElements}, \text{Party}) \leftarrow$ $\text{sensitiveTo}(\text{SetofDataElements}, \text{Business}, \text{Party})$	
$\text{routeinFlow}(\text{ListofSystems}, \text{SetofDataElements}) \leftarrow$ $\text{obligation} \left(\begin{array}{c} \text{submit}, \text{SetofDataElements}, \\ \text{Shipment}, \text{GovernmentAgency}, \text{Role} \end{array} \right),$ $\text{hasRole}(\text{Party}, \text{Shipment}, \text{Role}),$ $\text{systemOf}(\text{System}_1, \text{Party}),$ $\text{systemOf}(\text{System}_2, \text{GovernmentAgency}),$ $\text{index}(\text{System}_1, \text{ListofSystems}) < \text{index}(\text{System}_2, \text{ListofSystems}),$ $\neg \text{encrypted}(\text{SetofDataElements})$	9.2.3.1
$\text{routeinFlow}(\text{ListofSystems}, \text{SetofDataElements}) \leftarrow$ $\text{obligation} \left(\begin{array}{c} \text{submit}, \text{SetofDataElements}, \\ \text{Shipment}, \text{GovernmentAgency}, \text{Role} \end{array} \right),$ $\text{hasRole}(\text{Party}_2, \text{Shipment}, \text{Role}),$ $\text{authorized} \left(\begin{array}{c} \text{Party}_2, \text{Party}_1, \text{submit}, \text{SetofDataElements}, \\ \text{Shipment}, \text{GovernmentAgency} \end{array} \right),$ $\text{systemOf}(\text{System}_1, \text{Party}_1),$ $\text{systemOf}(\text{System}_2, \text{GovernmentAgency}),$ $\text{index}(\text{System}_1, \text{ListofSystems}) < \text{index}(\text{System}_2, \text{ListofSystems}),$ $\neg \text{encrypted}(\text{SetofDataElements})$	9.2.3.2
$\neg \text{routeinFlow}(\text{ListofSystems}, \text{SetofDataElements}) \leftarrow$ $\neg \text{public}(\text{SetofDataElements}),$ $\text{systemOf}(\text{System}, \text{Party}_1),$ $\text{index}(\text{System}, \text{ListofSystems}) < \text{length}(\text{ListofSystems}),$ $\text{agreement}(\text{Party}_2, \text{Party}_1, \text{notshare}, \text{SetofDataElements})$	9.2.3.3
$\text{routeinFlow}(\text{ListofSystems}, \text{SetofDataElements}) \leftarrow$ $\text{elementsInSet}(\text{ListofSystems}, \text{SetofParties}),$ $\text{tradesecret}(\text{SetofDataElements}, \text{Business}),$ $\text{agreements} \left(\begin{array}{c} \text{Business}, \text{SetofParties}, \\ \text{notshare}, \text{SetofDataElements} \end{array} \right)$	9.2.3.4
$\neg \text{routeinFlow}(\text{ListofSystems}, \text{SetofDataElements}) \leftarrow$ $\text{commerciallySensitive} \left(\begin{array}{c} \text{SetofDataElements}, \\ \text{Business}, \text{Competitor} \end{array} \right),$ $\neg \text{public}(\text{SetofDataElements}),$ $\text{systemOf}(\text{System}_1, \text{Business}),$ $\text{possibleAccess}(\text{System}_2, \text{SetofDataElements}, \text{Competitor}),$ $\text{index}(\text{System}_1, \text{ListofSystems}) < \text{index}(\text{System}_2, \text{ListofSystems})$	9.2.3.5

Table 34 (continued): The context rules for the context-aware architecture and the section where we describe their context relationships

Context rule	Section
<i>encrypted(SetofDataElements, Key) ← commerciallySensitive(</i> <i>SetofDataElementsBusiness,</i> <i>Competitor)</i>	9.2.3.6
<i>¬hasAccess({Key}, Competitor) ← commerciallySensitive(</i> <i>SetofDataElementsBusiness,</i> <i>Competitor)</i>	
<i>¬routeinFlow(ListofSystems, SetofDataElements) ← disturbMarket(SetofDataElements),</i> <i>elementOf(System, ListofSystems),</i> <i>makesPublic(System, SetofDataElements)</i>	9.2.3.7
<i>routeinFlow(ListofSystems, SetofDataElements) ← public(SetofDataElements)</i>	9.2.3.8

Table 34 (continued): The context rules for the context-aware architecture and the section where we describe their context relationships

11 The basic components for context-awareness

Adaptors, sensors and context rules by themselves are not enough to make a system or an architecture context-aware. We also need some components that any context-aware information sharing architecture would need, for which whether they are needed, and their role thus does not depend on the environment in which the architecture is used. More specifically, we need to store context information generated by the sensors and we need a decision component to make decisions using the context rules based on the context information. Furthermore, we need components to share the information according to the information flow upon which the decision component decides.

As we discuss in section 10.1.1, ensuring that the information is shared according to the appropriate flow of information is done by information routers. The information routers are needed in any context-aware information sharing architecture; however, they also manipulate some of the context elements. Therefore, they are adaptors as well as basic components. We thus discuss the information routers in section 10.1.1, instead of in this chapter. In this chapter, we discuss the other basic components, namely, the context information repository and the decision component.

We derived the additional basic components that our architecture requires from an existing architectural pattern described in the literature in section 11.1. The basic components should contribute to the goals of the architecture discussed in section 7.5.1 just like any other part of the architecture. We discuss these goals and the way in which the basic components could contribute to reaching them in section 11.2. We identified blockchain technology as providing an opportunity to reach our goals. Before we apply the technology, we need insight into its advantages and disadvantages. To obtain this insight, we describe its basics and typical characteristics in section 11.3. Finally, in section 11.4 and section 11.5, we present the decision component and the context information repository, including their use of blockchain technology.

Parts of this chapter have been published in van Engelenburg, Janssen and Klievink (n.d.) and Tan, Rukanova, van Engelenburg, Janssen and Ubacht (n.d.).

11.1 Architectural pattern and basic components

In section 3.3.3, we already provided an overview of the tools, guidelines and frameworks for designing context-aware systems in the literature. We concluded that this work does not support determining what belongs to the context of a context-aware system or architecture and determining what sensors, adaptors and context rules are required. However, this work does discuss what other parts of the architecture could look like.

Alegre et al. (2016) identify from the literature different architectural patterns (i.e., architectural design decisions applicable to recurring design problems) for context-aware systems. The patterns that they identify are context sources and managers hierarchy, blackboards, event-control-action, sense-compute-control, and actions pattern. Of these patterns, the event-control-action pattern seems to be closest to what is required for our architecture.

In the event-control-action pattern, there are three sets of tasks. The first set is the event-tasks that are concerned with the gathering and processing of context

information (Alegre et al., 2016; Dockhorn Costa, Ferreira Pires, & van Sinderen, 2005). The second set is the control-tasks, which are concerned with connecting events to actions (Alegre et al., 2016; Dockhorn Costa et al., 2005). The last set is the action-tasks, which are concerned with the behaviour of an application (Alegre et al., 2016; Dockhorn Costa et al., 2005).

This architectural pattern offers flexibility and extensibility and the distribution of responsibility to provide functionality amongst different parties (Alegre et al., 2016; Dockhorn Costa et al., 2005). This fits very well with the idea that our architecture overarches a variety of information systems providing various functionality, both existing and those that are still to be developed. Furthermore, this pattern uses conditional rules in which the condition specifies a situation in which an action in the head should be performed (Dockhorn Costa et al., 2005). These conditions are made up of several logical events (Dockhorn Costa et al., 2005). These conditional rules are very similar to our context rules. The events are similar to the sensor elements in those rules, while the heads of those rules are similar to adaptor elements.

The event-control-action pattern relies on the events triggering actions. This is not in conformance with what we need. We do not want information sharing to be triggered just because a business adds a rule that another business is allowed access to a set of data, for example. Being allowed access to data does not mean that actions should be undertaken to provide access to data. The other business might not even want the data. This would lead to a lot of potentially unnecessary information sharing. This means that we need to alter the process provided by this pattern by letting users trigger the architecture to share information.

For the event-tasks, components are required to gather and process context information. In the case of the context-aware architecture, context information is gathered by *sensors*, which have already been specified in section 10.2. We expect them to provide the context information already in the appropriate format, namely as literals or access control rules based on the syntax defined in section 5.2.2. This means that they also perform part of the processing of the context information. However, the context information also needs to be combined and stored somewhere before it can be used to make a decision. We thus need a *context information repository* to be a component of the architecture as well.

For the control tasks, events need to be connected to actions. This means that decisions need to be made on what actions to perform based on context information. The architecture thus also requires a *decision component*. As the architecture adapts to provide for a certain flow of information, this component should determine the appropriate flow of information based on the context information and context rules.

For the action-tasks, the actions decided upon by the decision component should be performed. This means in our case that the information should be shared according to the information flow decided upon by the decision component. This function is fulfilled by the *adaptors* in our architecture, including the information routers, which we already presented in section 10.1.

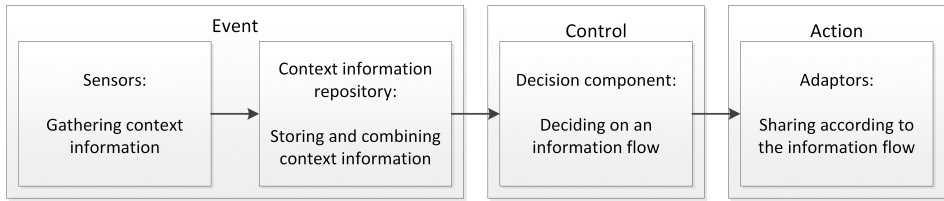


Figure 12: Components of the context-aware architecture based on the event-control-action pattern

Figure 12 shows the components needed in the architecture to perform the tasks in the event-control-action pattern. As we already have described the sensors and adaptors, what is left is to describe the context information repository and the decision component.

11.2 Meeting the goals for the architecture

In section 7.5.1, we specified the goals for the overall architecture, namely providing for information flows in which the businesses supplying information are willing to participate and that are lawful. Based on that, we established foci that can be used to determine how the architecture should adapt in different situations to meet these goals. Furthermore, we derived the sensors and adaptors the architecture requires based on these foci.

The decision component and the context information repository, however, should contribute to meeting these goals as well. This means that businesses should be willing to use them and that their use should be lawful. By providing different flows of information, the architecture effectively provides context-aware access control. However, the reliability and the trust of businesses in this access control depend on the decision component as well as the context information repository. If either of these is tampered with, there is a risk of information sharing that is not lawful. Furthermore, if businesses do not trust the way in which access control is arranged, they might not be willing to use the architecture.

It might be highly difficult to find a party that everyone can agree can be trusted to store all context information and make the decisions according to which information flow information will be shared. Such a party would have considerable power over data sharing in the architecture. The architecture is meant to be overarching and used at large scale. This means that if a party is chosen that turns out not to be trustworthy or that fails otherwise, there could be harm at a large scale as well.

In the literature, there are several proposals to use blockchain technology to provide for secure access control for various types of data and in various domains (see e.g., (Ouaddah, Abou Elkalim, & Ait Ouahman, 2016; Zyskind, Nathan, & Pentland, 2015)). In the next section, we discuss the typical characteristics of blockchain technology. Its advantages are that data stored on a blockchain is difficult to tamper with and transparent, control of data is decentralised and that it is fault tolerant.

If we store context information, including access history and access control rules of businesses as well as the context rules on a blockchain, they would become difficult to tamper with without requiring control of a central party. This could improve trust and security of access control. Furthermore, we could store the information flows that the

decision component decides upon on a blockchain as well. This makes these decisions auditable, as the input for the decision as well as the output is difficult to tamper with in that case. This means businesses do not fully need to rely on trusting the party governing the decision component either.

We need a thorough understanding of blockchain technology and its advantages and risks before we can apply it effectively. Before discussing the decision component and the context information repository, we will thus provide a short general overview of the basics of blockchain technology and its typical characteristics. This will help us to make full use of its advantages while reducing its disadvantages as much as possible.

11.3 Blockchain technology and its typical characteristics

As discussed in the previous section, we need to understand blockchain technology, its typical characteristics and its advantages and risks before we can use it in the architecture as a component to store context information. Blockchain was originally conceptualised by Nakamoto in 2008. The original purpose of blockchain technology was to solve the double-spending problem without requiring a central intermediary for the cryptocurrency Bitcoin (Nakamoto, 2008). Nakamoto used blockchain technology to store and share transactions in a chain of blocks in a distributed network such that 1) transactions are difficult to change once stored in the blockchain, 2) any node can check whether a transaction is valid, and 3) a transaction is considered to be accepted if the nodes with 51% of the CPU accept the transaction.

There are many different possible designs in which information sharing can be supported using blockchain technology. However, basic blockchain technology, such as proposed by Nakamoto (2008), comes with certain characteristics that are quite typical, for example, difficulty to tamper with data and issues with scalability. We discuss the basics of blockchain technology in the first subsection of this section. Next, we discuss its typical characteristics. These characteristics can be viewed as a point on a scale for different dimensions of information infrastructures, such as transparency and scalability. We will use the insight into these characteristics to make choices for how blockchain technology is applied in our architecture.

11.3.1 Blockchain basics

Blockchain technology is based on several components: 1) a chain of blocks with data (i.e. blockchain or distributed ledger), 2) a distributed network of nodes storing the blockchain, and 3) a consensus mechanism for deciding what blocks are acceptable (Nakamoto, 2008). Sometimes other components are mentioned as well, such as an incentive to add new blocks or tokenisation (see e.g., (Nakamoto, 2008; Tasca & Tessone, 2017)).

The data that is shared and stored using blockchain technology is shared and stored in blocks (Nakamoto, 2008). Each of these blocks consists of a header and a body (Nakamoto, 2008). The body contains the data (Nakamoto, 2008). This could be transactions in the case of cryptocurrencies (Nakamoto, 2008), but when blockchain technologies are used for other purposes other data, such as logistics event data, trade

documents (e.g. purchase order, invoice, packing list, bill of lading), or context information can be stored in the body instead (see e.g., (van Engelenburg, Janssen, & Klievink, 2017a)).

The header of a block contains two hashes. The first one is a Merkle root, which is the root of a tree of the hashes of the data in the body of a block (Nakamoto, 2008). Figure 13 provides an example of a Merkle tree of hashes of four data elements in a block.

The Merkle tree is built as described by Merkle (1987) and Massias et al. (1999). If we consider figure 13, data elements $d1, \dots, d4$ are hashed in to the hashes $H(d1), \dots, H(d4)$, respectively (van Engelenburg, Janssen, & Klievink, 2017a). These are the leaves of the Merkle tree (van Engelenburg, Janssen, & Klievink, 2017a). The leaves are concatenated by two, denoted by $H(d1)|H(d2)$ and $H(d3)|H(d4)$ (van Engelenburg, Janssen, & Klievink, 2017a). These concatenations are then hashed to single values, viz. $H(H(d1)|H(d2))$ and $H(H(d3)|H(d4))$ (van Engelenburg, Janssen, & Klievink, 2017a). The parent hashes obtained in that manner are then again concatenated by two and hashed again (van Engelenburg, Janssen, & Klievink, 2017a). This goes on until there is only a single hash left, the Merkle root (van Engelenburg, Janssen, & Klievink, 2017a).

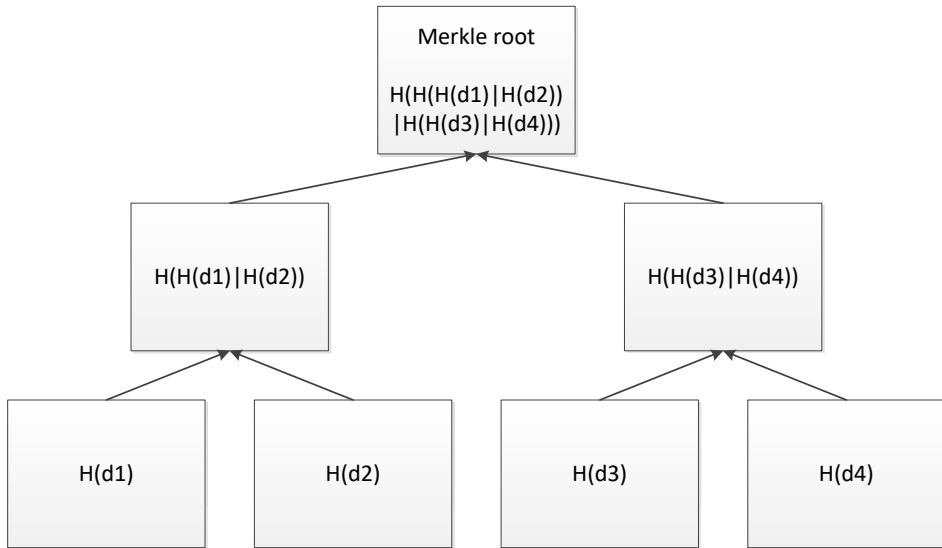


Figure 13: Example of a Merkle tree for four data elements (van Engelenburg, Janssen, & Klievink, 2017a)

A Merkle root is unique for the data elements it was built upon and it can be used to prove the existence of data elements in the tree (Luu et al., 2015; Merkle, 1987). The storing of a Merkle root in the header of a block, therefore, allows for checking whether data elements in the body of a block have been changed (Nakamoto, 2008).

The second hash that is stored in the header of blocks, is a hash of the header of the previous block in the blockchain (Nakamoto, 2008). Storing such a hash, links blocks to each other in a chain. The hash stored in the header of a block can be used to check

whether the header of the previous block has been changed, including its Merkle root (Nakamoto, 2008). This contributes to the high immutability of blockchain technology (see section 11.3.2.1).

Usually, the full blockchain is stored by each node in a distributed blockchain network (Nakamoto, 2008). The network might consist of different parties and organisations. There are various types of blockchain networks. When the network is public, such as in the case of Bitcoin, anyone can become a node in the network (Buterin, 2015). In fully private networks, a central party decides who can be a node in the blockchain (Buterin, 2015; Pilkington, 2016). There are also types of networks that are in between, such as consortium blockchain networks (Pilkington, 2016). Another distinction is commonly made between permissionless and permissioned blockchains. In the former type of blockchain network, any node can accept, reject and add new blocks, while in the latter only certain parties can do so (Walport, 2015).

Typically, transactions or other data are distributed throughout the network by parties, and nodes collect them in a block (Nakamoto, 2008). This block is then added to the existing chain of blocks by the nodes (Nakamoto, 2008). The other nodes then accept or reject this block according to a consensus mechanism (Nakamoto, 2008). The consensus mechanism used in the case of Bitcoin is proof of work (Nakamoto, 2008).

Proof of work requires nodes to ‘mine’ a block before they can add it to the chain. This mining involves finding a solution to a computationally hard problem that requires vast CPU effort, and is unique for the block to be added to the chain (Nakamoto, 2008). While it requires a lot of CPU to find the solution, it is easy for other nodes to establish whether the miner actually found the solution (Nakamoto, 2008). Miners provide proof that they found a solution by adding a number (i.e. a nonce) to the header of the new block, which the other nodes use to check the solution (Nakamoto, 2008). In the case of Bitcoin, a nonce is a number that, when it is added to the header, causes its hash to start with a certain number of zero bits (Nakamoto, 2008).

Nodes that accept a block can express this by adding new blocks after it in the chain (Nakamoto, 2008). If the block is accepted the miner receives a fee (Nakamoto, 2008). The longest chain is considered the ‘true’ chain and expresses the consensus of the nodes with 51% of the CPU power on what blocks should be accepted (Nakamoto, 2008). Namely, the nodes with the majority of CPU power will be able to add blocks faster than the other nodes in the network (Nakamoto, 2008).

Proof of work has various disadvantages, for example, it is difficult to scale and it requires a lot of electricity. Various other consensus mechanisms thus have been developed. An example is proof of stake, in which the nodes that have a higher stake, for example, because they have more cryptocurrency, have a higher chance to get selected to determine whether a new block should be accepted (see e.g., (Kiayias, Russell, David, & Oliynykov, 2017)). Another example is proof of authority in which the network is permissioned and some nodes are authorised to accept or reject blocks (Tasca & Tessone, 2017).

Blockchain technology is often used to store smart contracts. According to Delmolino et al. (2016, p. 1), smart contracts are *“user-defined programs that specify rules governing transactions, and that are enforced by a network of peers”*. Smart

contracts can be used to specify conditions in a contract and to execute ‘automatically’ certain actions when the conditions are met. For example, a seller and a buyer who do not know or trust each other can both put cryptocurrency in a smart contract (Dannen, 2017). The smart contract will automatically transfer the currency back to their wallets when the buyer sends a message that they have received their goods (Dannen, 2017).

In general, a user writes a smart contract in a programming language for smart contracts, for example, Solidity for Ethereum (Dannen, 2017). Storing a smart contract in a blockchain makes the code difficult to change, just as other data stored in a blockchain. Smart contracts can be triggered and executed in different ways. A smart contract can, for example, be triggered when it receives a message from another user or contract (Crosby, Nachiappan, Pattanayak, Verma, & Kalyanaraman, 2016). When the smart contract is triggered the code could be run by a node that adds the result of running the code in a new block to the chain (Dannen, 2017). The other nodes that validate the block can run the code as well to check the result (Dannen, 2017).

Blockchain technology can also be used to store a proof of existence (Crosby et al., 2016). Parties can store a hash of data, such as a declaration or other document, in the blockchain. Each hash is unique for the data and this means that the data should have existed in order to calculate the hash. As it is difficult to change or remove data, the stored hash in the blockchain ‘proves’ that the data existed when it was added to the blockchain.

This can also be used to check whether data for which a proof of existence is stored has been tampered with or changed. Namely, if this happens the hash does not fit the data anymore. In some cases, parties rather store a proof of existence in the blockchain than the actual data. It is difficult to protect the confidentiality of data stored on a blockchain. As is it very difficult, if not impossible, to translate a hash back to its original data, storing only the hash might be safer. In some cases, a link to the URL where the data for which the hash is generated is stored is added to the blockchain as well. Such a combination of a hash and a URL reference is referred to as a ‘hash pointer’.

11.3.2 Typical characteristics of blockchain technology

We have identified six typical characteristics of basic blockchain technology, namely 1) high immutability, 2) decentralised control over data, 3) high transparency, 4) high fault tolerance, 5) low scalability, and 6) low data confidentiality. Characteristics 1, 2, 3 and 4 contribute to solving the double-spending problem without requiring an intermediary, which was the original purpose of blockchain technology. Characteristic 5 and 6, however, seem to be more like a mere consequence of the design decisions made. In this subsection, we discuss each of these characteristics and dimensions as well as the tensions between them.

11.3.2.1 High immutability

Immutability can be viewed as the level of difficulty of changing information once it is stored. There are several design choices that contribute to data being difficult to change once it is stored in a block and a block is accepted in a chain. As already discussed, in the

header of a block, the Merkle root of the transactions or other data elements are stored. This Merkle root is unique for the data in the body.

Given a block i , if somebody tampers with a data element in its body, then the Merkle root stored in its header does not fit with the data in the body anymore. In this way, such tampering can be detected, unless somebody tampers with the Merkle root of block i as well. However, such tampering with the Merkle root of block i can also be detected. Namely, in the header of each block, a hash is stored that is unique to the header of the previous block in the chain. Thus, in the header of block $i + 1$, a hash is stored of the header of block i , which includes its Merkle root. If the Merkle root of block i is tampered with, then the hash of block $i + 1$ will not fit anymore, unless this header is tampered with as well. Again, this can be detected as well, as the header in block $i + 2$ contains a hash unique to the header of block $i + 1$, and so on. This means that the more blocks are added after a block, the more blocks need to be changed to make tampering with data undetectable (Nakamoto, 2008).

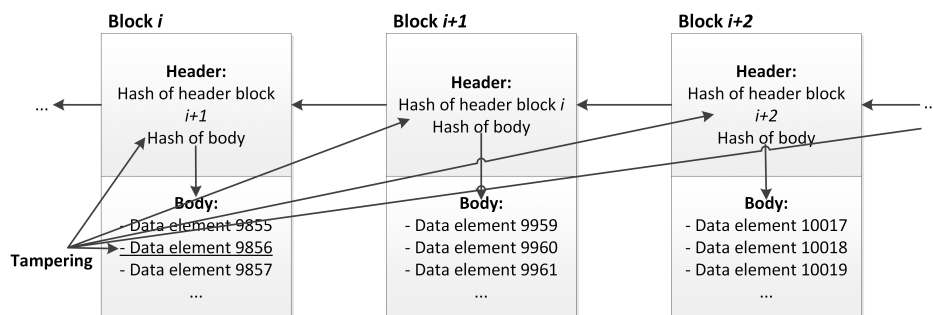


Figure 14: A blockchain with the elements that need to be tampered with in red to hide that ‘Data element 9856’ has been tampered with (adapted from (Nakamoto, 2008))

Something else that contributes to making it difficult to change data that is stored in a blockchain is that all nodes in the network store the full blockchain. If one node tampers with data, then all the other nodes still have the original data that has not been tampered with. Furthermore, the longest chain of blocks is typically considered the ‘true’ blockchain that expressing consensus on what blocks are accepted by the network (Nakamoto, 2008). A party, or parties, that want to tamper with data stored in a blockchain will need to add blocks after the block that they have tampered with faster than the rest of the network (Nakamoto, 2008). Namely, this how they can ensure that the tampered data will be part of the true, accepted blockchain. Depending on the consensus mechanism, this can be very difficult. For example, in the case of proof of work, parties would need to have more CPU power than the rest of the network together, as it requires a lot of CPU power to add blocks (Nakamoto, 2008). In the example of proof of stake, they would need to have the most stakes, as the chances that parties are chosen to add a new block depends on their stakes.

The mechanism for protecting against tampering with data in blocks means that it becomes more difficult to change data in a block over time when more and more blocks are added after it (Nakamoto, 2008). Namely, this means that making changes

undetectable would require adding more blocks after it to generate the longest chain (Nakamoto, 2008). This is also the reason why in the case of cryptocurrency a transaction is considered 'confirmed' only after a certain number of blocks are added after the block where it is stored.

It is important to note that while it is typically very difficult to change data stored in a blockchain, this is not impossible (Barber, Boyen, Shi, & Uzun, 2012). In the case of proof of work, for example, a 51% attack is possible, in which the nodes with 51% of the CPU make or accept changes to the data stored in the blockchain (Barber et al., 2012). In the case of proof of work and in the case of other consensus mechanisms, it could be possible to control nodes via vulnerabilities in software that are commonly used by the nodes. It could even be the case that the software used by the nodes to become part of the blockchain network and contribute to consensus is vulnerable to illegitimate control by others, intentionally or unintentionally.

11.3.2.2 Decentralised control over data

Control over data we can view as the ability to make decisions on whether it is stored, shared, changed, or deleted. When control over data is centralised, then one party makes these decisions. When control over data is decentralised, then more parties are involved with decision-making.

Blockchain technology offers decentralised control over data. This is due to its consensus mechanism and its transparency. Exactly what parties control the data depends on the consensus mechanism and who can add new blocks according to this mechanism. Namely, these nodes can create the longest chain.

There is a high variety of consensus mechanisms possible. We already discussed proof of work, in which the nodes with 51% of the CPU control the data and proof of stake in which the nodes with the highest stakes control the data. In the case of proof of authority, certain nodes are authorised to accept blocks and add new blocks after (Tasca & Tessone, 2017). Of course, if only one node has this authorisation, and maybe even if there is only one party authorising nodes then it becomes questionable whether control of data can still be considered decentralised.

In the case of Bitcoin, Nakamoto (2008) expressed concern with the whole money system having to depend on a single party that has to create new coins (i.e., a mint) and verify transactions (i.e., a bank). Bitcoin thus was developed to allow for decentralised control over data. However, this characteristic can also be useful in other cases where no single party is considered trustworthy enough to control data.

11.3.2.3 High transparency

Transparency can be viewed as the ease with which data can be viewed. In the case of blockchain technology, all nodes can store the blockchain and all data in it (Nakamoto, 2008). Who can be nodes in the network, and thus what parties can store the data in the blockchain depends on whether the network is public or private. In case a network is public, anybody potentially can become a node and be part of the blockchain. Blockchain technology thus can make the data that is stored in blocks highly transparent.

In the case of Bitcoin, as already mentioned, transparency of the transactions in the blocks contributes to the decentralised control over the data. If all nodes can accept or reject blocks, they should be able to determine whether the transactions are valid (Nakamoto, 2008). They can do so if they have a complete history of all transactions for each Bitcoin that is stored sequentially (Nakamoto, 2008). Namely, this allows nodes to check whether a party making a transaction actually owns the bitcoin they are transferring and whether they have not already spent it previously (Nakamoto, 2008). If double-spending happens, and a party tries to make two transactions with the same Bitcoin, then there is an agreement that a block containing the second transaction is rejected (Nakamoto, 2008). Such transparency of the data stored in the blockchain can have other applications as well. For example to make goods traceable, or to reduce information asymmetry between parties in a supply chain and thereby reducing the bullwhip effect (van Engelenburg, Janssen, & Klievink, 2018).

11.3.2.4 High fault tolerance

Fault tolerance can be viewed as the extent to which the infrastructure still functions when components fail (Randell, 1975). Blockchain technology typically has high fault tolerance, due to its decentral nature. If one or more nodes fail, then the other nodes still can maintain the blockchain and keep sharing and storing data.

11.3.2.5 Low scalability

Scalability can be viewed as the ability to change the levels of parameters that capture the performance aspects of a system (Ross, Rhodes, & Hastings, 2008). These parameters include the volume of data that is shared using blockchain technology and the number of nodes in the network. Blockchain technology has some notorious issues with scalability (see e.g., (Androulaki et al., 2018; Eyal, Gencer, Sirer, & van Renesse, 2016; Vukolić, 2016)).

There are several reasons for the low scalability of blockchain technology. First, nodes store the complete blockchain and no data is deleted (Nakamoto, 2008). We just discussed that this contributes to the other typical characteristics of blockchain technology, which might have advantages. Over time, however, this means that the chain becomes longer and longer and nodes need to store more and more data redundantly. For example, at the end of October 2018, the full size of the Bitcoin blockchain was over 176 GiB (“Blockchain size,” 2018). Nakamoto (2008) already noted this issue and proposed ‘simplified payment verification’. In his solution, nodes only store the headers of blocks and they do not fully verify transactions by themselves (Nakamoto, 2008). Instead, they check whether the previous transaction for the bitcoin was accepted by the network. They do so by using a timestamp to find the accepted block that should contain this previous transaction and by using the Merkle root stored in its header to determine whether it actually contains the previous transaction (Nakamoto, 2008).

Nakamoto (2008) already states that this solution leaves the network more vulnerable to attacks. This thus reduces immutability. Furthermore, this solution might not work when other data is shared than transactions. For example, when documents are shared using blockchain technology, nodes do not only need to have the data to determine

whether to accept a block, but they might actually be interested in its content and therefore they might want to have it. However, it is difficult, if not impossible, to establish whether this is the case by only looking at the Merkle root in the header of blocks.

There might be bigger issues even with the volume of data a blockchain network can add to the chain over time. For example, in the case of Bitcoin, a block can contain a maximum of 4000 transactions (Narayanan, Bonneau, Felten, Miller, & Goldfeder, 2016). As blocks are added every 10 minutes, Bitcoin can only handle 7 transactions per second (Narayanan et al., 2016). For comparison, VISA (2014) claims that in 2014 their network could process more than 56.000 transactions per second.

There are various reasons for these issues. A major culprit is proof of work (Vukolić, 2016). In the case of Bitcoin, the rate by which block can be added is determined by how fast a proof of work can be found. The requirements on the proof of work, and thus the difficulty of finding it, are adjusted over time such that a new block always is added to the chain about every 10 minutes (Nakamoto, 2008). The reason for keeping the difficulty of the proof of work at a certain level is that immutability is improved if it is difficult to tamper with data and then generate the new longest chain (Nakamoto, 2008). Many initiatives aim to improve the scalability of blockchain technology, but it remains an open issue.

11.3.2.6 Low data confidentiality

Data confidentiality can be viewed as the extent to which it is possible to restrict access to data to some parties while providing access to others. In the case of blockchain technology, it is very difficult to keep the data stored in the blockchain confidential. All nodes in the blockchain network can store all the data in the blockchain. This means that in principle, they all can access it. Furthermore, each of these nodes is a possible point of attack where illegitimate access to the data might be gained. This can be an issue when other data is stored than transactions. When contracts are stored, for example, they might contain information that is sensitive, such as prices. Furthermore, in some cases, it might even be unlawful to share such data, for example, because it causes unfair competition (van Engelenburg, Janssen, & Klievink, 2018).

There are some approaches to provide a higher level of confidentiality. However, they reduce the other possibly beneficial characteristics of blockchain technology, namely high immutability, decentralised control over data, high transparency, and high fault tolerance.

The first approach is to only include nodes in the network that should have access to the data. A large reduction in the number of nodes, however, means that there is a (central) party that decides who can be in the network and therefore indirectly controls the data and this reduces decentralisation. The fact that fewer nodes contribute to consensus if there are fewer nodes in the network can be viewed as a reduction of decentralisation as well. Furthermore, if there are fewer nodes in the network, then fewer nodes need to be compromised in order to make tampering with data possible. This thus reduces immutability. Additionally, to get access to the data, parties first need to be approved as a node, which reduces transparency. Last, but not least, if few nodes remain in the network, then fault tolerance also could be reduced.

Another approach is to encrypt the data that is stored in the blockchain and only share a key with the appropriate parties. However, this approach has some downsides as well. First, it means that there needs to be an additional way to share the keys, with all the additional risks and costs. This also reduces transparency, as the data needs to be obtained as well as a key. In addition, encryption makes it more difficult to establish whether a block should be accepted based on the content of that block. Only parties that have a key will be able to do so. Furthermore, the encrypted data can no longer be used as input for smart contracts, as these would need to incorporate a decryption key. This decryption key is stored by all nodes in that case as all of them store the smart contracts, including those that should not have access. Another issue is that any encryption can be decrypted brute force eventually, even though it might be a lot of effort for high-quality encryption. Every node that has the encrypted data could attempt this.

A third possible solution is to store a proof of existence for the data to be shared. This would keep the data confidential from all nodes in the network. This, of course, severely hampers transparency, as the data itself is no longer shared. Sharing a proof of work reduces immutability as well. It allows for the detection of the tampering of data, as in that case the proof of existence would not fit the data anymore. However, as the data itself is not stored in the blockchain and shared with others, it is not possible to obtain the original data from before it was tampered with.

11.4 Decision component: information flow planner

The decision component should decide on an appropriate information flow based on context rules and context information and business rules provided by the sensors and stored in the context information repository. The adaptors will then ensure that information is shared according to this information flow. As the decision component decides on what information flow is appropriate in our case, we call it an *information flow planner*.

To decide on a flow of information, the planner uses context rules and business rules provided by the businesses. The context rules, in essence, check whether flows of information are lawful and whether businesses are willing to participate in it. This means that they will not be used to generate an appropriate flow of information.

The decision system should work according to the following steps:

1. Receive request for a new flow of information from an information router (see section 10.1.1).
2. Generate a possible information flow
3. Test possible information flow using context rules and context information
4. Repeat step 1-3 until an appropriate flow of information is found
5. Share new flow of information with information routers

The efficiency of this process depends on step 2. The request the planner receives contains an identifier of the information router making the request, a set of identifiers of data elements, and an identifier of the information router with which the set of data elements is requested to be shared. This reduces greatly the set of information flows that should and can be taken into consideration by the planner. First, only flows ending with the information router of the system that needs to receive the data need to be taken into

consideration. Furthermore, only flows starting with an information router of a system that has (part of) the set of data elements according to their access history need to be taken into consideration.

The checking of different flows also could be prioritised to improve further efficiency. First, if a requester of the flow is also a party that has the set of data elements, they are likely willing to share. Secondly, shorter flows of information contain fewer systems for which sharing could be unlawful, or for which businesses are unwilling to participate in information sharing. Such flows should be prioritised as well.

The scope of the information flow that the information flow planner should assess is the full information flow from the system that originally shared the data, to the system with which the data needs to be shared. For example, business A could have generated data and business B could have previously requested a flow of information and obtained the data. Now, if a business C, later on, wants to have the data as well, then it is not enough to consider only the flow of information from business B to business C. For example, if A and C are competitors and the data is commercially sensitive, such a flow of information could be unlawful. If the information flow planner only takes into consideration the flow of information between B and C, then the context rule prohibiting the sharing with C would not fire and the architecture could provide for a flow of information that is not lawful. In addition, A is probably not willing to participate in this flow of information.

The flow of information assessed by the information flow planner, therefore, needs to be from the source of the data to the receiving system. However, this means that we need to bind the data elements to a source. The easiest way to do this is by letting a business that has generated the data add the first access history statement to the context information repository. This means that according to the information flow planner, this business will always be considered part of the flow of information. For an information flow to be considered appropriate, all businesses in it sharing the data should be willing to share. Their willingness is thus taken into account. They are able to control the access to the information by adding context information about what parties they consider the data sensitive from and what parties they consider entitled to access the data elements. As the identifiers of data elements are their hashes, the first business adding them in an access history statement must have had access to the data elements. This reduces the chances of parties trying to control access to data that they never had access to themselves.

The output of the information flow planner is an information flow that is appropriate according to the context rules. An information flow proposed by the information flow planner consists of the following:

- Set of identifiers of data elements
- Sequence of identifiers of information routers

The set of identifiers of data elements denote the data shared in the flow. The sequence of identifiers of information routers denotes the proposed flow of information. If an information router receives a proposed flow of information, then they automatically send the set of data elements specified to the next information router in the sequence.

When the context-aware architecture is used at a large scale, a centralised decision component might become a bottleneck that reduces scalability. In that case, an information flow planner could be assigned per market, for instance.

11.5 A blockchain-based context information repository

In section 11.2, we discussed that to meet the goals for the architecture, access control provided by the architecture should be secure. Furthermore, we discussed that if the architecture is used at a large scale, it might be difficult to find a party that is trusted by all parties to control the data shared. In section 11.3, we discussed blockchain technology and its typical characteristics. The characteristics that are advantages in our case are high immutability, decentralised control over data and high fault tolerance.

We can use blockchain technology as a basis for the context information repository to make it difficult to tamper with the context information, context rules and decisions made by the information flow planner on which access control is based, without businesses having to put full trust in a single party. By storing the information flows in a blockchain, the decisions made by the information flow planner can be made auditable as well. High fault tolerance is beneficial as well, as businesses' and customs' operations might depend on information sharing using the architecture.

However, from our analysis of the typical characteristics of blockchain technology, we can also derive that it is prone to scalability issues and issues with keeping data that is stored in the blockchain confidential. Therefore, we should make design choices that reduce these issues.

We can divide the design of the blockchain-based context information repository into four elements. First, we discuss what data is stored in different blocks in the blockchain and how they are linked in a chain. Next, we discuss the way in which the blocks are distributed in a blockchain network. Then, we discuss what the blockchain network looks like in our case and what organisations can be nodes. In the last subsection, we discuss the consensus mechanism according to which nodes accept or reject new blocks of data.

11.5.1 The data stored in the blockchain

In a blockchain, each block consists of a body and a header (Nakamoto, 2008). In the body, data, such as transactions, are stored. In the case of the context information repository, in the body of blocks, the data elements for context information (see section 10.2) and proposed information flows (see section 11.4) are stored. Furthermore, context rules (see section 10.3) should be stored in the body of blocks as well.

For this data, we should ensure that they are added to the blockchain by the appropriate parties. In section 10.2, where we present the sensors of the context-aware architecture, we already discussed that sensors, i.e., businesses, government organisations and third parties, can be identified by their public key. To make it possible to determine who added the data, they should thus encrypt the context information using their private key. In addition, the information flow planner can encrypt proposed information flows using its private key as well, to ensure that the proposed flow was not added by another

party. Furthermore, the context rules could be added by a governance body of the context-aware architecture and only be accepted if they are encrypted using their private key.

In the header of blocks, a Merkle root of the data in the body is stored (Nakamoto, 2008). The data in the body is linked to the data in the header in this way. In addition, in the header of a block, a hash of the header of the previous block is stored (Nakamoto, 2008). In this way, it is linked to the header of the previous block, which is linked to its body and a chain of blocks is created. This makes it difficult to tamper with the context information, context rules and flows of information that are stored in the blocks without detection, as this would require changing all blocks after it as well.

As the blocks are distributed to each node in the network, so is the context information and the proposed information flows contained in them. However, we expect context information to be less sensitive than, for example, the shipping information. However, if possible, the confidentiality of this data should be protected. A way to do so is by encrypting the data elements such that only the parties that actually need access to them have access and can decrypt them.

The information flow planner uses context information and access history to make decisions. This means that it should have a key to decrypt this data. Other parties do not necessarily require such a key. For the context information, an exception is made for the verification of the certificates that are used to identify parties, namely nodes need these for the consensus mechanism (see section 11.5.4). The proposed information flows do not need to be accessed by anybody, but the parties that are in that flow of information. As the context rules are the same for all parties, they do not need encryption.

If the information flow planner makes a decision, then it is in the interest of the parties in that flow of information to determine whether there has not been a party that has been provided access against their rules. These are exactly the parties that have a key to decrypt that flow of information. Furthermore, they have access to the rules that they have stored on the basis of which the decision has been made.

Data type	Encryption	Who has a key
Context information and access history (Except identity certificates)	Yes	Information flow planner
Context rules	No	-
Proposed flows	Yes	Parties in flow

Table 35: Encryption of information in the context information repository

11.5.2 Distributing blocks in the blockchain network

Storing and sharing data using the blockchain of the context repository is based on the six steps described by Nakamoto (2008) in the paper in which Bitcoin was introduced. In the case of Bitcoin, nodes can gain bitcoins by adding blocks of transactions to a chain (Nakamoto, 2008). To make this difficult, they, therefore, have to provide a proof of work first (Nakamoto, 2008).

In our case, there is no obvious advantage of requiring proof of work. Furthermore, requiring proof of work would make the architecture less scalable (Vukolić,

2016). Instead of making it difficult to add new blocks containing context information and proposed information flows, this should be easy not to harm timeliness of this information. Requiring proof of work might also be a way to prohibit parties from spamming new blocks. However, all parties using the context-aware architecture are known in our case (see section 11.5.3). A party that misbehaves thus can expect to be identified and dealt with. Therefore, we will not require a proof of work and thus skip step 3 in Nakamoto (2008) in which a node works on a proof of work.

The remaining steps to share context information and proposed information flows are the following (adapted from Nakamoto (2008)):

1. New context information and proposed information flows are broadcast to all nodes
2. Nodes collect the new information in a block
3. A node broadcasts the new block to all nodes
4. Nodes accept the block according to the consensus mechanism (see section 11.5.4)
5. Nodes express their acceptance of the block by working on creating the next block in the chain, using the hash of the accepted block as the previous hash.

In the case of Bitcoin, parties that mine a block receive bitcoins for their effort. In our case, the parties that are nodes in the blockchain network already have an incentive to add new blocks. By adding new blocks, they can ensure that their context information or the information flows that they need to share information are added to the blockchain and shared. This is necessary for them to control access to their data or benefit from information sharing. Furthermore, the costs of adding a block are much lower when proof of work is not required.

11.5.3 The network in which the blockchain is distributed

In the case of the context-aware architecture, each information router needs access to the flows of information that are proposed by the information flow planner. The proposed information flows are stored in the blocks in the blockchain and therefore distributed throughout the blockchain network. If each information router has access to a node in this network, then they can have the access to the proposed information flows that they need. In addition, the information flow planner should have access to nodes as well, to be able to obtain context information from the blockchain and to add proposed information flows. Furthermore, the sensors should have access to a node as well to add context information to the blockchain. As they will generally overlap with the other systems, they can use an information router to do so as well.

When looking from the perspective of the parties that are nodes in the network, then all parties that are involved in the information sharing process should have an information router and thus are nodes. However, this does not mean that a public network is preferable. A public network could have heightened security issues and it would be more difficult to keep the data stored on it confidential. In addition, a public network without proof of work could be susceptible to Sybil attacks (Douceur, 2002; Vukolić, 2016). Therefore, it is not desirable that any party can be a node.

The blockchain network for the context information repository is thus not public. We already included some third parties that determine who can use the context-aware architecture (see section 10.2.2). As the exact same parties should be nodes in the network, these same third parties could arrange this.

There is no reason to restrict the right to determine consensus to a single party, such as in a fully private and permissioned ledger. Having multiple parties maintain the ledger and determine consensus improves reliability. All parties in the network should thus be able to maintain the ledger.

11.5.4 The consensus mechanism

There is a variety of alternative consensus mechanisms to those relying on proof of work. In the case of the context information repository, we want only blocks to be accepted for which the information in it is added by parties that are certified by the identity managers. If the identity managers store certificates for each party that is accepted on the blockchain, including their public keys, then it is easy to verify for parties whether data was signed by a party that should be in the network using their private key.

It is in the interest of the parties using the context-aware architecture to accept only such data in their blocks. The use of the context-aware architecture should provide such a balance between benefits and risks that businesses are willing to use it. If they are willing to use it with that balance, they are unlikely to do something to disturb it. Furthermore, while they can check whether data has been signed by certain parties, they are unable to view its contents. Accepting data that was not signed by a party that is allowed to use the context-aware architecture could provide risks to illegitimate access of a businesses' own data.

12 A context-aware architecture for B2G information sharing

In this chapter, we present the overall design of the context-aware architecture that includes all components discussed in the previous sections. In addition, we will demonstrate the architecture in some scenarios.

Parts of this chapter have been published in van Engelenburg, Janssen and Klievink (n.d.).

12.1 The overall context-aware architecture for B2G information sharing

The context-aware architecture consists of the following components: adaptors (section 10.1), sensors (section 10.2), the systems of the businesses and government organisations involved in information sharing, the information systems providing additional functionality, the information flow planner (section 11.4), and the context information repository (section 11.5). We discussed the details of these components and the connections and in- and output they require from other components in detail in the previous sections. In this section, we provide an overview of the overall architecture.

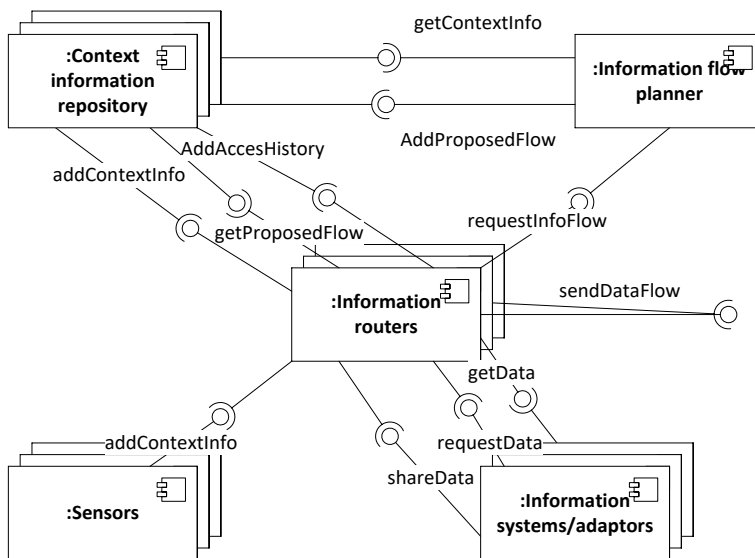


Figure 15: UML component diagram of the context-aware architecture

Figure 15 provides an overview of the components in the context-aware architecture and their connections. First, it is important to note that the sensors and information systems of businesses and government organisations often will be the same systems, as businesses and government organisations can be sensors. The reason that we added sensors as separate components is that trusted third parties can fulfil the function of a sensor as well. They usually will not be involved in the sharing of the actual data, in contrast with the

information systems of businesses, government organisations, systems that provide added functionality and the adaptors.

Each information system, adaptor and sensor has an information router. Each information router is connected to a node of the blockchain network that is used as a context information repository. The sensors add context information to the repository using their information routers. The other information systems use their information routers to share new data that they generated, to request data, or to receive data from other parties. The information routers request flows of information from the information flow planner when their information system does one of these things. The information flow planner obtains context information from the repository and derives what flow of information is appropriate in the situation. The information flow planner then adds the proposed flow to the repository. The information routers then view the proposed flow in their copy of the blockchain. They send the data to the next information router in the proposed flow and then add an access history statement to the repository.

12.2 Demonstration of the architecture

In this section, we demonstrate the use of the architecture in three scenarios. Our motivation for developing the architecture was to support B2G information sharing of additional information that customs can use for risk assessment. To illustrate the basics of the architecture, the first scenario is a simple scenario. It concerns the use of the architecture to share an ENS, which is a required document the carrier has to provide to customs.

The second scenario concerns an update to the information customs have received in the ENS. This scenario is based on an interview with a policy advisor at customs. As part of the interview conducted to evaluate the architecture, we asked them for scenarios in which additional information sharing would be useful to them (see section 13.1 for details). The third scenario is also based on the interview. This scenario concerns the sharing of an invoice.

12.2.1 Scenario 1: sharing of the Entry Summary Declaration with customs

When goods are imported into the European Union, the carrier of the goods should submit the Entry Summary Declaration (ENS) 24 hours before loading the container at the port of departure to customs (The Commission Of The European Communities, 2006b). To keep this first scenario relatively easy to understand, we will start with the ENS and not cover the information sharing that is needed to compile the ENS.

In this scenario, the carrier has hired a customs broker to act on their behalf, and generate and submit the ENS to customs. The carrier uses the architecture to add context information to the context information repository, expressing that they authorise the customs broker to submit the ENS to customs on their behalf. The corresponding literal that expresses this context information could be formulated as follows: *authorised(carrier, customs_broker, submit, ens, container_nr, customs)*.

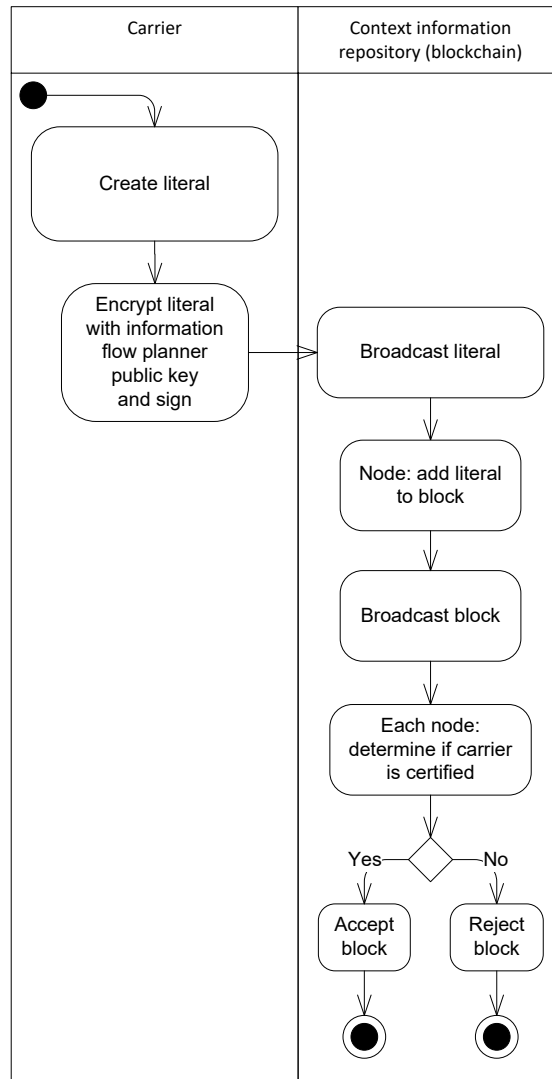


Figure 16: UML activity diagram for the adding of context information to the repository in scenario 1

Figure 16 shows the process according to which the carrier adds context information to the repository in this scenario. First, they create the literal mentioned above. Then, they encrypt it using the public key of the information flow planner, so only the information flow planner can access it and they sign it. Then, they distribute the context information in the blockchain network. One of the nodes in the network collects it and adds it to a block. The block is then distributed throughout the network. Each node in the network can then check whether the literal was actually signed by the carrier and they check whether the carrier is certified as a member of the architecture. To do so, nodes check

whether this certificate is stored on the blockchain (left out of the figure to protect intelligibility). Based on this, they either accept the blocks and add new blocks after it or reject it.

The next step is for the customs broker to share the ENS with customs. First, they create the ENS. Then they add the ENS to their information router and request a flow of information. In this case, they will include in the request that they want to share the ENS with customs. This means that the request consists of the following elements: identifier of the system of the customs broker, hashes of the data elements in the ENS, identifier of the system of customs.

Next, the information flow planner obtains context information and context rules from the blockchain to make its decision. This includes the information added by the carrier that the customs broker is authorised to submit the ENS on their behalf. The other context information that the information flow planner uses can have various sources. For example, the customs broker could claim initial control over the data elements by adding their first access history statement to the blockchain, and they could add the type of the data elements to make clear that these are the elements of an ENS.

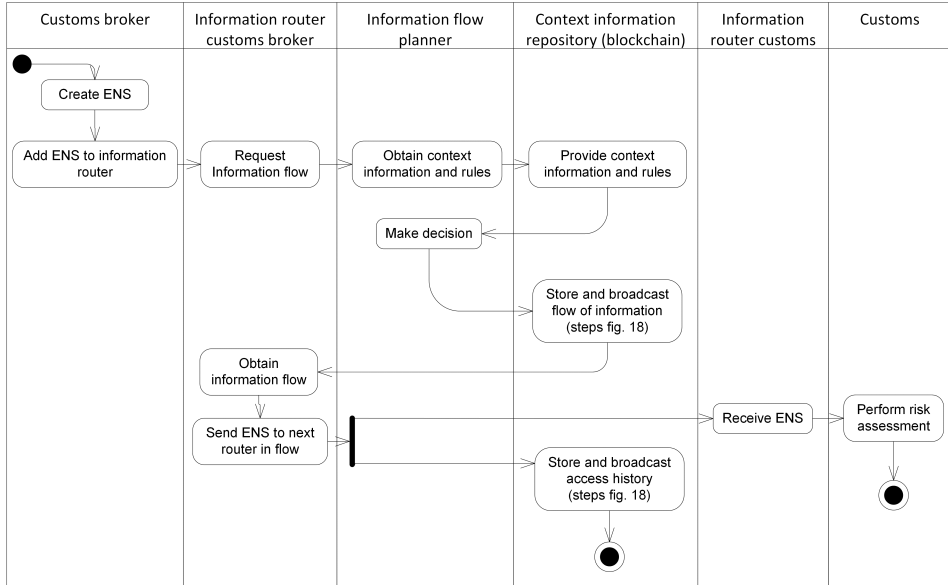


Figure 17: UML activity diagram of the customs broker sharing the ENS with customs in scenario 1

When the information flow planner has made a decision, the information flow that it proposes is added to the blockchain. This proposed information flow is then distributed throughout the blockchain network. This is done according to the same steps as for distributing context information (see figure 16). The information router of the customs broker is a node in the network as well and therefore receives this flow of information. It can automatically decrypt it and look up to which system they should send the information next. In this case, the ENS can directly be shared with customs, and the information router

of customs is thus the one identified in the flow of information. In addition, the router of the customs broker adds to the blockchain that customs has received access to the data elements. The information router of customs receives the information and customs can use it for risk assessment.

12.2.2 Scenario 2: updating the ENS

As stated in the previous section, businesses are required by legislation to share the information in the ENS at least 24 hours before they start loading the container in a foreign port (The Commission Of The European Communities, 2006b). It is possible that in those 24 hours things change and that this information becomes outdated. When the goods arrive in the European Union, the information in the ENS might be outdated. In that case, the information that customs base their risk assessment on is outdated as well. The possibility mentioned by the interviewee at customs is that a container's itinerary has changed and it might end up on a different ship or even be loaded in a different port. However, customs at the port of entry in the EU might have targeted this container for inspection. Yet, they end up looking for a container that is not there, which results in inefficiencies and delays.

Several parties could know that the transport plans for this container have changed. One of them is the freight forwarder who is responsible for arranging the transport. The freight forwarder might want to let their customer know about the new route of the container. Furthermore, they might want to inform customs at the port of entry and at the port of discharge that the schedule has changed.

For this scenario, consider the goods to be of high value and thus at risk of being stolen. Therefore, the freight forwarder might not want other parties to have access to the location of the container, as well as their contents. They formulate an access control rule that states that for this container, others cannot have access to its route as well as a description of its content, except for customs and the customer for which they transport the goods. They add this access control rule to the context information repository according to the process in figure 16. The freight forwarder then shares the new route of the container with both customs organisations via the process in figure 17. They can use the same process to share with their customer as well.

12.2.3 Scenario 3: sharing an invoice

In the third scenario, the invoice of a shipment is shared with customs. According to the interviewee at customs, for certain types of shipments, it is very common to share invoices with customs, but it is uncommon in the case of container shipping. Such an invoice would be useful for customs to crosscheck with other data they have on a shipment to determine the risk of fraud, *inter alia*. Several data elements in the invoice can be useful for this, but in this scenario, we focus on the price of the goods.

In this scenario, information on the prices of goods is not public in the relevant market. In fact, this information is highly competitively sensitive in this market. This means that it is not lawful for a competitor to have the opportunity to access this

information. In addition, the seller is not willing to share with their competitors either for this reason.

The seller does need to share the invoice with their buyer. Furthermore, they want to share it with the freight forwarder, as they need it to generate an import declaration. However, other parties might be interested in some of the information in the invoice as well. For example, it contains the weight of the goods, which might be important for the carrier to know at an early stage so they can include it in the calculations they need to perform to balance the ship and include it in the gross weight mentioned on the ENS.

As the information on the ENS needs to be shared with various parties in the same supply chain, the seller might want to use the functionality of the data pipeline to connect the invoice to the other information on the shipment and let supply chain members know that the information is available. To protect their data, they add access control rules that state that except for the buyer and the freight forwarder, no other parties can have access to information on the price in combination with any data element that could identify the shipment. They do so using the procedure as presented in figure 16.

The data pipeline that the seller wants to use to share the invoice is used by members of the supply chain of the shipment, but also by competitors. The data pipeline broadcasts information via a publish/subscribe mechanism. The seller now can use their information router to share the invoice according to the procedure as presented in figure 17. However, as a competitor could potentially have access to the invoice when it is stored in the data pipeline without any additional measures, the information flow planner proposes an information flow in which the information flow is made thin first and then only metadata and a link to the invoice is shared via the data pipeline. This avoids making the information flow unlawful according to the context relationship in section 9.2.3.5 and it conforms to the access control rule of the seller.

The data pipeline then receives the information in its information router. It then performs its functionality and it creates an event stating that the invoice is available. Furthermore, it connects this event to the other information on the same shipment. In addition, it broadcasts the event to all parties that are subscribed to information on the shipment (using the same procedure as for sharing other data). The buyer and customs now can request an information flow from the system that stores the invoice, according to the link they have received. Both of them will receive the full invoice, including prices as this is allowed according to the access control rules of the seller and the context rules. However, if a competitor does the same, they will not receive such information, as this does not conform to the context rules and to the access control rule of the seller. The carrier cannot receive pricing information either, according to the rules of the seller. However, if they just request the weight of the goods as stated on the invoice, this could be shared with them, as this information is not sensitive according to any rules.

PART IV: EVALUATION, CONCLUSION AND FUTURE RESEARCH

13 Evaluation of the architecture

In this section, we discuss the evaluation of the architecture and how we applied the methods described in section 8.6. First, we describe how we conducted the workshops and interviews. Then, we describe how we analysed the resulting data. In the next sections, we describe the results concerning the following questions formulated in section 8.6:

1. Is information sharing using the architecture lawful?
2. Are businesses willing to use the architecture?
3. Is the architecture useful to customs for compliance monitoring?

The validation of the context model is discussed in section 14.2.1.

As we view design as a cyclical process, the results of the evaluation should provide new input for the next cycle in the design process. We thus end this section with some conclusions on what changes should be made to the architecture in the next cycle and on what things require additional attention.

13.1 Interviews and workshops

For the evaluation, we have two main sources of data, namely, a workshop and expert interviews. The workshop was conducted at Maersk Line. It was attended by senior staff members with expertise in IT and innovation and senior staff members with juridical expertise. Furthermore, academics with juridical expertise, as well as expertise in ICT and in information sharing in international supply chains attended these workshops. The interviews were conducted with a senior policy advisor and expert in formal law at Dutch customs, an academic expert in trade law and an expert in IT and governance from business as well as academia.

During the final workshop at Maersk line and during the interviews we presented the architecture using a PowerPoint presentation. In this presentation, first the notion of context elements and context relationships were discussed and some examples were provided to ensure for a shared understanding of the notion of context and context-awareness. Then, each of the components of the architecture was explained. At the end of the presentation, an animation was shown of a scenario in which the architecture was used first to share packing list with a carrier and then to share the ENS with customs using the context-aware architecture.

Interviewees and workshop participants were then invited to provide feedback on the architecture during and after the presentation. The choice was made not to focus on obtaining a straightforward ‘yes or no’ answer to the question of whether businesses are willing to share information using the architecture. Participants and interviewees would not have been able to provide such a definitive answer without consulting others within their business or customs organisation, for example, and might be afraid that answering might commit them to participate in the architecture. In addition, answering this question might not even be possible for parties without knowing what the architecture would look like when implemented. Furthermore, such an answer would not necessarily provide insight into why businesses are or are not willing to use the architecture. A ‘yes’, or a ‘no’ could very well have to do with specific circumstances of the businesses that are

not related to the architecture. Therefore, we focussed on asking what participants and interviewees regarded the main advantages and disadvantages of the architecture.

Nullum crimen sine lege (i.e., no crime without law) is a fundamental legal principle (Dana, 2009). In our case, it means that we can expect information sharing using the architecture to be lawful unless there is a law according to which it is not lawful. Therefore, we asked whether the interviewees with a juridical background could assess whether there are any obvious juridical issues when information is shared using the architecture.

The interviews and the discussion during the workshops were semi-structured. Semi-structured interviews allow for ensuring that all questions are asked, but at the same time allows for improvisation and exploration (Runeson & Höst, 2009). This was suitable in our case as well, as we wanted to ensure an answer to our basic questions. At the same time, we wanted to provide an opportunity for participants to bring up new advantages and disadvantages.

The interviews and final workshop at Maersk Line were recorded and transcribed. This allowed us to have a full description of what was said as input for our analysis, which allowed us to determine what is important afterwards (Runeson & Höst, 2009; Walsham, 1995a). As recording interviewees might inhibit them, we promised to not share the recording with others (Darke et al., 1998; Walsham, 1995a). The interviewer checked several times whether she really understood the interviewees or workshop participants by summarising to them what they said.

Question 1 was directly asked to participants and therefore could be summarised from the transcripts directly. However, for the advantages and disadvantages, we wanted to establish what themes emerged from the answers of the participants. Therefore, we performed thematic analysis, similar to that described by Burnard (1991) and Vaismoradi et al. (2013). For the analysis of the transcript, we went through it and determined what subjects were discussed in different parts of the interviews and workshops. Each time a new subject was discussed, we put a new header above that part of the transcript. If a subject was not relevant to the evaluation (e.g., a discussion of the time left or a basic explanation of the architecture) the reasons for leaving it out of the evaluation were made explicit. As each of the interviews and workshops had a very specific role, there was limited overlap between the themes. The next step was to summarise what was said for each of the subjects and thereby describing the advantages and disadvantages mentioned in the data.

13.2 The lawfulness of information sharing using the architecture

The lawfulness of information sharing using the architecture can be evaluated at two levels. The first is the lawfulness of the information flows according to which information is shared in the architecture. The lawfulness of these information flows cannot be determined at the level of the architecture, as the architecture adapts the flow of information. Instead, the model of context that determines what the flow of information should look like in different situations should be validated. We do so in section 14.2.1.

The lawfulness of information sharing using the architecture can also be viewed at the level of the overall architecture. The sharing of context information and the decision making of the information flow planner, for example, should be lawful as well. This is what we are concerned with in this section.

The architecture was presented to several juridical experts in the workshop, including experts involved in the project of which this research was a part. While participants of the workshop were not explicitly asked whether they thought that information sharing using the architecture was lawful, they did not bring it up as a concern.

The juridical expert from customs that we interviewed did not raise such concerns either. Furthermore, the expert in trade law stated that they did not see any obvious juridical issue in the scenario presented to them. They could also not think of any scenarios in which the architecture could be used that would be unlawful.

The results do not guarantee that in all possible situations information sharing using the architecture is lawful. However, we could not identify any juridical issues that would make it unlawful. This indicates that in general, the architecture provides for lawful information sharing.

13.3 The willingness of businesses to use the architecture

In this section, we aim to determine to what extent businesses are willing to use the architecture. Several important subjects arose from the discussions during the workshop and the interviews, viz. the usefulness of the architecture to businesses, access control, the use of blockchain technology, the use of sensors, context information and context rules and the feasibility of the architecture. In this section, we summarise for each of these what was stated during the interviews and workshops. At the end of each subsection, we reflect on what these findings entail for the design of the architecture and the direction of future research.

13.3.1 Usefulness of the architecture to businesses

One of the factors that will most likely influence whether businesses are willing to use the architecture is whether it would be useful for them to do so. Unsurprisingly, this was therefore a subject addressed in the interviews and workshops.

Findings

When asked about the usefulness of the architecture in general, the interviewee with the background in IT and governance replied that the architecture is a ‘perfect fit’ for industrial organisations for supply chain management and supply chain networks. He states the following: *“So, they will definitely see directly the advantages of this kind of architecture. And they agree that their company requires such an architecture. So, it is not only the architecture within your own company, within your own IT/IS environment, instead, you need an overarching network architecture.”*

The interviewee with expertise in IT and governance viewed as one of the main advantages of the architecture that the lead-time to exchange information between

multiple organisations would decrease significantly. He stated the following about this: *“So, maybe in a classical type of organisation with separate systems and without this overarching architecture and integration, you will see that, for instance, your lead time is one day, and that could be reduced approximately to one hour 30 minutes or even less.”* The interviewee expected this because in the architecture businesses are directly connected with all member firms or sometimes competitors within the domain. In the existing situation, according to the interviewee, businesses all have separate systems and they have to agree and disagree about every transaction and exchange of information, and that costs a tremendous amount of time.

According to the interviewee, this will cause cost savings. He states the following: *“Companies who are interested will translate the key advantages to cost reductions and improve the quality of exchange of information. Because what I do recognise in existing separate systems, each and every one, a company has to implement their own quality system, risk system. So, when you are able to agree on a kind of overarching architecture and exchange the information, that will decrease the degree of risk, quality issues, operational issues and that kind of stuff. That will decrease cost level significantly.”* The interviewee does warn that this advantage of the architecture could be a trade-off with a tremendous amount of time that could be needed to design, build, develop and implement the architecture (see section 13.3.5).

Reflection

The usefulness of the architecture to businesses is highly important. If businesses believe that the architecture is useful to them, then they will be willing to use it. If businesses use the architecture to share information, then customs might piggyback on this information flow.

We agree with the observation that in the architecture there is a trade-off between the timeliness of data sharing at run time and the efforts to implement the architecture. This seems because some of the complexities of information sharing, such as determining what to share and with whom, are being transferred from the time when the data is shared, to the time when the architecture is designed and implemented. For example, instead of coming to an agreement on what data to share each time data is shared, agreements need to be made on how the information flow planner will need to be implemented.

13.3.2 Access control

A subject that was mentioned often is that of access control. In the architecture, businesses can control access to their data by providing context information, including business rules, for specifying who can access their data and under what circumstances. The information flow planner then uses this to determine the flow according to which information should be shared. In this way, businesses can control who accesses data, even when it is shared from other systems, for example, without having to make decisions on an individual basis for each data element, party and situation in which sharing takes place.

As an advantage, it was mentioned that the architecture provides a lot of control over data. In addition, there is no central party, such as the carrier, solely responsible for

access control, including its ‘hassles’. At the same, these ‘hassles’ being put on other parties were stated to be a disadvantage of the architecture.

13.3.2.1 Who controls access to the data

An important concern was who ‘controls’ or ‘owns’ data. Currently, in the architecture, the party that first adds a hash of a data element to the blockchain can control it (see sections 10.2.1.3 and 11.4). To calculate the hash a business needs to have the data element. By adding the hash to the blockchain, they provide proof that they control the data. Furthermore, the agreement is not to share data outside of the architecture. Therefore, if a business added the hash of a data element first, they likely did not receive it from other parties. In addition, the rules of all other parties for which their system are involved in the information flow should be taken into account. It seems reasonable that the system of a party only should be used to transfer the data if they are willing to do so.

During an interview performed for deriving the context relationships (see section 9.1.1.3.1), with an expert in IP law explained that there is no such thing as the ‘owner’ of data from his perspective. Instead, there are only parties that have control over data and that can, therefore, determine who data is shared with and under which conditions. The way in which it is determined who can control access to data is based on this idea.

Findings

The idea that there are only parties that control data instead of owners of data, interestingly, does not correspond with the view from the businesses’ perspective. The results of the evaluation suggest that what party should control data access depends on their role in the shipment that the data is about and on who is the owner of the shipment, for example. Who should control access to data seems an important issue, especially during the workshop. However, there seems to be no consensus on this.

According to participants, it might be necessary for parties to transfer power over the data, for example to a person that acquires the goods or to parties that have a certain role in information sharing. In addition, they might want to delegate this power. A suggestion provided was to base access control on the Incoterms under which goods are shipped.

Reflection

The idea that the owner of a shipment should control access to its data and that what party should control access to data depends on their role in the supply chain, as suggested, is not likely to work in all circumstances. For example, if the manufacturer of goods creates them according to a procedure that is a trade secret, then it is not up to the owner of the goods to control access to this trade secret. It should be further investigated in what circumstances access control should and should not be arranged in the way suggested.

The current design of the architecture does not provide for the delegation of access control. However, parties can use the architecture to establish who can submit documents to customs on their behalf by adding agreements to the blockchain (see section 10.2.1.1). This same mechanism of authorising others could be used to authorise others

to control access to data. This would simply mean that a party who controls the data adds an agreement authorising another party to control data as well. The information flow planner could read this and take the business rules of that business into account as well.

13.3.2.2 Prohibit mistakes

One concern discussed by participants was that the owner of the data would make a mistake and provide the wrong parties with access to data.

Findings

The participants of the workshop suggested adding rules to the architecture to detect contradictions and recognise mistakes. To illustrate, one of the participants stated the following: *“And then even when the transport owner makes a mistake, the system will not execute that order, because it is basically self-contradictory and the system recognises that this cannot be. It cannot be that we have let's say, the container being carried by Maersk and MSC at the same time, so it has to be the one or the other.”* The participant also suggested capturing in the rules that one carrier could license another to carry goods in a consortium agreement.

Reflection

The rules discussed by the participants seem to be used to check the quality of the context information and, for example, its consistency. There seems no clear reason for why the information flow planner could not consider such rules as well. However, it could be challenging to formulate these rules in a way such that they can help to detect mistakes, while at the same time not ‘fire’ unnecessarily and hamper information sharing. Furthermore, it needs to be determined what should happen when mistakes are detected.

13.3.2.3 Power dynamics between businesses

An additional concern that was brought up from the businesses’ perspective concerns the power balance between businesses.

Findings

The expert in IT and governance stated that he believes that very large companies will dictate the market. In this way, they can dictate what rules smaller companies have to comply with to control their data.

Reflection

When information is shared (or not shared) in the current situation this might be dictated by the larger parties as well. However, in the architecture, the access control rules are made explicit. This means that this dictating of how information should be shared by larger parties also might become more apparent. In addition, they could apply control that is more direct by formulating the exact rules others should use. It would be interesting to

determine in practice whether this is something that actually will happen and how to deal with it.

13.3.3 The use of blockchain technology

The participants of the workshop and interviewees did not provide many comments on the use of blockchain technology in the architecture. There could be several reasons for this, including them not completely understanding the technology. However, one participant did provide extensive comments on the use of blockchain technology.

Findings

One of the participants of the workshop stated that they think that blockchain technology was used just like a database in this case and that blockchain was not created for that purpose. The participant also disagreed that the use of blockchain technology would make it harder to tamper with the data stored than would be the case in a central solution. They stated the following: *“But it doesn't make it harder to tamper with, because you need to sign the transaction whether you put it on a database or on a blockchain and the fact that you signed the transaction means that you cannot tamper with it without changing the signature. So, you can't do that.”* In addition, the participant added the following: *“You are saying you don't trust the guy in the middle to handle the database. But yet you trust the flow.”*

The participant also argued that you need a permissioned network and that it is difficult and costly to set this up and for businesses to enter the architecture. In the end, the participant concluded that although using blockchain could lead to more security, it is the question whether this is worth the cost.

Reflection

Before we start with reflecting on the observations of the participant in the workshop, we provide our own analysis of the extent to which blockchain technology in our case contributes to reaching the objectives of the architecture and avoiding issues with scalability and keeping context information confidential.

The use of distributed ledger technology in the architecture allows businesses to store context information and rules for controlling access to their data in a blockchain, which makes them difficult to tamper with. Decisions on what the information flow should look like are made by a centralised information flow planner. Its decisions for a flow of information that are used to share information are stored on the blockchain as well, making access control auditable. This avoids the need for businesses to put full trust in a central party. However, it is important to note that this cannot be fully avoided either. Parties should still trust the governance body of the architecture and the software provider to a certain extent to admit only the appropriate parties in the network and to provide reliable software.

Scalability and data confidentiality are typical problems with blockchain technology. A major culprit of scalability issues is proof of work, which we did not use in our case (Vukolić, 2016). Furthermore, the network is not public, avoiding Sybil

attacks and reducing the number of nodes. The volume of data that nodes need to store can be reduced by allowing nodes to only store block headers after the data in their body is not needed anymore, as proposed by Nakamoto (2008). While scalability issues cannot be avoided completely, they are reduced by taking these measures.

To protect the confidentiality of data, the actual shipping information is not stored in the blockchain. Only context information, access control rules and information flows are stored in our blockchain, as we expected these to be less sensitive. Furthermore, this information is protected by encryption. The participants in the evaluation did not raise the confidentiality of the data stored in the blockchain in our architecture as a concern.

We believe that the participant in the workshop makes several valid points about our use of blockchain technology. First, we agree that several other solutions exist that make it also difficult to tamper with data, including the one mentioned by the participant. The main difference between storing data in a blockchain and these other solutions is that in the case of blockchain technology you do need to put less trust in an intermediary.

We made the assumption in section 11.2 that it might be highly difficult to agree upon such a party. However, as parties still need to put at least some trust in the governance body of the architecture and the software developer, this issue might be unavoidable in the end. In further research, it should be investigated in more detail whether the difficulty to establish a central party that arranges access control weights against the added complexity of developing and using an architecture based on blockchain technology.

We also agree with the participant that it might be difficult to determine who should and should not be in the blockchain network. However, this has more to do with the scale of the architecture than the use of blockchain technology, as the users of the architecture and the blockchain network are the same.

13.3.4 Sensors, Context information and context rules

The sensors, context information and context rules in the architecture were also a subject of discussion during the evaluation.

13.3.4.1 Identity management

An important part of the context information is the identities of parties in the architecture. Only parties that are certified can participate in the architecture. Decisions on this are made by identity managers (see section 10.2.2.1). They then add the certificate to the blockchain so parties can check other's certification and obtain basic information on them.

Findings

An issue raised during the evaluation was that the different systems might assign different IDs to different parties and that all of these might need to be linked. A potential solution to this that was suggested was to use the different initiatives for developing national IDs, such as eHerkenning in the Netherlands, and to map the internal IDs of the systems to the national IDs for the businesses.

Reflection

The solution offered by the participants in the evaluation is very suitable for the architecture. The initiatives they mention could be incorporated in the architecture in their existing role of identity manager.

13.3.4.2 Trusted parties

In the architecture, trusted parties are appointed per market as sensors to provide context information (see section 10.2.2.2).

Findings

One of the participants in the workshop was asked what precisely is a market. Furthermore, they stated the following *“The reason why I ask is because we have a global infrastructure, right? And there is millions of markets, there is several thousands of markets and then how you want to cope with those kinds of things?”* According to this participant, this would require assigning parties from the countries and segments in the countries that should have a full understanding of the market situation, which they considered impossible.

Reflection

The main reason for appointing a trusted party is to have an independent party that has insight into the market determine what the situation in the market is. However, there are different other possible sources for this information. The most important ones are of course the businesses in the market. They could provide information on what data is competitively sensitive (and they, in fact, do so for other purposes). It might be different per market how easy it is to establish what data is competitively sensitive from the point of view of competition law. When this is easy, then businesses can just provide the required information. When this is complex, however, parties can ask others to investigate and provide them with the required information. In that way, experts are only involved when necessary.

13.3.4.3 Customs as sensor

In the context-aware architecture, customs provides information on the obligation parties have to share data with them (see section 10.2.3). The interviewee with the customs background suggested that adding such rules could be similar to just asking for information in a letter and that from a juridical perspective there are no clear obstacles for doing this. However, they could not provide a definitive answer to whether customs is willing to participate without further analysis.

13.3.4.4 Incoterms as context information

The role of Incoterms was discussed during the evaluation as well.

Findings

During the workshop, it was suggested by a participant that in many cases businesses should have access to data if they are handling the goods in some way. This is similar to one of the context relationships that were found in this research. However, the participant suggested that it could be determined who is involved with handling the goods by looking at the Incoterms under which the goods are shipped. We agree that this can be a source of information for deriving the roles parties have.

Reflection

Agreements on the Incoterms could be one of the agreements that parties can add as context information on who is involved with a shipment and in what role (see section 10.2.1.1). This might be a useful addition, as Incoterms have defined semantics and need to be determined by parties anyway.

13.3.5 Feasibility

A variety of subjects was discussed considering the feasibility of the architecture and the way in which it should be implemented and used in practice. While we only present the architecture at a conceptual level, it still is only useful if it is feasible. In fact, determining how the architecture should be implemented in practice is an important next step. Therefore, discussing the comments considering the feasibility of the architecture is important.

13.3.5.1 Large scale implementation of the architecture

A concern mentioned during the evaluation is the large scale at which the architecture should be implemented.

Findings

The context-aware architecture has a large scope. It overarches the different solutions for information sharing, such as data pipelines and single windows. One of the participants in the workshop suggested that therefore it is important that in practice it can be build layer by layer, starting very simple and thin.

In a similar vein, according to the interviewee with expertise in IT and governance, the implementation of the architecture will cost a lot of time. The interviewee stated the following: *“If you take a company like Maersk and customs and suppliers and etcetera, my assumption is that it will cost you over a year to agree on such an implementation and design it. That means that all companies in the client's ecosystem needs to be involved because they all have their rules and regulations about information flows, thick flows, thin flows, all that kind of stuff. So, it will cost you certainly a lot of time taking the multitude of firms into account. Maybe the whole development about building it, agreeing on the implementation will take one year or one and a half year for the first time. And, of course, if you learn from this ecosystem type of networks, well, the next one can be developed, implemented faster. But I think for the first years to come, I expect a lot of effort in the development phase, and agreement phase.”*

In addition, according to the expert in IT and governance, there is still some work to be done on a practical level before the architecture can actually be implemented. Businesses might have to adapt their systems from a technical perspective to use the architecture to some degree. As other examples, the interviewee mentioned that new API's and new protocols would need to be designed.

Reflection

We agree that the context-aware architecture is complex and will not be easy to implement at a large scale in practice. At first sight, it does not seem like a layered solution that starts out simple when first implemented. However, as one of the other participants suggested, we might not need to start from nothing and instead reuse existing systems that are already there. For example, for identity management, the context-aware architecture could build upon initiatives in various European countries to have national identification for businesses. It could be useful to look at other 'existing layers' that the context-aware architecture could be built upon.

The costs in this case also seem to be due to the scope of the architecture and the number of parties involved. We agree with the interviewee that getting parties to agree with the way in which the architecture will be implemented and its governance will be a highly difficult task. However, parties can define their own rules for information sharing. This means that it is not required for all parties to agree on what information they have to share or cannot share when they start using the architecture. This does not need to be the same for all these parties, as they can specify it in the architecture using access control rules.

Detailing the implementation of the architecture in practice was out of scope for this version of the architecture. For this version, we were mainly concerned with the design at a conceptual level. However, we fully agree that this is something that will need to be addressed before the architecture can be used in practice as well.

13.3.5.2 Governance of the architecture

The governance of the architecture was not explicitly discussed in the current version of the architecture. It is merely suggested that a party that governs the architecture exists. However, it was mentioned during the evaluation.

Findings

According to the expert in IT and governance, governance will be a key topic especially concerning who is responsible for doing what. However, he also states that governance is still in a development phase for the type of environment of the research. This also means that it will cost a lot of time to agree on different aspects.

The interviewee also addressed the question of who will be the owner of the architecture. The interviewee stated the following about this: *"Because the whole idea is based on a coherent approach you have to do it together as members, partners of an ecosystem. That means that all parties, all members will be accountable and responsible for the architecture as a whole. So, that is an interesting governance question. How do*

you deal with these kinds of topics? And for sure, based on technology, it will work in the end. There will be mechanisms, there will technicians who say I will have a solution to deal with it. But the design, the governance rules and then willingness or unwillingness of parties to get involved and also political reasons within the companies will affect the outcome. Saying, oh, I do not want to be in the same network as competitors, all those kind of discussions that will form a barrier to design and implement such a network and design the architecture.”

Reflection

Who is the owner of the architecture is something that we have not yet taken into account. However, we agree that it is an important issue. Furthermore, the governance model suggested by the interviewee, namely one in which all parties together are accountable, seems to fit best with the idea that parties should be able to control their data.

The last remark by the interviewee is especially interesting. Businesses not wanting to share sensitive data with competitors or competitors not wanting to use the same information sharing system were a concern when designing the architecture. The architecture contains context rules to deal with this. However, the interviewee seems to suggest that a similar issue might happen at another level as well, namely that competitors might not be willing to use the same overarching architecture.

13.3.5.3 Standardisation

One of the interviewees also brought up the subject of standardisation on the process level and the context level.

Findings

The interviewee stated the following: *“If you have a container, a data container that says that you want to transport 100 flowers from the Netherlands to China and that specific data is stored in a data container and transported via this kind of network, what I already notice in practice from my experience in the field of robotisation, artificial intelligence, is that it will be crucial. Because every organisation will have their own set of data architecture, data modelling types. They don’t match. So, also on the deeper content level, you will see conflicts between the different systems. Because they all have their own type of data architecture that is more deep on the content level.”*

For the access control rules, it was also a concern that they might need to be standardised. The interviewee stated the following about this: *“But I think if each factor or each actor in the network will just decide their own business process and their own flow and business ruling, of course, it is a system, but I think the system won’t work anymore. Because how would you deal with that?”* The interviewee suggested that standards could provide a solution.

Reflection

To a certain extent, we have already provided a standard for the context rules by formulating these rules using a logic-programming paradigm. However, it should be

determined whether this is acceptable as a standard to parties. The interviewee suggests that it will be more complex and that more standardisation is required.

Considering standardisation at the content level, formats that do not match do not have to do with the architecture itself. The architecture should support the sharing of a high variety of data elements between a high variety of parties. This is what causes the need for standardisation at the content level, however, this is not because of anything in the design of the architecture.

13.4 The usefulness of the architecture to customs for compliance monitoring

The architecture should be useful to customs. After all, the motivation for this research is to support B2G information sharing in which customs reuses data from businesses for compliance monitoring. While the interviewee at customs made clear that they could not provide a definitive answer to whether customs would be willing to use the architecture without consulting others in their organisation, they did provide some insight into the extent the architecture could be useful to them and their possible issues with the architecture.

13.4.1 Advantages of the architecture

The interviewee mentioned several advantages of the architecture.

Findings

When asked about whether the interviewee from customs thought that the architecture would be useful to customs in general, at least considering the principles it was built upon, the interviewee stated the following²: *“I think the principles behind it in any case, because potentially you can obtain the relevant information earlier. You can contact several parties, where you can obtain information faster from not only the declarant, or the importer or exporter, but maybe also via the carrier and such. Potentially, it offers more information for the purpose of analysis. And, potentially, it offers an improvement of the level of correctness of the data and such. In that sense, I think it could certainly yield an improvement.”* The interviewee then indicated that they were not willing to make stronger statements than that about this.

² The quote was translated from the following Dutch text in the transcript: *“Ik denk de principes daarachter sowieso, omdat je in potentie eerder de beschikking kan krijgen over relevante informatie. Je kan in contact komen met meerdere partijen waarbij je dus sneller informatie kan krijgen van niet alleen de aangever dan wel de importeur of de exporteur, maar misschien wel via de carrier en dergelijken. Dus in potentie biedt het meer informatie ten behoeve van analyse doeleinden. En, in potentie biedt het een verbetering van het niveau van de gegevens qua juistheid en dergelijken. En, in die zin denk ik dat het zeker een verbetering kan opleveren.”*

The interviewee stated that establishing the correctness of data remains difficult. However, as an additional advantage, they mentioned that in the architecture, the route of documents has better visibility and therefore you know that there was no tampering in that route. Furthermore, the interviewee indicated that the way of working in the architecture could enhance trust with the between parties, which would be an advantage as well. Parties can sometimes say that they do not want to provide all data to customs and they can see precisely how everything goes. Customs has a right to the information that they can view, according to the interviewee. Furthermore, the business rules are transparent while at the same time businesses know that they cannot just commit fraud.

Reflection

The interviewee from Dutch customs mentions the same main advantage of the architecture as mentioned by the other interviewees. Namely, that information sharing using the context-aware architecture is likely to be timelier. Furthermore, they state that the architecture supports obtaining information from more parties and of better quality. This research was motivated by providing customs with additional information. From the statements of the interviewee, it seems that we reached this goal, at least potentially. The interviewee indicated earlier in the interview that they want to be careful and not make definitive statements about customs' willingness to use the architecture, without consulting others within customs. This could explain the comment by the interviewee that they are not willing to make stronger statements than this.

The other advantages mentioned by the interviewee are also important for establishing that we reached our objectives. However, the route of the information is only visible, if customs is allowed access to that information. This means that this advantage only plays a role if that happens. The remarks that the interviewee makes about the building of trust, seems to conform with our idea that businesses will be more willing to share if they can control their data and avoid risks.

13.4.2 Potential issues of the architecture

In addition to advantages, the interviewee from customs mentioned some disadvantages of the architecture as well.

Findings

When the interviewee at customs was asked about whether storing an access history that includes the data accessed by customs could be an issue, they stated that they could not provide a definitive answer to this question. However, they did state that they currently send a letter to a business when they want to perform a check to inform them about this. In such cases, the business already knows. However, the interviewee stated that on the other hand, depending on the nature of the investigation, it could be better not to record such things. The interviewee agreed with the suggestion that it would help to build in a 'way out' in which no access history is added for customs that they could use in special circumstances.

The interviewee also indicated that it could be an issue if, for instance, only three businesses participate. The interviewee agreed with the suggestion that the architecture only works when the other information sharing initiatives, such as the Global Trade Digitisation (GTD), are part of the architecture.

Reflection

One of our concerns was that it could be an issue to customs if in the access history there is a record of all the data they accessed. The reason for this concern was that this was suggested in an interview with another interviewee at customs that was performed as part of the design process of the architecture. In this previous interview, the interviewee raised the concern that if parties knew what information customs had requested, they could know whether they were under investigation, for example. This could influence the investigation process. To deal with this, the architecture was designed such that the access history and proposed information flows are encrypted and can only be decrypted by the information flow planners and the parties that are part of the information flow.

From the interview, it remains unclear whether we sufficiently solved this issue. However, it seems that making an exception, in which access history is not stored for customs, could be useful. Further investigation could focus on whether such a solution is possible from a technical point of view. Furthermore, the ‘special circumstances’ in which customs should be able to exclude their access to information from the record should be determined.

The last potential issue mentioned by the interviewee is a difficult one. For the architecture to be useful, enough parties need to participate in it. This means that we need to solve the dilemma of first growth (Hanseth & Lyytinen, 2004). Further research should focus on how to deal with this challenge in the case of the context-aware architecture.

13.5 Discussion and Input for the next design cycle

Overall, it seems that the context-aware architecture potentially is useful to customs as well as businesses. Furthermore, we can conclude that information sharing using the architecture is likely to be lawful. This means that the architecture thus seems to meet its goals (see section 7.5.1).

This does not mean, however, that the architecture could not be improved in a next design cycle in future research. We can use the results of this evaluation to set new requirements for an additional design cycle and recommending changes to the architecture.

One of the changes to the architecture that could be looked into in the next cycle is to provide additional functionality such that additional parties can be made ‘owner’ of data. Furthermore, functionality could be added that allows parties to transfer ownership of data and delegate access control. In addition, functionality could be added to the information flow planner for determining the quality of context information in order to prohibit mistakes that lead to the inappropriate parties having access to data. Moreover, it could be relevant to determine what the effect is of the power dynamics between parties on the access control rules that businesses will use in the architecture, and vice versa. This is relevant, as whether businesses can control their data can affect whether they are willing

to use the architecture. In addition, the design of the architecture might need to be changed to make it possible for customs to avoid recording their access to data for certain types of investigations.

The use of blockchain technology as a context information repository in the context-aware architecture should be reassessed. At the moment, there seems to be some arguments for and against its use, without one side or the other being clearly better. This is also due to the technology being quite new. Therefore, it is still being developed further and knowledge on the effect of using it in practice is limited.

Many of the challenges identified in the evaluation have to do with the large scale of the architecture. How to deal with implementing an architecture at such a large scale is therefore important to address in the next cycle of the design process. First, the practical challenges with having a trusted party per market for many markets should be further investigated and solved. Secondly, it should be determined how the architecture builds upon and reuse existing solutions concerning identity management.

The way in which the architecture should be implemented in practice was largely left out of scope for the current version of the architecture. However, as implementation could be particularly costly and difficult for this architecture, this is also something that needs to receive attention next. The same is the case for the governance of the architecture. As no governance model exists for types of architectures like the context-aware architecture that can simply be applied, such a model needs to be developed.

14 Evaluation of the method

In this section, we present the evaluation of the method for designing context-aware systems in complex environments. We start by discussing the case study design that we used for evaluating the method. Next, we discuss for each of the cases the results. We end this section with a discussion of what should be input for the next design cycle for the method based on the evaluation.

14.1 Case study design

In this section, we discuss the case study design we used for evaluating the method.

14.1.1 Type of case study and proposition

A case study can have a single-case design or a multiple-case design. A case study has a multiple-case design if it involves the study of multiple cases (Yin, 1994). These studies use a replication logic and they are often considered to be more robust (Yin, 1994). Multiple-case designs are suitable for theory testing (Benbasat et al., 1987).

Typically, case study research involves formulating and testing of hypotheses or theoretical propositions (Verschuren & Hartog, 2005; Yin, 1994). In our case, we have formulated a proposition based on the specification of the problem we would like to solve (see section 3.1).

Proposition:

The proposed method contributes to an efficient and effective design process when designing context-aware systems in complex environments.

14.1.2 Unit of analysis and cases

The unit of analysis in our case study is the design process for designing a context-aware system in a complex environment. This unit of analysis can directly be derived from the proposition.

Generalisation in case study research is not a statistical inference from samples, but an analytical induction from cases (Wieringa, 2014). This is explained by Yin (1994, p. 10) as follows: *“the case study, like the experiment, does not represent a “sample,” and the investigator’s goal is to expand and generalize theories (analytic generalization) and not to enumerate frequencies (statistical generalization).”* The selection of a case can be guided by the hopes of generalisation of the researcher.

In this research, we applied the method to design a context-aware architecture for B2G information sharing in international container shipping. The evaluation of that architecture and the context model that it incorporates, in particular, can contribute to the evaluation of the method. Namely, the quality of a design artefact provides information on the quality of the design process (Hevner et al., 2004).

However, using this as the only case on which the evaluation is based, would introduce various biases. Namely, it is developed and applied by the same designer and the need for the method was initially established in the same domain as where it is applied.

We would like to reduce bias and provide for a more general evaluation of the method. Therefore, we want to include cases in which the method is applied to develop context-aware systems in other complex environments.

Case study selection has two objectives, namely providing a representative sample, and providing a useful variation on the variables that are of interest to accommodate this hope for generalisation by the researcher (Seawright & Gerring, 2008). According to Seawright and Gerring (2008), to select cases for a case study, one should consider their within-case properties and the cross-case properties. Concerning the within-case properties, the choice for cases should respect the scope of the method. Therefore, the cases should involve the design of a context-aware system in a complex environment.

The appropriate choice for cross-case properties can be determined based on what would reduce bias. This means that the cases should involve the design of different types of systems in different complex environments. Furthermore, the designers should be different persons in the cases.

We identified three cases in which a context-aware system is developed in a complex domain. The first is the case we already mentioned of designing a context-aware architecture for B2G information sharing in the container-shipping domain. As we discuss in section 1.1, the environment of such an architecture is highly complex. The designer is the author of this dissertation. More information on the context-aware architecture and its design process can be found in part II of this dissertation.

The second case is the design of a context-aware decentralised marketplace for sensor data. The environment and system are different in this case. The system that is designed uses context-awareness to automate part of the information sharing process that is usually performed by an intermediary (Hannaert, 2018). More specifically, the system uses context to protect sensitive data and provide recommendations for sensor data sets to users (Hannaert, 2018). The environment of the decentralised marketplace can be considered complex. As the marketplace is public, there are a large number of possible participants and stakeholders (Hannaert, 2018). Furthermore, these participants upload and download a variety of sensor data sets, that vary in quality, and various other properties (Hannaert, 2018).

The decentralised marketplace was designed by a student as part of the research for their master's thesis. The student was under the supervision of the author of this dissertation. There were several extensive discussions about the method between the student and the author to explain the method, but the student applied the method mostly independently.

The third case is the design of a context-aware urban transport system. This system is different from the other systems, as its primary purpose is not to support information sharing. Instead, the system determines whether the route proposed to a truck driver in urban transport is optimal, or whether a new route should be proposed. Again, this is a highly complex environment, as there are a high variety of elements that could play a role. For example, there will be a high variety of other road users and local circumstances that influence the route the truck driver should take.

This system was designed using the method by another designer. The work on the design was performed mostly independently and the designer applied the method using the description of the method published in van Engelenburg (2019). The author of this dissertation only was involved if there were questions and kept track of any questions asked by the designer and used them as part of the evaluation.

An important limitation to this case is that at the time the information was gathered, the different parts of the design had not been integrated yet and no evaluation of the design had taken place. However, the designer did already obtain experience using the different steps of the method and thus could provide some information on the effectiveness and efficiency of the design process directly, without requiring an evaluation of the design artefact as an intermediary.

To summarise, the within-case properties of the cases are that they are a design process for a context-aware system in a complex environment. The cross-case properties are that in the cases a different designer is involved that designs a different system in a different domain. These properties are important to reduce researcher bias and to allow for broader generalisation of the result of the evaluation.

14.1.3 Data collection and analysis

As discussed in the introduction to section 4.4, the data gathering relies on human input. In other words, the designers of the context-aware system will provide information on whether they believe the method improves efficiency and effectiveness. As these designers actually use the method and are the party that execute the design process, they are able to provide information on the hypothesis.

For the case of the information sharing architecture, the designer of the architecture and the evaluator of the method are the same person, namely the author of this dissertation. This means that in this case, the researcher is a participant in the case as well. For determining the effectiveness of the method, the context model developed using the method was evaluated based on expert interviews. During the interviews, each of the context elements, context relationships and context rules were presented to the experts using the Excel-file where we collected them in the format of the table shell proposed in the method. The interviewees were asked to determine whether they were correctly derived from their support.

For the focus of willingness, we mostly relied on data from the CORE project for most context elements. We thus interviewed one of the researchers in this project to determine whether the context relationships and context elements that we derived from this information were correct and to determine whether we identified all relevant context relationships and elements. For the focus of the lawfulness of information sharing, we interviewed a juridical expert involved in our project with a background in trade law.

Due to previous cooperation, both interviewees were familiar with the research and the method prior to the interview. However, to exclude misunderstandings, the main notions, such as context element and context relationships and the goal of the architecture were discussed anyway at the start of the interview. The interviews were structured, as our main goal was to determine the quality of the context model and no other information was necessary. This also reduced the need for making a transcript, as we did not need to

apply extensive coding, for example. Instead, we just made a list of corrections and additions to the context elements and context relationships.

The experiences with the efficiency of applying the method of the author of this dissertation are of limited value. The author developed the method herself initially to help to ensure an efficient design process. Therefore, it is very likely that she experienced the design process using the architecture as efficient. To establish truly whether the method helps to provide efficiency, we need to rely on information on the experience of others as well. However, we cannot rely on outside sources to establish efficiency, as we can do for effectiveness for this case since efficiency is not established by evaluating the end product of the design process. This means that the other cases are a very important part of the evaluation, especially concerning efficiency. Nevertheless, it might still be useful to provide some reflection on the efficiency of the design process of the architecture. Of course, while keeping in mind these limitations.

For the case of the decentralised marketplace, we requested the designer to write a reflection on the use of the method. In an email, the designer was asked to focus on the effectiveness and the efficiency of the method, as well as focus broader on the advantages and disadvantages of the use of the method: *“Concerning feedback on the method, I am mainly interested in your thoughts on the effect of using the method on the efficiency and the effectiveness of the design process. With efficiency in this case I refer to the extent in which you could avoid performing steps that did not lead to a result (e.g., investigating parts of the environment that turn out not to be relevant). With effectiveness in this case I refer to the extent to which the artefact resulting from the use of the method met its requirements. Do you think that using the method made the design process more efficient and/or effective than if you would not have used the method? Why? What do you think in general are the main advantages and disadvantages of using the method in practice? Answers to these questions will be highly valuable to my research.”* As the reflection already had an appropriate structure, we summarised it for the purposes of this evaluation.

In the case of the urban transport system, we kept notes on the discussions with the designer and their work on applying the method. Furthermore, we interviewed the designer after they conducted step 1 and 2 of the method. After performing the third step, the designer wrote a reflection containing additional comments on the use of the method. Again, we focused on the effectiveness and efficiency of the design process, as well as any other more broad advantages and disadvantages. The interview was semi-structured. The reason was that we wanted to leave the possibility for the interviewee to bring up new advantages and disadvantages of the method that we might not have considered prior. We made a transcript of the interview. We then analysed the transcript by assigning codes to the comments in the interview and categorising the codes under effectiveness, efficiency, or other comments.

14.2 Results of the evaluation of the method

In this section, we discuss the results of evaluating the method for each of the cases. In each of the subsections, we discuss the results of each of the cases. Synthesis of these results will happen in section 14.3, where we discuss what they mean for the next design cycle of the method. For each of the cases, we discuss the efficiency of the design process,

which can be understood as the effort spent on deciding if environment elements are relevant (see section 3.1). Furthermore, we discuss the effectiveness of the design process, which refers to the extent to which the design goal is reached (see section 3.1).

14.2.1 A Context-aware architecture for B2G information sharing
The first case is the context-aware architecture for B2G information sharing in international container shipping. The architecture, its design process and its context are discussed elaborately in this dissertation.

14.2.1.1 Effectiveness of the design process

The researcher from the CORE project confirmed that the model of the context is correct and complete, as far as the data from CORE is concerned. The only change made based on this interview was that two context relationships were merged. Before the interview, there was another context relationship called ‘no pipeline without confidentiality’. This context relationship is shown in table 36. The interviewee stated the following about this context relationship: *“No, but this is again the issue we discussed before. They don't want competitors to see their information.”* When asked, the interviewee confirmed that she believed that this context relationship is the same as the context relationship ‘Not sharing sensitive data’ (see section 9.1.2.2). Therefore, the context relationships were merged.

Restriction/ Situation	Context elements	Adaptor/ Sensor
Restriction: Business is not willing A data pipeline is part of the flow of information. A competitor of the business uses this pipeline. Data confidentiality cannot be guaranteed.	$hasAccessSystem\left(\begin{smallmatrix} SetofDataElements, \\ System \end{smallmatrix}\right)$	Adaptor
	$systemType\left(\begin{smallmatrix} System, \\ datapipeline \end{smallmatrix}\right)$	Sensor
	$user(System, Party)$	Sensor
	$competitor\left(\begin{smallmatrix} Business, \\ SetofDataElements, \\ Party \end{smallmatrix}\right)$	Sensor
	$\neg confidentialityGuarantee(System)$	Sensor

Table 36: The context relationship ‘No pipeline without confidentiality’

The interviewee did have several remarks on other context relationships as well. However, these were either not relevant to our focus of willingness (for example remarks about scalability), or they were about the benefits of information sharing, which is out of scope as we only consider reducing the risks.

The interview with the juridical expert provided a lot of useful background information on the different sources of law that we took into consideration. This was useful to better introduce and describe these sources of law to the reader without a juridical background. Most of the context relationships identified were correct according to the interviewee. Based on the interview, only one change was made. Namely, a context

relationship was removed that restricted the focus to ‘unlawful’ in the situation that *“Data elements in the flow of information are shared via an architecture, the data elements are on a shipment, and businesses not involved in the shipment connect to the architecture.”* It is important to note that this context relationship was already flagged by the designer as requiring further investigation, as it was derived from statements of a single juridical expert that did not seem confident about it.

The interviewee also mentioned several areas of law in which additional context relationships could be identified. For example, there might be additional obligations to share information when a party suspects another party of a criminal act, such as money laundering. In addition, contract law might also be an interesting area of law for expanding the context relationships.

14.2.1.2 Efficiency of the design process

The issues with efficiency and the need for a new method were initially identified when developing the context-aware architecture. We ran into these issues early on in the project. During the first (pilot) interviews, we found that it is very easy for an interviewee or researcher to get confused about what should be modelled as part of context and what should not. For example, we found that the relationships between businesses (e.g., competition), is part of the context of information flows. Yet, the opportunity to build new relationships is part of the context of projects in which flows of information might be implemented. While the former is relevant to sense for the architecture, the latter is not. However, this is hard to discern without a criterion and method to do so. Similarly, one of the interviewees stated that the extent to which the systems in the flow of information are able to integrate with each other is important. We could view ease of integration with other systems as an attribute of a system. However, based on this statement alone, it remained unclear whether and how it should be taken into account. Another example is that an interviewee mentioned that it was hard to find a suitable data model for the data pipeline. However, this seemed the result of the number of parties involved and their legacy systems.

The method seemed to improve efficiency in two ways. First, it helped to stay on focus literally and figuratively during the interviews. As the focus of the context was made explicit, focusing and obtaining the appropriate information can be done by simply ask things like ‘How does that impact the willingness of businesses?’ Secondly, when analysing the transcripts of interviews and secondary case study data the criterion was used to quickly decide whether something is relevant. In this way, things that were not relevant could be discarded fast and early in the process.

14.2.2 A Context-aware decentralised marketplace for sensor data

In the second case of the evaluation, the method was applied by a master’s student to develop a decentralised market place for sensor data as part of the research for his master’s thesis (see (Hannaert, 2018)). This designer also wrote a reflection on his

experience with using the method, focusing on the effectiveness and efficiency of the design process. In this section, we provide a summary of this reflection.

14.2.2.1 Effectiveness of the design process

Overall, the designer considered the use of the method to be “*very effective*”. Concerning the effectiveness, the designer had several remarks. First, he considered the structure of the method its main strength. He states the following about this: “*It really forces the designer to carefully approach the problem in a structured way. It first requires articulating the main points to reach the goal i.e. articulating the foci. This helps steering the interviews to discuss what impacts these foci. I observed that the interviews I took with the foci in mind were much more concrete than the others, in the sense that interviewees were really providing me with detailed reasons for participating or not in the marketplace. Then, it asks the designer to formalize the answer of interviewees into situation and context elements. Decomposing the design process in these small pieces really helps understand why certain design choices should be made.*”

The second remark of the designer concerned the helpfulness of the method in understanding the context. The designer stated the following: “*The method is definitely helpful to figure out what is part of the context. One of the strong underlying reason for that is the suggested criterion which greatly helped me. Without this criterion, it would be quite hard to define what is part of the context, solely based on the definitions and examples. More specifically, the criterion is still somewhat abstract as the paper mentions, but it is the idea of testing whether the criterion is met or not that helps finding the situations that impact the focus.*”

As another advantage of the method, the designer mentions that it can be helpful for other designers who are new to the domain of context-aware systems to understand what constitutes them. He says the following: “*The method is also effective to design the context-aware system since it clearly describes what a context-aware system should have (sensors, adaptors, rules), which is not obvious for designers not used to context-aware systems.*”

The designer, however, also provides some information on the issues encountered when applying the method. First, he did not consider the table shell that has to be filled out by designers (see table 6, p. 94) clear enough. In his reflection, he says the following about this: “*The formulation of the situation and the related table (restriction to focus, situation and support) was a bit confusing for me on the following point. I did not understand in the first place that the focus restrictions could be different. I had the understanding that in my case these all had to be “not willing to participate”. I think that the paper could specify that there is some flexibility on that. Also, the method should emphasize (even more, as after checking it actually does already) that the way this restriction is formulated is important and should match with what the interviewees said. In my case I wrote some positively while the interviewees had actually turned his answer differently. Example: I translated “we need a mechanism to convey confidence in data quality to convince data users to participate” (interviewee) into Data quality does not match with the users’ requirements. Data users cannot judge the data quality by seeing the full dataset before buying. The focus was restricted to Data users do not want to*

participate. However, it should have been replaced by: they are willing to participate if the quality matches (positive relationship). At some point I thought it would be easy to change and just replace the formulation (since in the end the outcomes appeared to be the same), however there were more dependencies than I was expecting. As a consequence, it appeared too complex to change. It is of course my mistake since I did not fully understand this part when reading the paper, but future designers could also be led to make the same mistakes. I would therefore include a paragraph there with a word of caution for example. I think that one element that led me to this misunderstanding is the fact the distinction between negative and positive context relationship is made after introducing the table. As I probably applied the method while reading the paper, I may not have realized the importance of this distinction directly.”

In addition, the designer experienced a lack of information on how to combine and integrate physical components of the system: *“However, the method does not include everything that is required for the design of the whole context-aware system. Mainly, the physical components are the outputs of the method, but there is little information about how to integrate the physical parts that are derived from these components. In other words, we connect elements within a set of sensors, adaptors, and reasoning rules. Based on this we target which physical tool (e.g. a screen to show data) should be part of the design. However, we do not know how to connect this with physical tools defined from other sets (e.g. with the permission system). I must admit that it is complex to include this in any method since these physical tools are very specific and it is hard to imagine which general guidelines could help integrating such specific artefacts. Indications about how to evaluate the design could also be inserted in the paper.”*

Based on the reflection of the designer, we can conclude that they experienced that the method contributes to the effectiveness of their design process. However, the designer also suggests that the method could benefit from further clarification and a better connection to existing methods to improve effectiveness further.

14.2.2.2 Efficiency of the design process

According to the experience of the designer, the method has several points that make it efficient. However, he also recommends some modifications to the method to improve efficiency.

According to the designer, the systematic approach helped him not to go in the wrong direction. He stated the following: *“It saved time by having clear steps. This structure allowed me to reflect upon my work before entering new steps, helping to not get lost in irrelevant parts. As an example, in the first place I had defined three foci including one about scalability. After this step, I could assess these results and realize that the scalability was not context-aware and therefore deleted it. Without the method, I would probably have lost time by exploring this irrelevant focus.”* In addition, according to the designer, the running examples in the explanation of the method helped to make it less abstract.

The designer had some doubts about the use of schematic literals for the method. He stated the following about this: *“I think that it is indeed a strong argument to use these in order to have a language understandable by the machine. However, I am wondering if*

all designers have this objective. I would assume that some only care about coming with an architecture, but without having a structured logic understandable by a computing system. For such an objective, the complexity added because of schematic literals may refrain them from using the method. [...] In my research for example I wanted an architecture described by a BPMN but did not truly require the schematic literals since I was not actually implementing the design. Now it is true that if the designer writes with schematic literals, himself or others can more easily bring his job into a real implementation understandable by the machine. As a conclusion, the logic formalism creates a barrier for these less familiar with the topic or not willing to include it in their context-aware design. These would probably find the method rather inefficient unless they are able to quickly identify the relevant parts of the paper for them, which I doubt.” Later in the reflection, he adds to this “The context elements of the type “isData(D)” were initially not clear to me.”

14.2.3 A context-aware urban transport system

For the third case, a researcher applied the method mostly independently to design a context-aware urban transport system. In this section, we present the comments of this designer considering the effect of using the method on the effectiveness and efficiency of the design process. Some of his comments did not concern effectiveness or efficiency.

14.2.3.1 Effectiveness of the design process

Overall, the designer considered the use of the method useful. When asked why the designer wanted to use the method, he stated the following: *“So, while I was designing a smart system for synchro-model transport, there is a wealth of literature that talks about ontologies and domain knowledge and [...] rules. But they don't go the whole step. So, they don't go start from, okay, this is how I am going to collect data, this is the rules I am going to follow to interpret the data and this is my action based on the data. So, this is one of the few methods which I came across which covered the whole, the whole process.”* According to the interviewee, the alternative was taking a domain-specific approach for which it is debatable *“how methodically it will be and how applicable it will be”* and that thus might be less rigorous.

According to the designer, one of the main advantages of the method is that it is detailed and that for every step there is a logic, going from context investigation to rules. In addition, he mentions that it is an advantage that choices made in the design are documented and that it is possible to reflect on this. He states the following about this: *“So, every choice can be looked back in the next cycle of improvement, to see, oh this was the same and this is based on this requirement or this alteration and this has changed, so I need to change the rule or I need to investigate the context more. So, ehm, it made, for me, it makes it very explicit.”* When asked about the effect of this in the design process, he stated the following: *“So, the first thing which comes to mind is structure. Investigating context, it's a long and recursive process. So, you investigate the context many times during system development. And every time if you follow a cycle and you have a structured method, then you can go back in the next cycle. You can see the motivations of*

the decisions. So, the design decisions that are made in system development are well documented. So, every cycle of system improvement builds upon the previous cycle which is explicit to everybody who would want to replicate the process."

14.2.3.2 Efficiency of the design process

According to the designer, an advantage of the use of the method is that it is simple. He said the following about this: *"This is a simple method of coding context information and using the adaptors to change the, how do you say, the situation."*

The designer also noted that the examples used in the paper to illustrate how to use the method are not complex enough to *"show the full power of the method"*. He states about this: *"Because when I applied it to the transport scenario case, it is a very complex scenario and your method comes to light very quickly. It tells: 'These are the decision points.'"* The designer agreed that the method helped to decide what is relevant and what is not relevant and that this improved efficiency.

According to the designer, it is common in complex cases, such as the transport case, you need to scope. Using the method, the designer could make a list of context elements and choose from that to scope.

The designer indicated that he had some difficulty establishing the focus. The focus, in this case, had to do with the path of a vehicle in urban transport, as the context-aware system should propose such a path. The designer described his issues with establishing the focus as follows *"An optimised source and an optimised path, which is fastest, but not possible? Is that the focus? Or the focus is that a path is optimised also and feasible also."*

When asked what the impact of the use of the method was on efficiency the designer responded as follows: *"If I apply it let's say in transport, there is so much literature, that had this method not be there, I would have found my way and probably lead the same results. But if I apply it to a domain where there is not a lot of literature and there is not a lot of content over what context is, how decisions should be made, then it might be difficult for the designer to design a system there."*

When asked about the ease of understanding the method and applying it, the designer remarked that for anybody that does not have a background in logic, it might take time to understand the logic terms used. He also commented that it was unclear to him why a negative context relationship could only have one adaptor element. Furthermore, he states that it is unclear how conflicting context rules should be dealt with and that such rules should be identified.

The designer also indicated that several terms and notions were either difficult to understand or confusing. The designer asked what the difference is between a focus and the restriction to a focus. Furthermore, the designer indicated that the use of the words 'sensor', 'adaptor' and 'table shell' can be confusing as they have different meanings in other domains. For example, the designer associates 'sensor' with a physical sensor such as a GPS-sensor. The sensors in the case of the method provide context information and the designer suggested to call them context information sources instead.

14.3 Discussion and Input for the next design cycle

In this section, we discuss the findings for the different cases. In addition, we provide suggestions for the input for the next cycle in the development of the method based on the evaluation and discussion.

14.3.1 Effectiveness of the design process

The context model built using the method in the case of the context-aware architecture was largely correct according to the interviewees. This suggests that the use of the method was effective. Out of 21 context relationships, only two were changed. One of them was already flagged on beforehand as requiring further investigation.

The designer of the data marketplace and of the urban transport system both considered the use of the method to be effective. The designers mention that the structure and the systematic approach offered by the method helps to make the points at which decisions should be taken explicit and clear. According to the designer of the urban transport system, this offers the opportunity to document choices and evaluate them, which provides rigour. The designer of the marketplace mentions that the testing of whether the criterion is met helps them to identify what impacts the focus.

The designer of the marketplace mentioned some issues when applying the method. The notion of ‘focus’ and that of positive and negative context relationships was confusing to the designer. In his case, this resulted in formulating context relationships negatively where, in hindsight, it would have been better to formulate them positively. The designer of the urban transport system mentioned similar difficulty with understanding what a focus is and how to decide on whether a context relationship should be negative or positive. However, he considered it to have an impact on efficiency. The designer of the marketplace also reported that they experienced a lack of information on how to combine and integrate the physical components of the system and how to evaluate it.

14.3.2 Efficiency of the design process

The systematic approach in the method seems to contribute to efficiency as well. According to the designers of the cases of the marketplace and the urban transport system, it helps to identify things that are not relevant and this reduces the time spent on exploring things that are irrelevant. According to the designer in the case of the urban transport system, the method allows for simple coding of the context information. Furthermore, it helped them to provide a clear scope for his design.

The statements of the designer of the urban transport system were somewhat ambiguous, as he stated that without the method efficiency would be the same, as there already was a lot of literature about what the context is. However, later he states that he believes that the use of the method did improve the efficiency of their design process as it helped to identify things that are not relevant. The method focuses on improving efficiency by helping to make decisions for elements whether they are relevant. This conforms to the latter statement of the designer.

The designers encountered some difficulties when applying the method that affected efficiency. Both reported having difficulty with understanding how to use the literals and other parts of the logic that the method relies upon. It took time to understand this, which made it more difficult to apply the method. Furthermore, the designer of the marketplace suggested that formulating context rules was not necessary in their case.

Something else that could affect efficiency is that designers misunderstand the meaning of notions such as ‘sensor’ and ‘adaptor’. The meaning of these notions might not conform to how these notions are used more commonly. Especially in the case of ‘sensor’, it seems that designers tend to associate only physical sensors with that.

14.3.3 Input for the next design cycle

Based on the evaluation, it seems that the main issue with the method is that some of the notions used can be difficult to understand. Currently, our main focus was on the preciseness of the definition of context and the systematic nature of the method. However, this level of preciseness can mean that notions such as ‘focus’ are quite abstract and therefore might be more difficult to understand for some designers. The next design cycle should involve an investigation of how to make these concepts more tangible to designers and how to explain them better.

Another issue we can derive from the evaluation is that it is difficult for designers without a background in logic to understand schematic literals and logic programs, and the way in which they should be used. Furthermore, the designer of the marketplace suggests that the use of logic was unnecessary in his case.

In section 5.2.1, we discuss why we use logic in the method and more specifically use the logic-programming paradigm. We agree that in cases where a designer is not concerned with using the context rules in a decision system, for example, when they are merely interested in designing the high-level architecture of the system, it might not be useful to assign literals to context elements and generate the context rules. The designer should decide whether they need this for their design.

An approach to supporting designers without a background in logic who want to use the full method could be to provide some tool that allows designers to describe context elements in some structure that is closer to natural language and translate it to literals. When these literals are then categorised by the designer as ‘sensor elements’ or ‘adaptor elements’, the context rules can be generated automatically by the tool.

The notions of ‘sensor’ and ‘adaptor’ in the method are broader than what is commonly understood by them. In the case of the method, a sensor could be a business, for example. Commonly, only things like GPS sensors and thermometers are considered sensors. This could become an issue if it puts designers on the wrong path, which reduces efficiency. This would thus be an argument for replacing them with different notions.

However, it is questionable whether this does not add more confusion in the long term. The context-aware systems in the complex environments that we are concerned with can take into account a broader set of elements in the environment than has been common up until now. This is for example very clear in the case of the context-aware architecture in which things like the relationship between businesses and data sensitivity are taken into account. The businesses that provide information on this perform the exact same

function as a GPS sensor that senses location or a thermometer that senses temperature. From this perspective, it is not that strange to call these businesses ‘sensors’. From a technical point of view, both types of sensors deliver context information and there is no way to treat them differently.

We expect these new types of context-aware systems to become more common (see section 1.1). As this happens, the general idea of what a context-aware system is might change as well. At that stage, it might become confusing to have a different word for something that provides context information when it is similar to a business from when it is similar to a GPS sensor. Therefore, we recommend that in the next design cycle, it should be determined how to better explain these expected developments and the broader interpretation of the concepts of ‘sensor’ and ‘adaptor’ that leads to.

Currently, it has not been completely worked out how the method should be incorporated into the overall design process of a context-aware system. For example, the sensors and adaptors should connect to the basic components of the system. However, the method does not describe how this connection should be made. As suggested by the designer of the marketplace, this could be useful as well.

15 Conclusions

Our research shows that B2G information sharing in complex environments requires information flows to vary in different situations. This requires a context-aware architecture that adapts the flow of information such that information sharing is lawful and risks are avoided that might make businesses unwilling to share. The research problem we set out to solve was that there is a lack of knowledge on what the design of context-aware architectures that support business-to-government information sharing in complex environments should look like.

The motivation for solving this problem is that a context-aware B2G information sharing architecture can be used in complex environments to share additional information with government organisations, such as customs. Government organisations can use this information to improve their risk assessment. If they are better able to assess risks, then security and safety might improve. Furthermore, the chances that businesses that are not compliant are caught might increase as well. We expect businesses to be more compliant in response and this might further contribute to public safety and security.

To address the research problem, we developed a method that can be used in the design process for context-aware B2G information sharing architectures in complex environments. This method can be used to investigate context systematically and to derive the required sensors, adaptors and context rules from this insight. The method was based on a new and pragmatic definition of context that supports making easy decisions on whether something belongs to context and that can be used to model context. In addition, we provided a context-aware architecture for B2G information sharing in international container shipping. International container shipping is a typical instance of a complex environment and thus of the research problem. We applied the method to design this architecture. This work thus has three main scientific contributions: 1) a definition of context, 2) a method for designing context-aware systems, and 3) a context-aware architecture for B2G information sharing in international container shipping.

We first discuss how these contributions together help to solve the research problem and help to answer our research questions in section 15.1. This is our main contribution to scientific knowledge. In section 15.2, we discuss in more detail how each contribution by itself helps to fill a gap in scientific knowledge. In section 15.3, we discuss the limitations of this research. In section 15.4, we make recommendations for future research. In section 15.5, we reflect on technology hypes in the field of ICT.

15.1 Answering the research questions and solving the research problem

‘Design’ refers to 1) a process resulting in a design artefact, and 2) the design artefact itself. The process and its result cannot be viewed independently, because the quality of the design artefact depends on the quality of the process. In section 1.3, we formulated two corresponding research questions, namely: 1) “*What should the design process of context-aware architectures supporting business-to-government information sharing in complex environments look like?*”, and 2) “*What should a context-aware architecture*

that supports business-to-government information sharing in a complex environment look like?”

From an analysis of the complexities of designing context-aware systems in complex environments, we derived that there is a risk of a design process becoming either inefficient (i.e., spending a lot of effort on deciding if elements are relevant) or ineffective (i.e., not reaching design goals) (see section 3.1). To reduce these risks, we established that the method should meet the following objectives: 1) supporting the designer in systematically investigating and modelling the relevant context for their system and 2) supporting the designer in deriving the sensors, adaptors and context rules their system requires from their model of context (see section 3.2). Based on an analysis of the relevant literature, we concluded that it does not provide a method that meets these objectives (see section 3.3).

The answer to the first research question is provided in the form of the method we described (see chapter 6). Namely, the method provides knowledge on how to design context-aware systems, including context-aware architectures for B2G information sharing. The method relies on the definition of context presented in chapter 5. We demonstrated the method by applying it in the design process of the context-aware B2G information sharing architecture in international container shipping (see chapter 12). The validation of the context model we developed for this architecture contributed to the evaluation of the method, as it shows its effectiveness (see chapter 14).

As we discussed in chapter 3, there is a need for a method that can be used to systematically investigate and model context for designing context-aware systems in complex environments in general. To confirm that the method is generalizable, we evaluated the method in two other cases where it is used by other designers than the author of this dissertation, to design other systems than the context-aware architecture and in other domains than international container shipping (see chapter 14). In both cases, the method turned out to be useful. In a sense, for the method, we thus generalise beyond what is necessary to answer research question 1.

The answer to the second research question is provided in the form of the architecture for B2G information sharing discussed in part II of this thesis. The architecture was developed for a single complex environment, namely that of international container shipping. The international container-shipping domain is a typical instance of a complex environment and we thus solved a typical instance of the research problem (see chapter 7). We limited the scope for the model of context we incorporated in the architecture. For its focus of the lawfulness of information flows, we only took into account customs law, intellectual property law and competition law. For its focus of willingness of businesses to participate in information sharing, we limited the investigations to reducing risks of information sharing.

The context-aware architecture for B2G information sharing in international container shipping consists of two parts, which allows us to generalise partially the results to other domains than international container shipping. The first part consists of the sensors, adaptors and context rules (see chapter 10). This part of the architecture is derived from the context model (see chapter 9) using the new method and thus is specific for the container-shipping domain.

The second part of the architecture consists of its basic components and how these are related to each other and the sensors and adaptors (see chapters 11 and 12). This part of the architecture supports the storing of context information, deciding on an appropriate flow of information, and sharing information according to the flow decided upon (see section 11.1). Components to provide this functionality would be necessary for any context-aware B2G information sharing architecture and they thus do not depend on the domain.

The second ‘basic’ part of the presented in chapter 11 could be used to provide the same functionalities in other domains and could thus be used as a reference architecture. The method presented in chapter 6 can then be used to derive the context-dependent components (i.e., sensors, adaptors and rules) for that domain. They can then be connected to the basic components in the same way as we did. In this way, this work provides a solution to the research problem in various domains.

15.2 Scientific contribution

The main scientific contribution of this work is provided by the combination of different contributions, which answers the research questions and helps to solve the research problem. However, each of the contributions by themselves also adds to the existing scientific knowledge. We discuss these individual contributions in this section.

15.2.1 A new definition of context

In chapter 5, we presented a new definition of context. Where existing definitions rely on more positivistic assumptions (e.g., (Dey & Abowd, 1999)) or interpretivistic assumptions (e.g., (Dourish, 2004)) the complex environments that are the focus of this research require a more pragmatic perspective. Pragmatist assumptions have influenced the definition in two ways: 1) what is considered context depends on what is useful to designers, and 2) the form of the definition should be useful to designers. More specifically, we defined context such that something belongs to context if it is useful to take into account for a designer when designing a system, considering their design goal. In addition, the definition is highly specified and formalised, such that it provides a domain-independent conceptual model that can be useful to designers to systematically investigate and represent context.

The literature on designing context-aware systems describes that designers need a shared conceptual model of context that can help them to better understand and represent context (see section 3.3). The need for such a conceptual model might be even greater in complex environments as these contain many elements that could belong to the relevant context. In contrast with the existing work, our definition offers a conceptual model that can be used to systematically investigate and represent context in complex environments. That is to say, we developed a method for designing context-aware systems in complex environments that uses the definition to do so.

The specificity that our definition offers comes at the cost of a high level of abstractness and complexity. This makes the definition less useful when used in less complex environments. A designer considering using our definition should consider this

and determine whether the complexity of the context they want to investigate and model merits relying on this definition.

15.2.2 A method for designing context-aware systems

In chapter 6, we presented a method for designing context-aware systems in complex environments. We established the lack of such a method when designing the context-aware architecture for B2G information sharing in international container shipping. What sensors, adaptors and context rules should be included in the architecture directly depends on what it should sense and adapt to, and in what way. However, the international container-shipping domain is highly complex. This means that it contains many elements that could be relevant to take into account. Because of this, there are two risks. The first is that the design process might become inefficient, as a lot of time is spent on investigating irrelevant context elements and on making decisions. The other risk is that the design process might become ineffective when instead of investigating the context systematically, assumptions are made on what is relevant. As we discuss in chapter 3, these risks are not only present in the international container-shipping domain, but in any complex environment in which there are a lot of elements that could be taken into account in the design.

Therefore, to support designing context-aware systems in complex environments, a method is needed for supporting the designer in systematically investigating and modelling the relevant context for their system and supporting the designer in deriving the sensors, adaptors and context rules their system requires from their model of context (see section 3.2). The existing work does not offer a method that provides enough support in complex environments (see section 3.3). The method we present in chapter 6 does.

We evaluated the method by applying it in three cases of design processes to ensure that the method was useful to other designers than the author of this dissertation, to design different types of context-aware systems, and in different environments. From the evaluation, we can derive that the main advantages of the method are that it is very systematic and forces designers to be concrete and explicit about the decisions they make and to document them. This helps to provide for an effective and rigorous design process. In addition, it enhances efficiency, as decisions to take into account elements are made explicitly and early in the process.

As the method is based on a very specific, albeit abstract and complex definition of context, some designers might find some notions difficult to understand. Therefore, it should be investigated how to better explain these notions. Furthermore, designers might have difficulty to understand how to use the formal logic that is part of the method.

In addition, some designers might only want to provide for a very high-level design of their system and therefore establishing context rules might be out of their scope. However, designers that do want to derive context rules could be supported by a tool that helps to translate a description of context elements in simple natural language to literals. Future research could also focus on how to connect the different sensors and adaptors found using the method to the basic components of a context-aware system in different

domains. Furthermore, it could focus on how conflicts between context rules in the logic program should be dealt with in different domains and for different purposes.

15.2.3 A context-aware architecture for B2G information sharing in international container shipping

The model of the context of B2G information sharing in the international container-shipping domain presented in chapter 9 and the related adaptors and sensors presented in chapter 10 are a contribution to scientific knowledge as well. This part of the research contributes specifically to the knowledge in the domain of B2G information sharing in international container shipping. For establishing the model, we focused on the lawfulness of information flows, considering IP law, customs law and competition law. Furthermore, we focused on reducing the risks of information sharing for businesses such that they are willing to participate in information flows.

The information on lawfulness and willingness was obtained from interviews with several experts and case study data that we performed a secondary data analysis on (see sections 9.1.1 and 9.2.1). We made this knowledge explicit and structured it in such a way that sensors, adaptors and context rules can be derived from it. Such knowledge is a useful addition to the existing knowledge on B2G information sharing in international container shipping. Namely, it is required to design a context-aware architecture that supports information sharing in this domain (see section 7.5).

For the context model, we took into account the lawfulness of information flows and the willingness of businesses to participate in information flows. Concerning the lawfulness of information flows, we only took into account the juridical domains of customs law, intellectual property law and competition law. Concerning the willingness of businesses to participate in information flows the scope was limited to reducing the risks of information sharing. Expanding the scope might lead to identifying additional context elements, relationships and rules that can be taken into account in the model.

In chapter 12, we presented a context-aware architecture for B2G information sharing in international container shipping. We established the need for such an architecture to improve compliance (see chapter 7). Furthermore, we found that especially the lawfulness of information sharing and the willingness of businesses to participate in information sharing are important to support B2G information sharing in the domain (see section 7.5).

The architecture provides knowledge on how a combination of an information flow planner, a context repository and information routers can be used to adapt information flows to the context to support information sharing in a variety of situations (see chapter 11). This knowledge is on how to support these various information flows from a technical point of view. This is not dependent on the container-shipping domain. In addition, we show how domain-specific sensors, adaptors and context rules can be derived from a model of context and be incorporated in the architecture (see section 10).

The architecture has several important properties that are necessary to support B2G information sharing. First, it overarches several other solutions for sharing information in the international container-shipping domain. These solutions, such as single windows and data pipelines are not replaced by the context-aware architecture,

which means that their functionality is not lost but incorporated in the architecture instead. Access control rules are used to control access to data from a technical point of view, and at the same time, they provide juridical protection over data. This provides control over data to users.

We use blockchain technology to store context information, context rules and proposed information flows in the architecture to provide for secure and auditable access control, without providing a central party with a lot of control over the information shared (see section 11.5). While this makes it difficult to tamper with access control, blockchain technology does suffer from scalability issues and some control of central parties (e.g., to provide software or identity management) seems to be unavoidable. This left us not fully convinced that blockchain technology is the best technology to rely on for the architecture. Further research is necessary to obtain more insight into its effects and to establish how fundamental these issues are. It is important to weight the costs of using blockchain technology, compared with other technologies, against its benefits. This research already contributes to further investigating in blockchain technology, as it provides knowledge on what trade-offs there are between the different properties of blockchain technology and it suggests what issues are relevant to address.

Based on the evaluation of the architecture, we can conclude that the overall architecture is useful to customs as well as businesses (see chapter 13). The results also suggest that an important next step is to investigate how the architecture can be implemented in practice at a large scale and how to arrange governance of the architecture.

15.3 Research limitations

This work is based on a pragmatic research approach in which we want the knowledge we generated to be useful as well as true. The extent to which this is the case is determined by the relevance and the rigour of the research conducted. We thus agree with Hevner (2007) that our research should be relevant as well as rigorous. We will use this to analyse the limitations for each of the contributions.

15.3.1 A new definition of context

For the new definition of context, we reviewed the existing literature on definitions of context to ensure that the definition was innovative. For the syntax of the definition, we relied on the logic-programming paradigm as described by Lifschitz (1996). Furthermore, we used some existing terms, such as context element and focus.

The definition we developed was evaluated by evaluating a method that is based on it. We thus did not directly evaluate the definition. Instead, we assumed that if the method reaches its objectives, the definition of context is a suitable basis for it.

Concerning the relevance of this contribution, we established the lack of a method in practice when designing the context-aware architecture. We then established the need for a definition of context when developing the method. In addition, the literature describes that designers have a need for a conceptual model of context and that this need is not met based on previous scientific work.

15.3.2 A method for designing context-aware systems

At the start of the research, we reviewed the relevant literature and found that there are no existing methods for systematically investigating context in complex environments. In addition, we did not find any work we could use as kernel theories for developing our method. Therefore, it was difficult to establish rigour of the research. However, on the other hand, we did specify a clear design process based on the activities of Peffers et al. (2007), which improves rigour from a methodological point of view.

We established the relevance of developing a method during our own research into a context-aware architecture for B2G information sharing. We derived that the method could be useful in other cases as well, based on an analysis and a literature review. However, the evidence from the literature is indirect and states that designers need a conceptual model and ways to get more insight into context. It does not conclude that the efficiency and effectiveness of the design process might be at risk otherwise.

The efficiency and effectiveness of a design method are very difficult to measure objectively. How much effort is spent on deciding if elements are relevant or the extent to which the appropriate components are taken into account depends highly on the system, environment and the designer. Comparing efficiency and effectiveness with and without the method would thus not yield useful information, as there are too many variables that cannot be controlled. However, a designer using the method can discuss whether the method helped them to spend less effort investigating elements and on whether it helped them to design an effective system. The experience of the designer could thus be a useful source of information.

For the evaluation, we thus relied on the subjective experiences of the designers in the evaluation. While this information can be useful, experiences still can be influenced by many other variables, including characteristics of the designers. We did not use a control group and we cannot rule out the influence of other variables.

The role of the researcher as a designer might result in a bias. Therefore, tried to avoid interfering with the results of the evaluation by asking others to use the method. We asked a student to apply the method to develop a context-aware marketplace. Based on his reflection we found that the method does contribute to ensuring an efficient and effective design process, but that the explanation of the method could be improved.

There needed to be a lot of communication between the author of this dissertation and the student about the method and therefore the influence of the author was relatively big. Even though the author tried to avoid interfering with the decisions the student made on what belongs to context and deriving a design from it. In addition, it is important to mention that the author of this dissertation was involved with assessing the work of the student and providing them with a mark for their master's thesis. The student was ensured several times that any positive or negative reflection on the method would not influence the results of the assessment of their work. Furthermore, the student was asked to send his reflection only after he had received his mark. While this might have made it easier for the student to provide an honest reflection on the method, the dynamics are still such that complete independency of the student cannot be expected. Nevertheless, the student discussed some advantages as well as some disadvantages of the use of the method. This suggests that the influence of the researcher on the reflection of the student were limited.

The researcher that applied the method to develop a context-aware urban transport system can be considered in the position to make an independent assessment of the method. For this part of the evaluation, we also found that the method contributes to efficiency and effectiveness, but that it could benefit from an additional explanation. However, there might be a selection bias in this case, as the researcher was already convinced of the usefulness of the method before applying it. Otherwise, they would not have been willing to put in the effort to use the method.

Another limitation of the research is that the method was only evaluated based on reflections and interviews and that the context-aware systems that were developed using it were not implemented and field-tested in practice. This was not possible, as it would have required many resources and might have taken years. However, this does mean that in practice still unforeseen issues with the systems could be encountered. It is therefore important that future research does incorporate this final step of the evaluation and that this can be used to update the method.

15.3.3 A context-aware architecture for B2G information sharing in international container shipping

The context model of the lawfulness of information flows and the willingness of businesses to participate in information flows in B2G information sharing in international container shipping was based on information from interviews and a secondary analysis of case study data. These data sources and manner of gathering data are associated with some limitations.

First, the data in the case study was not gathered for the purpose for which we used it. This is associated with the risk of important details being left out (Bowen, 2009; Runeson & Höst, 2009). To reduce these risks, we complemented this analysis with interviews with experts and we were in regular contact with the researchers involved in the original study. Furthermore, we involved one of these researchers in validating the context model.

The interviewees we involved consisted of experts from academia as well as industry. All of them were willing to cooperate with the interviews. This means that a priori they might have been more open to possible benefits of information sharing and supporting it, otherwise, they might not have been interested to be involved in the research. The same goes for the parties involved in the case studies that we analysed. Especially the results concerning willingness might have been different when we would have included experts from industry that are more sceptical about information sharing in the domain.

The design problem for the context-aware architecture was established based on the relevant literature, which provided a detailed overview of the need as well as the difficulties with supporting information sharing in the international container-shipping domain. Based on the literature we established that the architecture would be innovative as well.

When designing the system we applied several existing principles from the literature, such as the piggybacking principle. Furthermore, we designed the architecture such that existing systems supporting B2G information sharing in international container

shipping can be incorporated in it. Concerning the methods used, it was more difficult to ensure rigour. In fact, designing the architecture merited the development of a new method for designing context-aware systems. However, as this new method is very systematic, we believe that the fact that we applied it contributes to rigour as well.

For the evaluation of the architecture, we presented the work at workshops to domain experts with a background in ICT and with a juridical background. Furthermore, we interviewed a customs expert and an expert in ICT and governance from academia. For the evaluation, we relied on their judgement of what they thought the advantages and disadvantages of the architecture would be, and whether it would be lawful. We did not implement a prototype and evaluated the architecture in practice. This would have required more resources than those available as the architecture is designed to overarch other existing systems and is most useful at a large scale. However, using the architecture in practice might reveal new properties and yield different results for the evaluation.

15.4 Recommendations for future research on context-aware systems

In the previous section, we already provide some recommendations on how the existing work could be used to extend the results of this research. However, we also want to discuss our views on the more broad developments within the domain of context-aware systems.

In the introduction of this dissertation, we discussed the notion of complex multi-actor environments that involves elements with a variety of attributes and relationships with each other, such as a high variety of systems, stakeholders and data. Society seems to rely more and more on technology and the sharing and processing of data. As we discussed in section 1.1, when developments such as the Internet of Things (IoT), big data analytics and blockchain technology are used at a larger scale, systems relying on these technologies might need to function in ever more complex environments.

It seems likely that with the environment of systems becoming more complex, more often requirements will be different in different situations. As we discussed in this research, a way of dealing with such conditional requirements is by making systems context-aware and making them adapt automatically. We thus might need systems to be context-aware more often in the future.

However, we might also need new types of context-aware systems. The context-aware architecture we developed does not conform to the typical view of what a context-aware system is. A typical example of a traditional context-aware system is that of the context-aware tour-guide. Such a system gathers information on location using a GPS sensor and then provides relevant information to its user. In contrast, in the context-aware architecture, businesses and various other parties act as sensors and provide context information, and then the architecture adapts by providing for a certain information flow. At first sight, this seems different. While things like a GPS sensor will certainly conform to the general idea of a sensor, the same cannot be said of a business, for example.

This raises important questions. First, can the architecture we developed be viewed as ‘context-aware’ in the traditional sense? We defined a context-aware system as having a higher level of autonomy than systems that are not context-aware. However,

we did not specify what level of autonomy a system should have to be considered context-aware. Can a system receiving context information from businesses and other parties be considered as context-aware as a system only receiving context information from sensors that perform physical measurements? Can a business providing information that data is sensitive be viewed as providing a command to a system and thereby reducing its autonomy? If that is the case, what about the user of a tour guide that walks to the vicinity of a certain sight because they want information on that?

In addition, there is also the class of adaptive systems that have an even higher level of autonomy as they even automate the formulation of the rules for adapting using algorithms based on techniques like machine learning. Is this just another class of systems further on the same scale? Future research could focus on answering these questions, which could lead to *a better understanding of 'context-awareness' and the kind of context-awareness that is required in complex environments.*

Whether or not the context-aware architecture should be considered of a different type, viewing context more broadly, like for the architecture, can be beneficial. It allows the system to take into account and adapt to a broader set of elements, which means that it could be used in more situations. There is no clear reason to exclude such elements just because they cannot be measured by what is traditionally considered a sensor. However, future research could focus on *determining how and to what extent such context-aware systems are useful in practice.*

15.5 Reflections on technology hypes in the field of ICT

In the field of ICT, new technological developments follow each other rapidly. These developments are often accompanied by great spikes in interest for the new technology in question in society as well as academia. An example of such a development during the research for this thesis was blockchain technology. Previously, the same thing happened for other technologies, such as the cloud.

These technologies are often presented as being completely new. Blockchain technology, for example, is presented as a revolutionary new type of technology. The issue with this is that it ignores the fact that most of these technologies consist of components that have been studied for years and on which a lot of information is available. For example, blockchain technology relies on distributed information sharing which has been studied for years. There will be effects of distributed information sharing that are likely to occur with blockchain technology.

To deal with the hypes and the rapid developments of new technologies, previous work should not be ignored. Recognising the parts of new technologies that already are familiar might help with understanding them faster. Furthermore, focusing more on synthesising knowledge from the existing work will make it easier to predict the effects of using new technologies, such as blockchain technology.

REFERENCES

- Aamodt, A., & Nygård, M. (1995). Different roles and mutual dependencies of data, information, and knowledge - An AI perspective on their integration. *Data and Knowledge Engineering*, 16(3), 191–222. [https://doi.org/10.1016/0169-023X\(95\)00017-M](https://doi.org/10.1016/0169-023X(95)00017-M)
- Abowd, G. D., Dey, A., Brown, P., Davies, N., Smith, M., & Steggles, P. (1999). Towards a better understanding of context and context-awareness. *Handheld and Ubiquitous Computing*, 304–307. https://doi.org/10.1007/3-540-48157-5_29
- Ackoff, R. L. (1989). From data to wisdom. *Journal of Applied Systems Analysis*, 16(1), 3–9.
- Adriaans, P. (2013). Information. In *The Stanford Encyclopedia of Philosophy* (Fall 2013). Retrieved from <http://plato.stanford.edu/entries/information/>
- Alegre, U., Augusto, J. C., & Clark, T. (2016). Engineering context-aware systems and applications: A survey. *Journal of Systems and Software*, 117, 55–83. <https://doi.org/10.1016/j.jss.2016.02.010>
- Alter, S. (1996). *Information Systems: A Management Perspective*. The Benjamin/Cummings Publishing Company.
- Alter, S. (2008). Defining information systems as work systems: implications for the IS field. *European Journal of Information Systems*, 17(January), 448–469. <https://doi.org/10.1057/ejis.2008.37>
- Androulaki, E., Barger, A., Bortnikov, V., Cachin, C., Christidis, K., Caro, A. De, ... Yellick, J. (2018). Hyperledger fabric: A Distributed Operating System for Permissioned Blockchains. *Proceedings of the Thirteenth EuroSys Conference*. <https://doi.org/10.1145/3190508.3190538>
- Anhalt, J., Smailagic, A., Siewiorek, D. P., Gemperle, F., Salber, D., Weber, S., ... Jennings, J. (2001). Toward Context-Aware Computing: Experiences and Lessons. *IEEE Intelligent Systems*, 16(3), 38–46.
- Anton, A. I. (1997). *Goal identification and refinement in the specification of software-based information systems*. Georgia Institute of Technology.
- Atzori, L., Iera, A., & Morabito, G. (2010). The Internet of Things: A survey. *Computer Networks*, 54(15), 2787–2805. <https://doi.org/10.1016/j.comnet.2010.05.010>
- Augustin, I., Yamin, A. C., Da Silva, L. C., Real, R. A., Frainer, G., & Geyer, C. F. R. (2006). ISAMadapt: Abstractions and tools for designing general-purpose pervasive applications. *Software - Practice and Experience*, 36(11–12), 1231–1256. <https://doi.org/10.1002/spe.756>
- Baldauf, M., Dustdar, S., & Rosenberg, F. (2007). A survey on context-aware systems. *International Journal of Ad Hoc and Ubiquitous Computing*, 2(4), 263. <https://doi.org/10.1504/IJAHUC.2007.014070>
- Barak, A. (2007). What is Legal Interpretation? In *Purposive Interpretation in Law* (Vol. 34, pp. 3–15).
- Barber, S., Boyen, X., Shi, E., & Uzun, E. (2012). Bitter to Better — How to Make Bitcoin a Better Currency. *International Conference on Financial Cryptography and Data Security*, 399–414. https://doi.org/10.1007/978-3-642-32946-3_29
- Batubara, R., Ubacht, J., & Janssen, M. (2018). Challenges of Blockchain Technology Adoption for e-Government: A Systematic Literature Review. *Proceedings of 19th Annual International Conference on Digital Government Research (Dg.o'18)*. <https://doi.org/10.1145/3209281.3209317>
- Bauer, J. S., Newman, M. W., & Kientz, J. A. (2014). What Designers Talk About When They Talk About Context. *Human-Computer Interaction*, 29(5–6), 420–450. <https://doi.org/10.1080/07370024.2014.896709>
- Ben-daya, M., Hassini, E., & Bahrour, Z. (2017). Internet of things and supply chain management: a literature review. *International Journal of Production Research*, 7543, 1–23. <https://doi.org/10.1080/00207543.2017.1402140>
- Benbasat, I., Goldstein, D. K., & Mead, M. (1987). The Case Research Strategy in Studies of Information Case Research. *Management Information Systems Quarterly*, pp. 369–386.
- Benou, P., & Vassilakis, C. (2010). The conceptual model of context for mobile commerce

- applications. *Electronic Commerce Research*, 10(2), 139–165. <https://doi.org/10.1007/s10660-010-9050-4>
- Berry, D. M., Cheng, B. H. C., & Zhang, J. (2005). The Four Levels of Requirements Engineering for and in Dynamic Adaptive Systems. *11th International Workshop on Requirements Engineering: Foundation for Software Quality (REFSQ'05)*, 113–120.
- Bettini, C., Brdiczka, O., Henriksen, K., Indulska, J., Nicklas, D., Ranganathan, A., & Riboni, D. (2010). A survey of context modelling and reasoning techniques. *Pervasive and Mobile Computing*, 6(2), 161–180. <https://doi.org/10.1016/j.pmcj.2009.06.002>
- Bharosa, N., Janssen, M., van Wijk, R., de Winne, N., van der Voort, H., Hulstijn, J., & Tan, Y.-H. (2013). Tapping into existing information flows: The transformation to compliance by design in business-to-government information exchange. *Government Information Quarterly*, 30(1), S9–S18. <https://doi.org/10.1016/j.giq.2012.08.006>
- Blockchain size. (2018). Retrieved November 21, 2018, from <https://charts.bitcoin.com/btc/chart/blockchain-size>
- Bolchini, C., Schreiber, F. A., & Tanca, L. (2007). A methodology for a Very Small Data Base design. *Information Systems*, 32(1), 61–82. <https://doi.org/10.1016/j.is.2005.05.004>
- Bostrom, R. P., & Heinen, J. S. (1977). MIS Problems and Failures: A Socio-Technical Perspective. Part I: The Causes. *Management Information Systems Quarterly*, 1(3), 17–32.
- Bowen, G. A. (2009). Document Analysis as a Qualitative Research Method. *Qualitative Research Journal*, 9(2), 27–40. <https://doi.org/10.3316/QRJ0902027>
- Bray, R. L., & Mendelson, H. (2012). Information Transmission and the Bullwhip Effect: An Empirical Investigation. *Management Science*, 58(5), 860–875. <https://doi.org/10.1287/mnsc.1110.1467>
- Brenton, P., & Imagawa, H. (2005). Rules of Origin, Trade and Customs. In L. De Wulf & J. B. Sokol (Eds.), *Customs Modernization Handbook*. The World Bank.
- Brézillon, P. (2005). Task-realization models in contextual graphs. *International and Interdisciplinary Conference on Modeling and Using Context*, 55–68. https://doi.org/10.1007/11508373_5
- Brézillon, P., & Pomerol, J. (1999). Contextual Knowledge Sharing And Cooperation In Intelligent Assistant Systems. *Le Travail Humain*, 62(3), 223–246.
- Bryman, A. (2006). Paradigm peace and the implications for quality. *International Journal of Social Research Methodology: Theory and Practice*, 9(2), 111–126. <https://doi.org/10.1080/13645570600595280>
- Burrell, G., & Morgan, G. (1979). *Sociological Paradigms and Organisational Analysis Elements*. Ashgate Publishing Company.
- Burnard, P. (1991). A method of analysing interview transcripts in qualitative research. *Nurse Education Today*, 11(July), 461–466. [https://doi.org/10.1016/0260-6917\(91\)90009-Y](https://doi.org/10.1016/0260-6917(91)90009-Y)
- Burnett, K., Ng, K. B., & Park, S. (1999). A comparison of the two traditions of metadata development. *Journal of the American Society for Information Science*, 50(13), 1209–1217. [https://doi.org/10.1002/\(SICI\)1097-4571\(1999\)50:13<1209::AID-ASI6>3.0.CO;2-Y](https://doi.org/10.1002/(SICI)1097-4571(1999)50:13<1209::AID-ASI6>3.0.CO;2-Y)
- Buterin, V. (2015). On Public and Private Blockchains. Retrieved April 18, 2017, from Ethereum website: <https://blog.ethereum.org/2015/08/07/on-public-and-private-blockchains/>
- Cachin, C. (2016). Architecture of the Hyperledger Blockchain Fabric. *Online*. <https://doi.org/10.4230/LIPIcs.OPODIS.2016.24>
- Cano, J., Delaval, G., & Rutten, E. (2014). Coordination of ECA rules by verification and control. *International Conference on Coordination Languages and Models*, 33–48. Springer, Berlin, Heidelberg.
- Carvalho, J. Á. (2000). Information System? Which One Do You Mean? *Information Systems Concepts: An Integrated Discipline Emerging. Proceedings of the ISCO 4 Conference*, 259–280. https://doi.org/10.1007/978-0-387-35500-9_22
- Casas, R., Cuartielles, D., Marco, Á., Gracia, H. J., & Falcó, J. L. (2007). Hidden issues in deploying an indoor location system. *IEEE Pervasive Computing*, 6(2), 62–69. <https://doi.org/10.1109/MPRV.2007.33>

- Chen, H., Finin, T., & Joshi, A. (2005). SemanticWeb in a Pervasive Context-Aware Architecture. In *Defense Technical Information Center*. Retrieved from <https://apps.dtic.mil/dtic/tr/fulltext/u2/a439730.pdf>
- Chen, M., Mao, S., & Liu, Y. (2014). Big data: A survey. *Mobile Networks and Applications*, 19(2), 171–209. <https://doi.org/10.1007/s11036-013-0489-0>
- Chen, W., & Hirschheim, R. (2004). A paradigmatic and methodological examination of information systems research from 1991 to 2001. *Information Systems Journal*, 14, 197–235.
- Chow, H. K. H., Choy, K. L., & Lee, W. B. (2007). A strategic knowledge-based planning system for freight forwarding industry. *Expert Systems with Applications*, 33(4), 936–954. <https://doi.org/10.1016/j.eswa.2006.08.004>
- Cole, R., Purao, S., Rossi, M., & Sein, M. (2005). Being Proactive: Where Action Research meets Design Research. *ICIS 2005 Proceedings*, 1–21.
- Colmerauer, A. (1975). *Les grammaires de métamorphose*.
- Colombo, E., Mylopoulos, J., & Spoletini, P. (2005). Modeling and analyzing context-aware composition of services. *International Conference on Service-Oriented Computing*, 3826 LNCS, 198–213. https://doi.org/10.1007/11596141_16
- Cooper, M. C., Ellram, L. M., Gardner, J. T., & Hanks, A. M. (1997). Meshing multiple alliances. *Journal of Business Logistics*, 18(1), 67.
- Creswell, J. W. (2014). The Selection of a Research Approach. In *Research design: Qualitative, quantitative, and mixed methods approaches*. (4th ed., pp. 3–23). SAGE Publications, London.
- Crosby, M., Nachiappan, Pattanayak, P., Verma, S., & Kalyanaraman, V. (2016). BlockChain Technology: Beyond Bitcoin. *Applied Innovation Review*, (2).
- Cross, N. (2001). Designerly Ways of Knowing: Design Discipline Versus Design Science. *Design Issues*, 17(3), 49–55. <https://doi.org/10.1162/074793601750357196>
- Crowley, J. L. (2003). Context driven observation of human activity. *Ambient Intelligence. EUSAI*, 101–118. https://doi.org/10.1007/978-3-540-39863-9_9
- Crowley, J. L., Coutaz, J., & Reigner, P. (2002). Perceptual Components for Context Aware Computing. *International Conference on Ubiquitous Computing*, 117–134. <https://doi.org/10.1007/3-540-45809-3>
- Customs Administration of the Netherlands. (2014). *Pushing boundaries: The Customs Administration of The Netherlands' Point on the Horizon for the Enforcement on Continuously Increasing Flows of Goods*. the Hague.
- Dabholkar, P. A., & Neeley, S. M. (1998). Managing interdependency: a taxonomy for business to business relationships. *Journal of Business & Industrial Marketing*, 13(6), 439–460. <https://doi.org/10.1108/08858629810246797>
- Dana, S. (2009). Dana, Shahram. "Beyond retroactivity to realizing justice: A theory on the principle of legality in international criminal law sentencing. *The Journal of Criminal Law & Criminology*, 99(4), 857–928.
- Dannen, C. (2017). *Introducing Ethereum and Solidity*. <https://doi.org/10.1007/978-1-4842-2535-6>
- Darke, P., Shanks, G., & Broadbent, M. (1998). Successfully completing case study research: combining rigour, relevance and pragmatism. *Information Systems Journal*, 8(4), 273–289. <https://doi.org/10.1046/j.1365-2575.1998.00040.x>
- Dawes, S. S. (1996). Interagency information sharing: Expected benefits, manageable risks. *Journal of Policy Analysis and Management*, 15(3), 377–394. [https://doi.org/10.1002/\(SICI\)1520-6688\(199622\)15:3<377::AID-PAM3>3.0.CO;2-F](https://doi.org/10.1002/(SICI)1520-6688(199622)15:3<377::AID-PAM3>3.0.CO;2-F)
- Delmolino, K., Arnett, M., Kosba, A., Miller, A., & Shi, E. (2016). Step by Step towards Programming a Safe Smart Contract: Lessons and Insights from a Cryptocurrency Lab. *International Conference on Financial Cryptography and Data Security*, 1–10. https://doi.org/10.1007/978-3-662-53357-4_6
- Dempsey, L., & Heery, R. (1998). Metadata: a current view of practice and issues. *Journal of Documentation*, 54(2), 145–172. <https://doi.org/10.1108/EUM0000000007164>

- Denscombe, M. (2008). Communities of Practice: A Research Paradigm for the Mixed Methods Approach. *Journal of Mixed Methods Research*, 2(3), 270–283. <https://doi.org/10.1177/1558689808316807>
- Denzin, N. K. (2012). Triangulation 2.0. *Journal of Mixed Methods Research*, 6(2), 80–88. <https://doi.org/10.1177/1558689812437186>
- Dey, A. K. (2000). Providing Architectural Support for Building Context-Aware Applications (Vol. 16). https://doi.org/10.1207/S15327051HCI16234_02
- Dey, A. K. (2001). Understanding and Using Context. *Personal Ubiquitous Comput.*, 1(5), 4–7. <https://doi.org/10.1007/s007790170019>
- Dey, A. K., & Abowd, G. D. (1999). Towards a Better Understanding of Context and Context-Awareness. *Computing Systems*, 40(3), 304–307. https://doi.org/10.1007/3-540-48157-5_29
- DG Taxation and Customs Union. (2018). EU Customs Union – unique in the world. Retrieved August 21, 2018, from https://ec.europa.eu/taxation_customs/facts-figures/eu-customs-union-unique-world_en
- Dockhorn Costa, P., Ferreira Pires, L., & van Sinderen, M. J. (2005). Architectural patterns for context-aware services platforms. *Second International Workshop on Ubiquitous Computing*, 3–18. Retrieved from <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.96.7367&rep=rep1&type=pdf>
- Douceur, J. R. (2002). The Sybil Attack. *International Workshop on Peer-to-Peer Systems*, 1–6. https://doi.org/10.1007/3-540-45748-8_24
- Dourish, P. (2004). What we talk about when we talk about context. *Personal and Ubiquitous Computing*, 8(1), 19–30.
- Eisenhardt, K. M. (1989). Building Theories from Case Study Research. *Academy of Management Review*, 14(4), 532–550. <https://doi.org/10.5465/AMR.1989.4308385>
- Elsevier B.V. (2017). *Scopus: Content Coverage Guide* (p. 28). p. 28. Retrieved from https://www.elsevier.com/_data/assets/pdf_file/0007/69451/0597-Scopus-Content-Coverage-Guide-US-LETTER-v4-HI-singles-no-ticks.pdf
- European Commission Directorate-General Taxation and Customs Union. (2016). *SAD Guidance During the UCC transitional period Disclaimer* (pp. 1–81). pp. 1–81.
- European Economic and Social Committee. (2003). Opinion of the European Economic and Social Committee on the ‘Security of Transports.’ *Official Journal of the European Union*, 46, 174–183.
- Eyal, I., Gencer, A. E., Sirer, E. G., & van Renesse, R. (2016). Bitcoin-NG : A Scalable Blockchain Protocol. *Proceedings of the 13th USENIX Symposium on Networked Systems Design and Implementation (NSDI '16)*, 45–59.
- Fang, X., Misra, S., Xue, G., & Yang, D. (2012). Smart Grid — The New and Improved Power Grid: A Survey. *IEEE Communications Surveys & Tutorials*, 14(4), 944–980. <https://doi.org/10.1109/SURV.2011.101911.00087>
- Fang, Z. (2002). E-government in digital era: concept, practice, and development. *International Journal of The Computer, The Internet and Management*, 10(2), 1–22.
- Fawcett, S. E., Osterhaus, P., Magnan, G. M., Brau, J. C., & McCarter, M. W. (2007). Information sharing and supply chain performance: the role of connectivity and willingness. *Supply Chain Management: An International Journal*, 12(5), 358–368. <https://doi.org/10.1108/13598540710776935>
- Fay, B. (1987). An alternative view: Interpretive social science. *Interpreting Politics*, 82–100.
- Feilzer, M. Y. (2010). Doing mixed methods research pragmatically: Implications for the rediscovery of pragmatism as a research paradigm. *Journal of Mixed Methods Research*, 4(1), 6–16. <https://doi.org/10.1177/1558689809349691>
- Finkelstein, A., & Savigni, A. (2001). A framework for requirements engineering for context-aware services. *First International Workshop From Software Requirements to Architectures-STRAW'01*, 36–41. <https://doi.org/10.1.1.25.16>
- García, A. J., & Simari, G. R. (2004). Defeasible Logic Programming An Argumentative Approach. *Theory and Practice of Logic Programming*, 4(2), 95–138.

- Garlan, D., & Shaw, M. (1992). An Introduction to Software Architecture. In *Advances in Software Engineering and Knowledge Engineering* (pp. 1–40). https://doi.org/10.1142/9789812798039_0001
- Gawer, A., & Cusumano, M. (2013). Industry Platforms and Ecosystem Innovation. *Journal of Product Innovation Management*, 31(3). <https://doi.org/10.1111/jpim.12105>
- Glass, G. V. (1976). Primary , Secondary , and Meta-Analysis of Research '. *Educational Researcher*, 5(10), 3–8.
- Glinz, M. (2007). On Non-Functional Requirements. *15th IEEE International Requirements Engineering Conference (RE 2007)*, 21–26. <https://doi.org/10.1109/RE.2007.45>
- Global Logistics Research Team at Michigan State University. (1995). *World Class Logistics: The Challenge of Managing Continuous Change*.
- Goldkuhl, G. (2012). Pragmatism vs interpretivism in qualitative information systems research. *European Journal of Information Systems*, 2(21), 135–146. <https://doi.org/10.1057/ejis.2011.54>
- Goles, T., & Hirschheim, R. (2000). The paradigm is dead , the paradigm is dead ... long live the paradigm: the legacy of Burrell and Morgan. *The International Journal of Management Science*, 28(3), 249–268. [https://doi.org/10.1016/S0305-0483\(99\)00042-0](https://doi.org/10.1016/S0305-0483(99)00042-0)
- Gong, Y., & Janssen, M. (2014). A Framework for Translating Legal Knowledge into Administrative Processes: Dynamic Adaption of Business Processes. *Software Engineering and Formal Methods 2012*, 204–211. https://doi.org/10.1007/978-3-642-54338-8_17
- Goodwin, C., & Duranti, A. (1992). Rethinking context: an introduction. In *Rethinking context: language as an interactive phenomenon* (pp. 1–42). Cambridge University Press.
- Graham, I. (2006). *Business Rules Management & Service Oriented Architecture*. John Wiley & Sons, Ltd.
- Greenberg, J. (2003). Metadata and the World Wide Web. In *Encyclopedia of Library and Information Science* (pp. 1876–1888). <https://doi.org/10.1081/E-ELIS>
- Greenberg, J. (2005). Understanding Metadata and Metadata Schemes. *Cataloging & Classification Quarterly*, 40(3–4), 17–36. <https://doi.org/10.1300/J104v40n03>
- Gregg, D. G. (2001). Understanding the Philosophical Underpinnings of Software Engineering Research in Information Systems. *Information Systems Frontiers*, 3(2), 169–183. <https://doi.org/10.1023/A:1011491322406>
- Gregor, S., & Jones, D. (2007). The Anatomy of a Design Theory. *Journal of the Association for Information Systems*, 8(5), 312–335. <https://doi.org/10.17705/1jais.00129>
- GS1. (2013). *An introduction to the global shipment identification number (GSIN)*.
- Gubbi, J., Buyya, R., Marusic, S., & Palaniswami, M. (2013). Internet of Things (IoT): A Vision, Architectural Elements, and Future Directions. *Future Generation Computer Systems*, 29(7), 1645–1660. <https://doi.org/10.1109/I-SMAC.2017.8058399>
- Hannaert, R. (2018). *Designing a Context-aware Decentralized Marketplace for Sensor Data*. Delft University of Technology.
- Hanseth, O., & Lyytinen, K. (2004). Theorizing about the design of Information Infrastructures: design kernel theories and principles. *Sprouts: Working Papers on Information Systems*, 4(4), 207–241. <https://doi.org/10.2752/146069206789377159>
- Harris, I., Wang, Y., & Wang, H. (2015). ICT in multimodal transport and technological trends: Unleashing potential for the future. *International Journal of Production Economics*, 159, 88–103. <https://doi.org/10.1016/j.ijpe.2014.09.005>
- Hart, P., & Saunders, C. (1997). Power and Trust: Critical Factors in the Adoption and Use of Electronic Data Interchange. *Organization Science*, 8(1), 23–42. <https://doi.org/10.1287/orsc.8.1.23>
- Haslhofer, B., & Klas, W. (2010). A Survey of Techniques for Achieving Metadata Interoperability. *ACM Computing Surveys*. <https://doi.org/10.1145/1667062.1667064>
- Henricksen, K., & Indulska, J. (2006). Developing context-aware pervasive computing applications: Models and approach. *Pervasive and Mobile Computing*, 2(1), 37–64. <https://doi.org/10.1016/j.pmcj.2005.07.003>

- Herschel, R., & Miori, V. M. (2017). Technology in Society Ethics & Big Data. *Technology in Society*, 49, 31–36. <https://doi.org/10.1016/j.techsoc.2017.03.003>
- Hertz, S., & Alfredsson, M. (2003). Strategic development of third party logistics providers. *Industrial Marketing Management*, 32(2), 139–149. [https://doi.org/10.1016/S0019-8501\(02\)00228-6](https://doi.org/10.1016/S0019-8501(02)00228-6)
- Hesketh, D. (2010). Weaknesses in the supply chain: who packed the box. *World Customs Journal*, 4(2), 3–20.
- Hevner, A. R. (2007). A Three Cycle View of Design Science Research. *Scandinavian Journal of Information Systems*, 19(2), 87–92.
- Hevner, A. R., & Chatterjee, S. (2004). Design Research in Information Systems. *Design Research in Information Systems*, 28, 75–105. <https://doi.org/10.1007/978-1-4419-5653-8>
- Hevner, A. R., March, S. T., Park, J., & Ram, S. (2004). Design Science in Information Systems Research. *Management Information Systems Quarterly*, 28(1), 75–105. <https://doi.org/10.2307/25148625>
- Hirschheim, R., & Klein, H. K. (1989). Four Paradigms of Information Systems Development. *Communications of the ACM*, 32(10), 1199–1216.
- Hofman, W. (2011). Supply chain visibility with linked open data for supply chain risk analysis. *1st Workshop on IT Innovations Enabling Seamless and Secure Supply Chains WITNESS 2011*, 20–31. <https://doi.org/10.3233/978-1-60750-785-7-77>
- Holden, M. T., & Lynch, P. (2004). Choosing the Appropriate Methodology: Understanding Research Philosophy. *The Marketing Review*, 4(4), 397–409. <https://doi.org/10.1362/1469347042772428>
- Hong, J.-Y., Suh, E.-H., & Kim, S.-J. (2009). Context-aware systems: A literature review and classification. *Expert Systems with Applications*, 36(4), 8509–8522. <https://doi.org/10.1016/j.eswa.2008.10.071>
- Hong, J., & Landay, J. (2001). An infrastructure approach to context-aware computing. *Human-Computer Interaction*, 16(2), 287–303. <https://doi.org/10.1207/S15327051HCI16234>
- Howell, K. E. (2012). *An introduction to the philosophy of methodology*. SAGE.
- Hulstijn, J., Hofman, W., Zomer, G., & Tan, Y.-H. (2016). Towards Trusted Trade-Lanes. *Electronic Government: Proceedings of the 15th IFIP WG 8.5 International Conference, EGOV 2016*, (Egov), 299–311. https://doi.org/10.1007/978-3-319-44421-5_24
- Iansiti, M., & Lakhani, K. R. (2017). The Truth About Blockchain. *Harvard Business Review*, 95(1), 118–127.
- IBM Corporation, & Maersk GTD. (2018). TradeLens Overview. Retrieved December 24, 2018, from TradeLens Documentation website: https://docs.tradelens.com/learn/tradelens_overview/
- Ilic, A., Staake, T., & Fleisch, E. (2009). Using Sensor Information to Reduce the Carbon Footprint of Perishable Goods Alexander. *IEEE Pervasive Computing*, 8(1), 1–17. <https://doi.org/10.1109/MPRV.2009.20>
- Janssen, M., & Smeele, F. (2013). *JUridical and context-aware Sharing of informaTion for ensuring compliance (JUST)*. NWO.
- Järvinen, P. (2007). Action Research is Similar to Design Science. *Quality & Quantity*, 41, 37–54. <https://doi.org/10.1007/s11135-005-5427-1>
- Jeffery, K. G. (2000). Metadata: The future of information systems. *12th Conference on Advanced Information Systems Engineering*. London: Springer Verlag.
- Jensen, T. (2017). *Shipping information pipeline: An information infrasstructure to improve international containerized shipping*. Copenhagen Business School.
- Jensen, T., Bjørn-Andersen, N., & Vatrapu, R. (2014). Avocados crossing borders: The missing common information infrastructure for international trade. *Culture in International Context*, 15–24. <https://doi.org/10.1145/2631488.2631500>
- Jensen, T., & Tan, Y.-H. (2015). Key Design Properties for Shipping Information Pipeline. In M. Janssen, M. Mäntymäki, J. Hidders, B. Klievink, & D. Hutchison (Eds.), *Open and Big Data Management and Innovation*. https://doi.org/10.1007/978-3-319-25013-7_40

- Jensen, T., Tan, Y.-H., & Bjørn-Andersen, N. (2014). Unleashing the IT potential in the complex digital business ecosystem of international trade: The case of fresh fruit import to European Union. *BLED 2014 Proceedings*.
- Jensen, T., & Vatrapu, R. (2015). Ships & roses: A revelatory case study of affordances in international trade. *ECIS 2015 Proceedings*, 1–18. <https://doi.org/10.18151/7217370>
- Johnson, R. B., & Onwuegbuzie, A. J. (2004). Mixed Methods Research: A Research Paradigm Whose Time Has Come. *Educational Researcher*, 33(7), 14–26. <https://doi.org/10.3102/0013189X033007014>
- Johnston, H. R., & Vitale, M. R. (1988). Creating Competitive Advantage With Interorganizational Information Systems. *Management Information Systems Quarterly*, 12(2), 153–165.
- Kaltz, J. W., Ziegler, J., & Lohmann, S. (2005). Context-aware web engineering: Modeling and applications. *Revue d'Intelligence Artificielle*, 19(3), 439–458. <https://doi.org/10.3166/ria.19.439-458>
- Karampetsou, A. (2016). Container Information & Privacy Concerns: Opening the Pandora's box? Legal challenges of a Business-to-Customs Information Sharing with regard to Containerized Cargo. *Current Issues in Maritime & Transport Law*, 1–17. Bonomo Editore, Bologna.
- Kettinger, W. J., & Li, Y. (2010). The infological equation extended: towards conceptual clarity in the relationship between data, information and knowledge. *European Journal of Information Systems*, 19(4), 409–421. <https://doi.org/10.1057/ejis.2010.25>
- Khedo, K. K. (2006). Context-Aware Systems for Mobile and Ubiquitous Networks. *Proceedings of the International Conference on Networking, International Conference on Systems and International Conference on Mobile Communications and Learning Technologies (ICNICONSMCL'06)*, 123–130. <https://doi.org/10.1109/ICNICONSMCL.2006.68>
- Kiayias, A., Russell, A., David, B., & Oliynykov, R. (2017). Ouroboros: A provably secure proof-of-stake blockchain protocol. *Advances in Cryptology – CRYPTO 2017*, 357–388. https://doi.org/10.1007/978-3-319-63688-7_12
- Klein, H. K., & Myers, M. D. (1999). A Set of Principles for Conducting and Evaluating Interpretive Field Studies in Information Systems. *Management Information Systems Quarterly*, 23(1), 67–94. <https://doi.org/10.2307/249410>
- Klievink, B., Aldewereld, H., & Tan, Y.-H. (2014). Establishing Information Infrastructures for International Trade: Discussing the Role and Governance of Port-Community Systems. *5th International Conference on Information Systems, Logistics and Supply Chain (ILS2014)*, 1–10. Dinalog.
- Klievink, B., Janssen, M., & Tan, Y.-H. (2012). A Stakeholder Analysis of Business-to-Government Information Sharing. *International Journal of Electronic Government Research*, 8(4), 54–64. <https://doi.org/10.4018/jegr.2012100104>
- Klievink, B., & Lucassen, I. (2013). Facilitating adoption of international information infrastructures: a Living Labs approach. *International Conference on Electronic Government*, 250–261. https://doi.org/10.1007/978-3-642-40358-3_21
- Klievink, B., van Stijn, E., Hesketh, D., Aldewereld, H., Overbeek, S., Heijmann, F., & Tan, Y.-H. (2012). Enhancing Visibility in International Supply Chains: The Data Pipeline Concept. *International Journal of Electronic Government Research (IJEGR)*, 8(4), 14–33. <https://doi.org/10.4018/jegr.2012100102>
- Kolos-Mazuryk, L., Poullisse, G.-J., & van Eck, P. (2005). Requirements engineering for pervasive services. *OOP-SLA'05 Workshop on Creating Software for Pervasive Services*, 1–5.
- Korpela, K., Hallikas, J., & Dahlberg, T. (2017). Digital Supply Chain Transformation toward Blockchain Integration. *Proceedings of the 50th Hawaii International Conference on System Sciences*, 4182–4191. <https://doi.org/10.24251/HICSS.2017.506>
- Kothari, C. R. (2004). *Research Methodology: Methods and Techniques*. New Age International.
- Kowalski, R. (1974). Predicate Logic as a Programming Language. *IFIP Congress*, 569–574.
- Kuechler, B., & Vaishnavi, V. (2008). On theory development in design science research: anatomy of a research project. *European Journal of Information Systems*, 17(5), 489–504.

- <https://doi.org/10.1057/ejis.2008.40>
- Kuechler, W., Vaishnavi, V. K., & Petter, S. (2005). The Aggregate General Design Cycle as a Perspective on the Evolution of Computing Communities of Interest. *Computing Letters*, 1(3), 123–128. <https://doi.org/10.1163/1574040054861221>
- La Londe, B. J., & Masters, J. M. (1994). Emerging Logistics Strategies: Blueprints for the Next Century. *International Journal of Physical Distribution & Logistics Management*, 24(7), 35–47. <https://doi.org/10.1108/09600039410070975>
- Lambert, D. M., & Cooper, M. C. (2000). Issues in Supply Chain Management. *Industrial Marketing Management*, 29, 65–83.
- Lambert, D. M., Cooper, M. C., & Pagh, J. D. (1998). Supply Chain Management: Implementation Issues and Research Opportunities. *International Journal of Logistics Management*, Vol. 9, pp. 1–19. <https://doi.org/10.1108/09574099810805807>
- Lamsfus, C., Wang, D., Alzua-Sorzabal, A., & Xiang, Z. (2015). Going Mobile: Defining Context for On-the-Go Travelers. *Journal of Travel Research*, 54(6), 691–701. <https://doi.org/10.1177/0047287514538839>
- Laney, D. (2001). 3D data management: Controlling data volume, velocity and variety. *META Group Research Note*, 6(70).
- Lee, A. S., & Baskerville, R. L. (2003). Generalizing Generalizability in Information Systems Research. *Information Systems Research*, 14(3), 221–243.
- Lee, A. S., & Hubona, G. S. (2009). A Scientific Basis for Rigor in Information Systems Research. *Management Information Systems Quarterly*, 33(2), 237–262.
- Lee, A. S., & Nickerson, J. V. (2010). Theory as a case of design: Lessons for design from the philosophy of science. *Proceedings of the 43rd Hawaii International Conference on System Sciences*, 1–8. <https://doi.org/10.1109/HICSS.2010.484>
- Lee, A. S., Thomas, M., & Baskerville, R. L. (2015). Going back to basics in design science: From the information technology artifact to the information systems artifact. *Information Systems Journal*, 25(1), 5–21. <https://doi.org/10.1111/isj.12054>
- Lee, H. L., Padmanabhan, V., & Whang, S. (1997a). Information Distortion in a Supply Chain: The Bullwhip Effect. *Management Science*, 43(4), 546–558. <https://doi.org/10.1287/mnsc.43.4.546>
- Lee, H. L., Padmanabhan, V., & Whang, S. (1997b). The bullwhip effect in supply chains. *Sloan Management Review*, 38(3), 93–102. <https://doi.org/10.1016/j.jipe.2008.08.035>
- Lee, H. L., & Whang, S. (2000). Information sharing in a supply chain. *International Journal of Manufacturing Technology*, 1(1), 79–93.
- Levi, M. D. (2005). *International Finance* (4th ed.). Routledge.
- Levinson, M. (2010). The World the Box Made. In *The Box: How the Shipping Container Made the World Smaller and the World Economy Bigger* (pp. 1–15). Princeton University Press.
- Lifschitz, V. (1996). Foundations of logic programming. *Principles of Knowledge Representation*, 3, 69–127.
- Lincoln, Y. S., & Guba, E. G. (1985). Postpositivism and the naturalistic paradigm. *Naturalistic Inquiry*, 14–46.
- Liu, K. (2000). *Semiotics in information systems engineering*. Cambridge University Press.
- López-Pintado, O., García-Bañuelos, L., Dumas, M., & Weber, I. (2017). Caterpillar: A blockchain-based business process management system. *Proceedings of the BPM Demo Track and BPM Dissertation Award Co-Located with 15th International Conference on Business Process Modeling*, 1–5.
- Losee, R. M. (1997). A Discipline Independent Definition of Information. *Journal of the American Society for Information Science*, 48(3), 254–269. [https://doi.org/10.1002/\(SICI\)1097-4571\(199703\)48:3<254::AID-ASIS6>3.0.CO;2-W](https://doi.org/10.1002/(SICI)1097-4571(199703)48:3<254::AID-ASIS6>3.0.CO;2-W)
- Lucassen, I., Klievink, B., Griffioen, H., & Commission, E. (2010). *Cassandra – WP400 – Asia-NL/UK trade lane Living Lab report*.
- Luu, L., Narayanan, V., Baweja, K., Zheng, C., Gilbert, S., & Saxena, P. (2015). SCP: A Computationally-Scalable Byzantine Consensus Protocol For Blockchains. *IACR*

- Cryptology EPrint Archive*, 1–16.
- Malenstein, J., Schewe, W., Zomer, G., Klievink, B., Nijdam, M., & Visscher, W. (2014). *CASSANDRA - D6.3 Final Protocol*.
- March, S. T., & Smith, G. F. (1995). Design and natural science research on information technology. *Decision Support Systems*, 15(4), 251–266. [https://doi.org/10.1016/0167-9236\(94\)00041-2](https://doi.org/10.1016/0167-9236(94)00041-2)
- March, S. T., & Storey, V. C. (2008). Design science in the information systems discipline: An introduction to the special issue on design science research. *Management Information Systems Quarterly*, 32(4), 725–730. <https://doi.org/10.2307/25148869>
- Markus, M. L., Majchrzak, A., & Gasser, L. (2002). A design theory for systems that support emergent knowledge processes. *Management Information Systems Quarterly*, 26(3), 179–212. <https://doi.org/10.2307/4132330>
- Marshall, P., Kelder, J., & Perry, A. (2005). Social constructionism with a twist of pragmatism: a suitable cocktail for information systems research. *16th Australasian Conference on Information Systems*.
- Massias, H., Serret Avila, X., & Quisquater, J.-J. (1999). Design of a secure timestamping service with minimal trust requirement. *20th Symposium on Information Theory in the Benelux*. <https://doi.org/10.1.1.13.6228>
- Matthias, H., Stephen, D., Jan, H., & Johanna, S. (2005). Supply Chain Collaboration: Making Sense of the Strategy Continuum. *European Management Journal*, 23(2), 170–181. <https://doi.org/10.1016/j.emj.2005.02.008>
- McAfee, A., & Brynjolfsson, E. (2012). Big data: the management revolution. *Harvard Business Review*, 90(10), 60–68. <https://doi.org/10.1007/s12599-013-0249-5>
- Mendling, J., Weber, I., van der Aalst, W., vom Brocke, J., Cabanillas, C., Daniel, F., ... Zhu, L. (2017). Blockchains for Business Process Management - Challenges and Opportunities. *ACM Transactions on Management Information Systems (TMIS)*, 9(1), 1–16. <https://doi.org/10.1145/3183367>
- Mentzer, J. T., Dewitt, W., Keebler, J. S., Min, S., Nix, N. W., Smith, C. D., & Zacharia, Z. G. (2001). Defining Supply Chain Management. *Journal of Business Logistics*, 22(2), 1–25. <https://doi.org/10.1002/j.2158-1592.2001.tb00001.x>
- Merkle, R. C. (1987). A Digital Signature Based on a Conventional Encryption Function. In C. Pomerance (Ed.), *Advances in Cryptology — CRYPTO '87*. https://doi.org/10.1007/3-540-48184-2_32
- Mertens, D. M. (2003). Mixed methods and the politics of human research: The transformative-emancipatory perspective. *Handbook of Mixed Methods in Social and Behavioral Research*, 135–164.
- Meuser, M., & Nagel, U. (2009). The Expert Interview and Changes in Knowledge Production. In A. Bogner, B. Littig, & W. Metz (Eds.), *Interviewing experts* (pp. 17–42). Palgrave Macmillan, London.
- Min, H., & Zhou, G. (2002). Supply chain modeling: past, present and future. *Computers & Industrial Engineering*, 43(1–2), 231–249. [https://doi.org/10.1016/S0360-8352\(02\)00066-9](https://doi.org/10.1016/S0360-8352(02)00066-9)
- Mingers, J. (2001). Combining IS Research Methods: Towards a Pluralist Methodology. *Information Systems Research*, 12(3), 240–259. <https://doi.org/10.1287/isre.12.3.240.9709>
- Mingers, J. (2002). The Long and Winding Road: Getting Papers Published in Top Journals. *Communications of the Association for Information Systems*, 8, 330–339. <https://doi.org/http://aisel.aisnet.org/cais/vol8/iss1/22>
- Mohanty, H. (2015). Big Data: An Introduction. In H. Mohanty, P. Bhuyan, & D. Chenthati (Eds.), *Big data a Primer* (pp. 1–28). Springer.
- Monroe, R. T., Kompanek, A., Melton, R., & Garlan, D. (1997). Architectural styles, design patterns, and objects. *IEEE Software*, 14(1), 43–52. <https://doi.org/10.1109/52.566427>
- Morgan, D. L. (2007). Methodological Implications of Combining Qualitative and Quantitative Methods. *Journal of Mixed Methods Research*, 1(1), 48–76. <https://doi.org/10.1177/2345678906292462>

- Morgan, D. L. (2014). Pragmatism as a Paradigm for Social Research. *Qualitative Inquiry*, 20(8), 1045–1053. <https://doi.org/10.1177/1077800413513733>
- Morgan, G. (1983). *Beyond Method: Strategies for Social Research*. SAGE.
- Myers, M. D. (1997). Qualitative Research in Information Systems. *Management Information Systems Quarterly*, 21(2).
- Nakamoto, S. (2008). *Bitcoin: A Peer-to-Peer Electronic Cash System* (p. 9). p. 9. <https://doi.org/10.1007/s10838-008-9062-0>
- Narayanan, A., Bonneau, J., Felten, E., Miller, A., & Goldfeder, S. (2016). *Bitcoin and cryptocurrency technologies: A comprehensive introduction*. Princeton University Press.
- Novick, G. (2008). Is there a bias against telephone interviews in qualitative research? *Research in Nursing and Health*, 31(4), 391–398. <https://doi.org/10.1002/nur.20259>
- Nunamaker, J. F. J., Chen, M., & Purdin, T. D. M. (1991). Systems development in information systems research. *Journal of Management Information Systems*, 7(3), 89–106. <https://doi.org/10.1109/HICSS.1990.205401>
- Nuseibeh, B., & Easterbrook, S. (2000). Requirements engineering: a roadmap. *Proceedings of the Conference on The Future of Software Engineering - ICSE '00*, 35–46. <https://doi.org/10.1145/336512.336523>
- Offermann, P., Blom, S., Schönherr, M., & Bub, U. (2010). Artifact Types in Information Systems Design Science – A Literature Review. *Management Information Systems Quarterly*, 61(5), 77–92. https://doi.org/10.1007/978-3-642-13335-0_6
- Ølnes, S., Ubacht, J., & Janssen, M. (2017). Blockchain in government: Benefits and implications of distributed ledger technology for information sharing. *Government Information Quarterly*, 34(3), 355–364. <https://doi.org/10.1016/j.giq.2017.09.007>
- Orlikowski, W. J., & Baroudi, J. J. (1991). Studying Information Technology in Organizations: Research Approaches and Assumptions. *Information Systems Research*, 2(1), 1–28. <https://doi.org/10.1287/isre.2.1.1>
- Ormerod, R. (2006). The history and ideas of pragmatism. *Journal of the Operational Research Society*, Vol. 57, pp. 892–909. <https://doi.org/10.1057/palgrave.jors.2602065>
- Ouaddah, A., Abou Elkalam, A., & Ait Ouahman, A. (2016). FairAccess: a new Blockchain-based access control framework for the Internet of Things. *Security and Communication Networks*, 9(18), 5943–5964. <https://doi.org/10.1002/sec.1748>
- Ouaddah, A., Mousannif, H., Abou, A., & Ait, A. (2017). Access control in the Internet of Things : Big challenges and new opportunities. *Computer Networks*, 112, 237–262. <https://doi.org/10.1016/j.comnet.2016.11.007>
- Overbeek, S., Janssen, M., & Tan, Y.-H. (2012). An Event-Driven Architecture for Integrating Information, Processes and Services in a Plastic Toys Supply Chain. *International Journal of Cooperative Information Systems*, 21(04), 343–381. <https://doi.org/10.1142/S0218843012500062>
- Overbeek, S., Klievink, B., Hesketh, D., Heijmann, F., & Tan, Y.-H. (2011). A Web-based data pipeline for compliance in international trade. *WITNESS 2011*, 32–48.
- Oxford Dictionaries. (2018). Context. Retrieved July 27, 2018, from Oxford Dictionaries website: <https://en.oxforddictionaries.com/definition/context>
- Papazoglou, M. P., & van den Heuvel, W.-J. (2007). Service oriented architectures: Approaches, technologies and research issues. *VLDB Journal*, 16(3), 389–415. <https://doi.org/10.1007/s00778-007-0044-3>
- Peppers, K., Tuunanen, T., Rothenberger, M. A., & Chatterjee, S. (2007). A design science research methodology for information systems research. *Journal of Management Information Systems*, 24(3), 45–77. <https://doi.org/10.2753/MIS0742-1222240302>
- Perry, D. E., & Wolf, A. L. (1992). Foundations for the study of software architecture. *ACM SIGSOFT Software Engineering Notes*, 17(4), 40–52. <https://doi.org/10.1145/141874.141884>
- Perttunen, M., Riekk, J., & Lassila, O. (2009). Context Representation and Reasoning in Pervasive Computing. *International Journal of Multimedia and Ubiquitous Engineering*, 4(4), 1–28.

- Pilkington, M. (2016). Blockchain technology: Principles and applications. In F. X. Olleros & M. Zhegu (Eds.), *Research Handbook on Digital Transformations* (pp. 227–253). Edward Elgar Publishing.
- Pisanias, N., & Willcocks, L. (1999). Understanding slow Internet adoption: “infomediation” in ship-broking markets. *Journal of Information Technology*, 14, 399–413.
- Power, M. (2007). *Organized uncertainty: Designing a world of risk management*. Oxford University Press.
- Pries-Heje, J., Baskerville, R., & Venable, J. (2008). Strategies for design science research evaluation. *ECIS 2008 Proceedings*, 1–12. <https://doi.org/10.1177/1933719108329095>
- Pruksasri, P., van den Berg, J., & Hofman, W. (2014). Global Monitoring of Dynamic Information Systems A Case Study in the International Supply Chain. *Computer Science and Engineering Conference (ICSEC)*.
- Pruksasri, P., van den Berg, J., Hofman, W., & Tan, Y.-H. (2016). Enhancing Process Visibility of the Supply Chain “Data Pipeline.” *24 Ecti Transactions on Computer and Information Technology*, 10, 15–25.
- Qiu, L., Chang, L., Lin, F., & Shi, Z. (2007). Context optimization of AI planning for semantic Web services composition. *Service Oriented Computing and Applications*, 1(2), 117–128. <https://doi.org/10.1007/s11761-007-0010-3>
- Rafaeli, S., & Raban, D. R. (2005). Information sharing online: a research challenge. *International Journal of Knowledge and Learning*, 1(1/2), 62–79. <https://doi.org/10.1504/IJKL.2005.006251>
- Randell, B. (1975). System Structure for Software Fault Tolerance. *Ieee Transactions on Software Engineering*, SE-1(2), 220–232.
- Rettler, B., & Bailey, A. M. (2017). Object. In *The Stanford Encyclopedia of Philosophy* (pp. 1–22). Retrieved from <https://plato.stanford.edu/archives/win2017/entries/object/>
- Roman, R., Zhou, J., & Lopez, J. (2013). On the features and challenges of security and privacy in distributed internet of things. *Computer Networks*, 57(10), 2266–2279. <https://doi.org/10.1016/j.comnet.2012.12.018>
- Romme, A. G. L. (2003). Making a Difference: Organization as Design. *Organization Science*, 14(5), 558–573. <https://doi.org/10.1287/orsc.14.5.558.16769>
- Rood, M. A. (1994). Enterprise Architecture: Definition, Content, and Utility. *Enabling Technologies: Infrastructure for Collaborative Enterprises*, 106–111.
- Rorty, R. (1996). Introduction: Relativism: Finding and Making. In *Philosophy and Social Hope* (pp. XVII–XXXII). Penguin Books.
- Ross, A. M., Rhodes, D. H., & Hastings, D. E. (2008). Defining changeability: Reconciling flexibility, adaptability, scalability, modifiability, and robustness for maintaining system lifecycle value. *Systems Engineering*, 11(3), 246–262. <https://doi.org/10.1002/sys.20098>
- Rosson, M. B., & Carroll, J. M. (2002). Scenario-Based Design. In *The human-computer interaction handbook: fundamentals, evolving technologies and emerging applications* (pp. 1032–1050). <https://doi.org/10.1016/j.jbi.2011.07.004>
- Rukanova, B., Bjørn-Andersen, N., Ipenburg, F. van, Klein, S., Smit, G., & Tan, Y.-H. (2011). Introduction. In Y.-H. Tan, N. Bjørn-Andersen, S. Klein, & B. Rukanova (Eds.), *Accelerating Global Supply Chains with IT-Innovation: ITAIDE Tools and Methods* (pp. 3–30). <https://doi.org/10.1007/978-3-642-15669-4>
- Rukanova, B., Henningsson, S., Henriksen, H. Z., & Tan, Y.-H. (2018). Digital Trade Infrastructures : A Framework for Analysis. *Complex Systems Informatics and Modeling Quarterly*, (14), 1–21. <https://doi.org/10.7250/csimq.2018-14.01>
- Rukanova, B., Huiden, R., & Tan, Y.-H. (2017). Coordinated Border Management Through Digital Trade Infrastructures and Trans- National Government Cooperation: The FloraHolland Case. *Electronic Government: Proceedings of IFIP WG 8.5 International Conference, EGOV2017, 10428*, 240–252. <https://doi.org/10.1007/978-3-319-64677-0>
- Runeson, P., & Höst, M. (2009). Guidelines for conducting and reporting case study research in software engineering. *Empirical Software Engineering*, 14(2), 131–164.

- <https://doi.org/10.1007/s10664-008-9102-8>
- Saeed, A., & Waheed, T. (2012). An extensive survey of context-aware middleware architectures. *American Journal of Computer Architecture*, 1(3), 51–56. <https://doi.org/10.5923/j.ajca.20120103.02>
- Sagaya Priya, K. S., & Kalpana, Y. (2016). A review on context modelling techniques in context aware computing. *International Journal of Engineering and Technology*, 8(1), 429–433.
- Salber, D., Dey, A. K., & Abowd, G. D. (1999). The context toolkit: Aiding the development of context-enabled applications. *Conference on Human Factors in Computing Systems - Proceedings*, 434–441. <https://doi.org/10.1145/302979.303126>
- Saveen, A., & Monfared, R. P. (2016). Blockchain Ready Manufacturing Supply Chain Using Distributed Ledger. *International Journal of Research in Engineering and Technology*, 05(09), 1–10. <https://doi.org/10.15623/ijret.2016.0509001>
- Sawyer, P., Bencomo, N., Whittle, J., Letie, E., & Finkelstein, A. (2010). Requirements-aware systems: A research agenda for RE for self-adaptive systems. *Proceedings of the 2010 18th IEEE International Requirements Engineering Conference, RE2010*, 95–103. <https://doi.org/10.1109/RE.2010.21>
- Schilit, B., Adams, N., & Want, R. (1994). Context-aware computing applications. *Proceedings of the 1994 First Workshop on Mobile Computing Systems and Applications*, 85–90. <https://doi.org/10.1109/MCSA.1994.512740>
- Schilit, B., & Theimer, M. M. (1994). Disseminating Active MapInformation to Mobile Hosts. *IEEE Network*, 8(5), 22–32. <https://doi.org/10.1109/65.313011>
- Schmohl, R., & Baumgarten, U. (2008). A Generalized Context-aware Architecture in Heterogeneous Mobile Computing Environments. *The Fourth International Conference on Wireless and Mobile Communications*, 118–124. <https://doi.org/10.1109/ICWMC.2008.59>
- Schuurman, N., Deshpande, A., & Allen, D. M. (2008). Data integration across borders: a case study of the Abbotsford-Sumas aquifer (British Columbia/Washington State). *Journal of the American Water Resources Association*, 44(4), 921–934.
- Schweizer, A., Schlatt, V., Urbach, N., & Fridgen, G. (2017). Unchaining Social Businesses - Blockchain as the Basic Technology of a Crowdlending Platform. *38th International Conference on Information Systems*, 4801(September), 1–21.
- Schwinger, W., Grün, C., Proll, B., Retschitzegger, W., & Schauerhuber, A. (2005). Context-awareness in Mobile Tourism Guides – A Comprehensive Survey. In *Technical report*.
- Seawright, J., & Gerring, J. (2008). Case Selection Techniques in Case Study Research A Menu of Qualitative and Quantitative Options. *Source: Political Research Quarterly*, 61226(2), 294–308. <https://doi.org/10.1177/1065912907313077>
- Shishkov, B. (2019). Tuning the Behavior of Context-Aware Applications, Using Semiotic Norms and Bayesian Modeling to Establish the User Situation. In B. Shishkov (Ed.), *Business Modeling and Software Design. BMSD 2019. Lecture Notes in Business Information Processing, vol 356*. Springer. Springer, Cham.
- Shishkov, B., Larsen, J. B., Warnier, M., & Janssen, M. (2018). Three Categories of Context-Aware Systems. In B. Shishkov (Ed.), *International Symposium on Business Modeling and Software Design* (Vol. 319, pp. 185–202). https://doi.org/10.1007/978-3-319-94214-8_12
- Shishkov, B., & van Sinderen, M. (2007). Model-Driven Design of Context-Aware Applications. *Proceedings of the Ninth International Conference on Enterprise Information Systems, ICEIS*, 105–113.
- Shishkov, B., van Sinderen, M., & Verbraeck, A. (2009). Towards Flexible Inter-enterprise Collaboration : A Supply Chain Perspective. *Enterprise Information Systems. ICEIS 2009*, 513–527. https://doi.org/10.1007/978-3-642-01347-8_43
- Silveira, P., Rodriguez, C., Birukou, A., Casati, F., Daniel, F., D’Andrea, V., ... Taheri, Z. (2012). Aiding compliance governance in service-based business processes. In *Handbook of Research on Service-Oriented Systems and Non-Functional Properties: Future Directions* (pp. 524–548). <https://doi.org/10.4018/978-1-61350-432-1.ch022>
- Simon, H. A. (1996). The sciences of the artificial. In *Computers & Mathematics with Applications*

- (Vol. 33). [https://doi.org/10.1016/S0898-1221\(97\)82941-0](https://doi.org/10.1016/S0898-1221(97)82941-0)
- Sitou, W., & Spanfelner, B. (2007). Towards requirements engineering for context adaptive systems. *Proceedings - International Computer Software and Applications Conference*, 2, 593–598. <https://doi.org/10.1109/COMPSAC.2007.223>
- Smeele, F. (2009). Bill of Lading Contracts under European National Laws: Civil law approaches to explaining the legal position of the consignee under bills of lading. In D. R. Thomas (Ed.), *The Evolving Law and Practice of Voyage Charterparties* (pp. 251–280). London: Informa.
- Smeele, F. (2010). The Maritime Performing Party in the Rotterdam Rules 2009. *European Journal of Commercial Contract Law*, 1–23. Retrieved from http://repub.eur.nl/res/pub/23175/maritime_performing.pdf
- Smeele, F. (2016). Legal Conceptualisations of the Freight Forwarder: some Comparative Reflections on the Disunified Law of Forwarding. *The Journal of International Maritime Law*, 21, 445–459.
- Software Engineering Standards Committee of the IEEE Computer Society. (2000). 1471-2000 - IEEE Recommended Practice for Architectural Description for Software-Intensive Systems. In *October*. <https://doi.org/10.1109/IEEESTD.2000.91944>
- Sterling, L., & Shapiro, E. (1999). *Advanced Programming techniques: The Art of Prolog* (2nd ed.). The MIT Press.
- Stopford, M. (2009). The Organization of the Shipping Market. In *Maritime Economics* (pp. 47–90). Taylor and Francis Group.
- Strang, T., & Linnhoff-Popien, C. (2004). A Context Modeling Survey. *Workshop on Advanced Context Modelling, Reasoning and Management, UbiComp 2004 - The Sixth International Conference on Ubiquitous Computing*, 1–8. <https://doi.org/10.1.1.2.2060>
- Sun, S., & Yen, J. (2005). Information Supply Chain: A Unified Framework for Information-Sharing. *Intelligence and Security Informatics*, 422–428. Springer, Berlin, Heidelberg.
- Sundin, O., & Johannisson, J. (2005). Pragmatism, neo-pragmatism and sociocultural theory: Communicative participation as a perspective in LIS. *Journal of Documentation*, 61(1), 23–43. <https://doi.org/10.1108/00220410510577998>
- Swan, M. (2015). *Blockchain: Blueprint for a New Economy*. O'Reilly Media.
- Tan, Y.-H., Rukanova, B., van Engelenburg, S., Janssen, M., & Ubacht, J. (n.d.). Transforming Information Sharing Through a Blockchain-based Digital Trade Infrastructure. *Manuscript in Preparation*.
- Tasca, P., & Tessone, C. J. (2017). Taxonomy of Blockchain Technologies . Principles of Identification and Classification. *ArXiv Preprint ArXiv:1708.04872*. <https://doi.org/10.2139/ssrn.2977811>
- Tellis, W. (1997). Application of a Case Study Methodology. *The Qualitative Report*, 3(3), 1–19.
- The British Broadcasting Corporation. (2012). Fire-hit MSC Flaminia “safe” to be towed. Retrieved August 21, 2018, from BBC News website: <http://www.bbc.com/news/uk-england-cornwall-19448577>
- The Commission Of The European Communities. *Annex 30A of the CCIP Data requirements for entry and exit summary declarations and for simplified procedures*. , (2006).
- The Commission Of The European Communities. (2006b). EU Regulation 1875/2006 of 18 December 2006 amending Regulation (EEC) No 2454/93 laying down provisions for the implementation of Council Regulation (EEC) No 2913/92 establishing the Community Customs Code. *Official Journal of the European Union*.
- The European Commission. Commission Implementing Regulation (EU) 2015/2447 of 24 November 2015 laying down detailed rules for implementing certain provisions of Regulation (EU) No 952/2013 of the European Parliament and of the Council laying down the Union Customs Code. , Official Journal of the European Union § (2015).
- The European Commission. (2018). The Union Customs Code (UCC) - Introduction. Retrieved November 25, 2018, from https://ec.europa.eu/taxation_customs/business/union-customs-code/ucc-introduction_en
- The European Commission Directorate-General Taxation and Customs Union. (2016). Authorised

- Economic Operators Guidelines. Retrieved December 23, 2018, from https://ec.europa.eu/taxation_customs/sites/taxation/files/resources/documents/customs/policy_issues/customs_security/ao_guidelines_en.pdf
- The European Parliament and the Council of the European Union. Regulation (EU) No 952/2013 of the European Parliament and of the Council of 9 October 2013 laying down the Union Customs Code (recast). , Official Journal of the European Union § (2013).
- The European Parliament and the Council of the European Union. Directive (EU) 2016/943 of the European Parliament and of the Council of 8 June 2016 on the protection of undisclosed know-how and business information (trade secrets) against their unlawful acquisition, use and disclosure. , Official Journal of the European Union § (2016).
- Tian, F. (2016). An Agri-food Supply Chain Traceability System for China Based on RFID & Blockchain Technology. *13th International Conference on Service Systems and Service Management (ICSSSM)*, 1–6. <https://doi.org/10.1109/ICSSSM.2016.7538424>
- Tilson, D., Lyytinen, K., & Sørensen, C. (2010). Digital infrastructures: The missing IS research agenda. *Information Systems Research*, 21(4), 748–759. <https://doi.org/10.1287/isre.1100.0318>
- Tiwana, A., Konsynski, B., & Bush, A. a. A. (2010). Platform evolution: Coevolution of platform architecture, governance, and environmental dynamics. *Information Systems Research*, 21(4), 675–687. <https://doi.org/10.1287/isre.1100.0323>
- Tsay, A. A., Nahmias, S., & Agrawal, N. (1999). Modeling Supply Chain Contracts: A Review. In S. Tayur, R. Ganeshan, & M. Magazine (Eds.), *Quantitative Models for Supply Chain Management* (pp. 299–336). https://doi.org/https://doi.org/10.1007/978-1-4615-4949-9_10
- Turner III, D. W. (2010). The Qualitative Report Qualitative Interview Design: A Practical Guide for Novice Investigators. *The Qualitative Report*, 15(3), 754–760.
- Ulrich, W. (2007). Philosophy for professionals: towards critical pragmatism. *Journal of the Operational Research Society*, 58, 1109–1117. <https://doi.org/10.1057/palgrave.jors.2602336>
- UN/CEFACT. (2005). Recommendation and Guidelines on establishing a Single Window: To Enhance the Efficient Exchange of Information Between Trade and Government. *Recommendation 33*.
- Urciuoli, L., Hintsä, J., & Ahokas, J. (2013). Drivers and barriers affecting usage of e-Customs - A global survey with customs administrations using multivariate analysis techniques. *Government Information Quarterly*, 30(4), 473–485. <https://doi.org/10.1016/j.giq.2013.06.001>
- Vaishnavi, V. K. (2007). Introduction to Design Science Research in Information and Communication Technology. In *Design Science Research Methods and Patterns* (pp. 7–30). Taylor & Francis Group, LLC.
- Vaismoradi, M., Turunen, H., & Bondas, T. (2013). Content analysis and thematic analysis: Implications for conducting a qualitative descriptive study. *Nursing and Health Sciences*, 15(3), 398–405. <https://doi.org/10.1111/nhs.12048>
- van Baalen, P., Zuidwijk, R., & van Nunen, J. (2009). Port Inter-Organizational Information Systems: Capabilities to Service Global Supply Chains. *Foundations and Trends in Technology, Information and Operations Management*, 2(2–3), 81–241. <https://doi.org/10.1561/02000000008>
- van der Aalst, W. M. P., De Masellis, R., Di Francescomarino, C., & Ghidini, C. (2017). Learning Hybrid Process Models From Events: Process Discovery Without Faking Confidence. *International Conference on Business Process Management*, (September). <https://doi.org/10.1007/978-3-319-65000-5>
- Van Emden, M. H. (1975). Programming with resolution logic. In *Report CS-75-30*.
- van Engelenburg, S., Janssen, M., & Klievink, B. (n.d.). A context-aware information sharing architecture based on blockchain technology. *Manuscript in Preparation*.
- van Engelenburg, S., Janssen, M., & Klievink, B. (2015). Design of a Business-to-Government Information Sharing Architecture Using Business Rules. In *Software Engineering and*

- Formal Methods* (Vol. 9509, pp. 124–138). <https://doi.org/10.1007/978-3-319-15201-1>
- van Engelenburg, S., Janssen, M., & Klievink, B. (2017a). Design of a software architecture supporting business-to-government information sharing to improve public safety and security. *Journal of Intelligent Information Systems*. <https://doi.org/10.1007/s10844-017-0478-z>
- van Engelenburg, S., Janssen, M., & Klievink, B. (2017b). What belongs to context? A definition, a criterion and a method for deciding on what context-aware systems should sense and adapt to. In A. Cerone & M. Roveri (Eds.), *Software Engineering and Formal Methods 2017: Vol. 10729 LNCS* (pp. 101–116). https://doi.org/10.1007/978-3-319-74781-1_8
- van Engelenburg, S., Janssen, M., & Klievink, B. (2018). A Blockchain Architecture for Reducing the Bullwhip Effect. In B. Shishkov (Ed.), *BMSD 2018, LNBIP 319* (pp. 69–82). <https://doi.org/10.1007/978-3-319-94214-8>
- van Engelenburg, S., Janssen, M., & Klievink, B. (2019). Designing context-aware systems: a method for understanding and analysing context in practice. *Journal of Logical and Algebraic Methods in Programming*, 103, 79–104. <https://doi.org/10.1016/J.JLAMP.2018.11.003>
- van Engelenburg, S., Janssen, M., Klievink, B., & Tan, Y.-H. (2017). Comparing a Shipping Information Pipeline with a Thick Flow and a Thin Flow. *Electronic Government, 16th IFIP WG 8.5 International Conference, EGOV2017, St. Petersburg, Russia, September 4-7, 2017, Proceedings, 10428*, 228–239. <https://doi.org/10.1007/978-3-319-64677-0>
- van Engelenburg, S., Janssen, M., Klievink, B., Tan, Y.-H., & Rukanova, B. (2018). Comparing the Openness of Archetypical Business-to-Government Information Sharing Architectures. In A. Zuiderwijk & C. C. Hinnant (Eds.), *Proceedings of 19th Annual International Conference on Digital Government Research (dg.o'18)*. <https://doi.org/10.1145/3209281.3209350>
- van Sinderen, M. J., van Halteren, A. T., Wegdam, M., Meeuwissen, H. B., & Eertink, E. H. (2006). Supporting Context-Aware Mobile Applications: An Infrastructure Approach. *Communications Magazine, IEEE*, (September), 96–104.
- van Stijn, E., Hesketh, D., Tan, Y.-H., Klievink, B., Overbeek, S., Heijmann, F., ... Butterly, T. (2011). Annex 3: The Data Pipeline. *Connecting International Trade : Single Windows and Supply Chains in the Next Decade*, 158–183. United Nations Economic Commission for Europe.
- Veenstra, A. W., Hulstijn, J., Christiaanse, R., & Tan, Y.-H. (2013). Information Exchange in Global Logistics Chains: an application for Model-based Auditing. *BNAlC 2013: Proceedings of the 25th Benelux Conference on Artificial Intelligence*. Delft, Netherlands.
- Venable, J. R. (2006). The Role of Theory and Theorising in Design Science Research. *DESIRIST 2006*, 1–18.
- Venable, J. R., Pries-Heje, J., Bunker, D., & Russo, N. L. (2010). Creation, Transfer, and Diffusion of Innovation in Organizations and Society: Information Systems Design Science Research for Human Benefit. *IFIP Working Conference on Human Benefit through the Diffusion of Information Systems Design Science Research*, 1–10.
- Verhagen, S. (2017). *Port community systems interoperability with the data pipeline, a case study in the port of Rotterdam*. Erasmus University Rotterdam.
- Verschuren, P., & Hartog, R. (2005). Evaluation in Design-Oriented Research. *Quality & Quantity*, 39, 733–762. <https://doi.org/10.1007/s11135-005-3150-6>
- Vieira, V., Tedesco, P., & Salgado, A. C. (2011). Designing context-sensitive systems: An integrated approach. *Expert Systems with Applications*, 38(2), 1119–1138. <https://doi.org/10.1016/j.eswa.2010.05.006>
- VISA. (2014). Benefits of accepting Visa. Retrieved November 21, 2018, from https://usa.visa.com/content_library/modal/benefits-accepting-visa.html
- Vukolić, M. (2016). The quest for scalable blockchain fabric: Proof-of-work vs. BFT replication. *International Workshop on Open Problems in Network Security*, 112–125. https://doi.org/10.1007/978-3-319-39028-4_9

- Wahyuni, D. (2012). The research design maze: understanding paradigms, cases, methods and methodologies. *Journal of Applied Management Accounting Research*, 10(1), 69–80.
- Waller, M. A., & Fawcett, S. E. (2013). Data science, predictive analytics, and big data: A revolution that will transform supply chain design and management. *Journal of Business Logistics*, 34(2), 77–84. <https://doi.org/10.1111/jbl.12010>
- Walls, J., Widmeyer, G., & El-Sawy, O. A. (1992). Building an Information System Design Theory for Vigilant EIS. *Information Systems Research*, Vol. 3, pp. 36–59. <https://doi.org/10.1287/isre.3.1.36>
- Walport, M. (2015). Distributed ledger technology: Beyond block chain. In *UK Government Office for Science*.
- Walsham, G. (1995a). Interpretive case studies in IS research: nature and method. *European Journal of Information Systems*, 4, 74–81.
- Walsham, G. (1995b). The Emergence of interpretivism in IS Research. *Information Systems Research*, 6(4), 376–394. <https://doi.org/10.1287/isre.6.4.376>
- Wang, Y.-K. (2004). Context Awareness and Adaptation in Mobile Learning. *2nd IEEE International Workshop on Wireless and Mobile Technologies in Education (WMTE'04)*, 154–158. <https://doi.org/10.1109/WMTE.2004.1281370>
- Ward, J. S., & Barker, A. (2013). Undefined By Data: A Survey of Big Data Definitions. *ArXiv Preprint ArXiv:1309.5821*, 2.
- Warren, D. H. D., Pereira, L. M., & Pereira, F. (1977). Prolog-the language and its implementation compared with Lisp. *ACM SIGART Bulletin*, 64(12), 109–115.
- Weber, I., Xu, X., Governatori, G., Ponomarev, A., & Mendling, J. (2016). Untrusted Business Process Monitoring and Execution Using Blockchain. *International Conference on Business Process Management*, 9850, 329–347. <https://doi.org/10.1007/978-3-319-45348-4>
- Wei, Q., Farkas, K., Prehofer, C., Mendes, P., & Plattner, B. (2006). Context-aware Handover Using Active Network Technology. *Journal of Computer Networks*, 50(15), 2855–2872.
- Whitmore, A., Agarwal, A., & Da Xu, L. (2015). The Internet of Things A survey of topics and trends. *Information Systems Frontiers*, 17(2), 261–274. <https://doi.org/10.1007/s10796-014-9489-2>
- Wieringa, R. (2014). Design Science Methodology for Information Systems and Software Engineering. In *Springer Berlin Heidelberg*. <https://doi.org/10.1145/1810295.1810446>
- Wimmer, M. A., & Tambouris, E. (2002). Online One-Stop Government. In R. Traunmüller (Ed.), *Information Systems: The e-Business Challenge* (pp. 117–130). https://doi.org/10.1007/978-0-387-35604-4_9
- Wimmer, M. A., Tambouris, E., Krimmer, R., Gil-Garcia, J. R., & Chatfield, A. T. (2017). Once Only Principle: Benefits, Barriers and Next Steps. *Proceedings of the 18th Annual International Conference on Digital Government Research - Dg.o '17*, 602–603. <https://doi.org/10.1145/3085228.3085296>
- Winograd, T. (2001). Architectures for Context. *Human-Computer Interaction*, 16(2), 401–419. https://doi.org/10.1207/S15327051HCI16234_18
- Winter, R. (2008). Design science research in Europe. *European Journal of Information Systems*, 17(5), 470–475. <https://doi.org/10.1057/ejis.2008.44>
- Wood, G. (2017). Ethereum: A Secure Decentralised Generalised Transaction Ledger. In *Ethereum project yellow paper*. Retrieved from <http://www.cryptopapers.net/papers/ethereum-yellowpaper.pdf>
- Yang, Y., & Padmanabhan, B. (2005). Evaluation of online personalization systems: A survey of evaluation schemes and a knowledge-based approach. *Journal of Electronic Commerce Research*, 6(2), 112–122.
- Yang, Z., Qilun, Z., & Fagui, L. (2008). An extended context model in a rfid-based context-aware service system. *Proceedings - 2nd 2008 International Symposium on Intelligent Information Technology Application Workshop, IITA 2008 Workshop*, 693–697. <https://doi.org/10.1109/IITA.Workshops.2008.255>
- Yildiz, M. (2007). E-government research: Reviewing the literature, limitations, and ways forward.

- Government Information Quarterly*, 24(3), 646–665.
- Yin, R. K. (1994). *Case study research: Design and methods* (2nd ed.; D. S. Foster, Ed.). Sage publications.
- Yu, Z., Yan, H., & Cheng, T. C. E. (2001). Benefits of Information Sharing with Supply Chain Partnerships. *Industrial Management & Data Systems*, 101(3), 114–121. <https://doi.org/10.1108/02635570110386625>
- Zachmann, J. A. A. (1987). A framework for information systems architecture. *IBM Systems Journal*, 26(3), 276–292.
- Zimmermann, A., Lorenz, A., Oppermann, R., & Augustin, S. (2007). An Operational Definition of Context. *CONTEXT'07 Proceedings of the 6th International and Interdisciplinary Conference on Modeling and Using Context*, 4635 LNAI, 558–571. https://doi.org/10.1007/978-3-540-74255-5_42
- Zins, C. (2007). Conceptual Approaches for Defining Data, Information, and Knowledge. *Journal of the American Society for Information Science and Technology*, 58(4), 479–493.
- Zomer, G. (2011). Smart Trade Logistics - Compliance as an Opportunity. *IT Innovations Enabling Seamless and Secure Supply Chains Witness 2011*, 769, 9–19.
- Zomer, G., de Putter, J., van Oosterhout, M., Dalen, J. van, Mueller, R., & Barz, A. (2014). *CASSANDRA - D5.5 – Integrated Evaluation Report – Integration of D5.2, D5.3 and D5.4*.
- Zomer, G., Tan, Y.-H., & Hofman, W. (2014). *D9.1 CASSANDRA Final Report*.
- Zwitter, A. (2014). Big Data ethics. *Big Data & Society*, 1–6. <https://doi.org/10.1177/2053951714559253>
- Zyskind, G., Nathan, O., & Pentland, A. S. (2015). Decentralizing privacy: Using blockchain to protect personal data. *IEEE Security and Privacy Workshops*, 180–184. <https://doi.org/10.1109/SPW.2015.27>

APPENDICES

Appendix A: Search query and overview of literature on context-aware systems

For the literature review on existing definitions of context (see section 5.1), we used the following query in Scopus:

TITLE-ABS-KEY ("definition of context" AND context-aware) OR
 TITLE-ABS-KEY ("defining context" AND context-aware) OR
 TITLE-ABS-KEY ("defining of context" AND context-aware) OR
 TITLE-ABS-KEY ("define context" AND context-aware) OR
 TITLE-ABS-KEY ("defined context" AND context-aware) OR
 TITLE-ABS-KEY ("context definition" AND context-aware) OR
 TITLE-ABS-KEY ("understanding of context" AND context-aware) OR
 TITLE-ABS-KEY ("understanding context" AND context-aware) OR
 TITLE-ABS-KEY ("understand context" AND context-aware) OR
 TITLE-ABS-KEY ("meaning of context" AND context-aware) OR
 TITLE-ABS-KEY ("context means" AND context-aware) OR
 TITLE-ABS-KEY ("conceptualization of context" AND context-aware) OR
 TITLE-ABS-KEY ("conceptualisation of context" AND context-aware) OR
 TITLE-ABS-KEY ("conceptualising context" AND context-aware) OR
 TITLE-ABS-KEY ("conceptualizing context" AND context-aware) OR
 TITLE-ABS-KEY ("concept of context" AND context-aware) OR
 TITLE-ABS-KEY ("conceptualize context" AND context-aware) OR
 TITLE-ABS-KEY ("conceptualise context" AND context-aware) OR
 TITLE-ABS-KEY ("context concept" AND context-aware) OR
 TITLE-ABS-KEY ("definition of context" AND context-awareness) OR
 TITLE-ABS-KEY ("defining context" AND context-awareness) OR
 TITLE-ABS-KEY ("defining of context" AND context-awareness) OR
 TITLE-ABS-KEY ("define context" AND context-awareness) OR
 TITLE-ABS-KEY ("defined context" AND context-awareness) OR
 TITLE-ABS-KEY ("context definition" AND context-awareness) OR
 TITLE-ABS-KEY ("understanding of context" AND context-awareness) OR
 TITLE-ABS-KEY ("understanding context" AND context-awareness) OR
 TITLE-ABS-KEY ("understand context" AND context-awareness) OR
 TITLE-ABS-KEY ("meaning of context" AND context-awareness) OR
 TITLE-ABS-KEY ("context means" AND context-awareness) OR
 TITLE-ABS-KEY ("conceptualization of context" AND context-awareness) OR
 TITLE-ABS-KEY ("conceptualisation of context" AND context-awareness) OR
 TITLE-ABS-KEY ("conceptualising context" AND context-awareness) OR
 TITLE-ABS-KEY ("conceptualizing context" AND context-awareness) OR
 TITLE-ABS-KEY ("concept of context" AND context-awareness) OR
 TITLE-ABS-KEY ("conceptualize context" AND context-awareness) OR
 TITLE-ABS-KEY ("conceptualise context" AND context-awareness) OR
 TITLE-ABS-KEY ("context concept" AND context-awareness) OR
 TITLE-ABS-KEY ("definition of context" AND context-sensitive) OR
 TITLE-ABS-KEY ("defining context" AND context-sensitive) OR
 TITLE-ABS-KEY ("defining of context" AND context-sensitive) OR
 TITLE-ABS-KEY ("define context" AND context-sensitive) OR
 TITLE-ABS-KEY ("defined context" AND context-sensitive) OR
 TITLE-ABS-KEY ("context definition" AND context-sensitive) OR
 TITLE-ABS-KEY ("understanding of context" AND context-sensitive) OR
 TITLE-ABS-KEY ("understanding context" AND context-sensitive) OR

TITLE-ABS-KEY ("understand context" AND context-sensitive) OR
 TITLE-ABS-KEY ("meaning of context" AND context-sensitive) OR
 TITLE-ABS-KEY ("context means" AND context-sensitive) OR
 TITLE-ABS-KEY ("conceptualization of context" AND context-sensitive) OR
 TITLE-ABS-KEY ("conceptualisation of context" AND context-sensitive) OR
 TITLE-ABS-KEY ("conceptualising context" AND context-sensitive) OR
 TITLE-ABS-KEY ("conceptualizing context" AND context-sensitive) OR
 TITLE-ABS-KEY ("concept of context" AND context-sensitive) OR
 TITLE-ABS-KEY ("conceptualize context" AND context-sensitive) OR
 TITLE-ABS-KEY ("conceptualise context" AND context-sensitive) OR
 TITLE-ABS-KEY ("context concept" AND context-sensitive) OR
 TITLE-ABS-KEY ("definition of context" AND context-sensitivity) OR
 TITLE-ABS-KEY ("defining context" AND context-sensitivity) OR
 TITLE-ABS-KEY ("defining of context" AND context-sensitivity) OR
 TITLE-ABS-KEY ("define context" AND context-sensitivity) OR
 TITLE-ABS-KEY ("defined context" AND context-sensitivity) OR
 TITLE-ABS-KEY ("context definition" AND context-sensitivity) OR
 TITLE-ABS-KEY ("understanding of context" AND context-sensitivity) OR
 TITLE-ABS-KEY ("understanding context" AND context-sensitivity) OR
 TITLE-ABS-KEY ("understand context" AND context-sensitivity) OR
 TITLE-ABS-KEY ("meaning of context" AND context-sensitivity) OR
 TITLE-ABS-KEY ("context means" AND context-sensitivity) OR
 TITLE-ABS-KEY ("conceptualization of context" AND context-sensitivity) OR
 TITLE-ABS-KEY ("conceptualisation of context" AND context-sensitivity) OR
 TITLE-ABS-KEY ("conceptualising context" AND context-sensitivity) OR
 TITLE-ABS-KEY ("conceptualizing context" AND context-sensitivity) OR
 TITLE-ABS-KEY ("concept of context" AND context-sensitivity) OR
 TITLE-ABS-KEY ("conceptualize context" AND context-sensitivity) OR
 TITLE-ABS-KEY ("conceptualise context" AND context-sensitivity) OR
 TITLE-ABS-KEY ("context concept" AND context-sensitivity) OR
 TITLE-ABS-KEY ("definition of context" AND context-oriented) OR
 TITLE-ABS-KEY ("defining context" AND context-oriented) OR
 TITLE-ABS-KEY ("defining of context" AND context-oriented) OR
 TITLE-ABS-KEY ("define context" AND context-oriented) OR
 TITLE-ABS-KEY ("defined context" AND context-oriented) OR
 TITLE-ABS-KEY ("context definition" AND context-oriented) OR
 TITLE-ABS-KEY ("understanding of context" AND context-oriented) OR
 TITLE-ABS-KEY ("understanding context" AND context-oriented) OR
 TITLE-ABS-KEY ("understand context" AND context-oriented) OR
 TITLE-ABS-KEY ("meaning of context" AND context-oriented) OR
 TITLE-ABS-KEY ("context means" AND context-oriented) OR
 TITLE-ABS-KEY ("conceptualization of context" AND context-oriented) OR
 TITLE-ABS-KEY ("conceptualisation of context" AND context-oriented) OR
 TITLE-ABS-KEY ("conceptualising context" AND context-oriented) OR
 TITLE-ABS-KEY ("conceptualizing context" AND context-oriented) OR
 TITLE-ABS-KEY ("concept of context" AND context-oriented) OR
 TITLE-ABS-KEY ("conceptualize context" AND context-oriented) OR
 TITLE-ABS-KEY ("conceptualise context" AND context-oriented) OR
 TITLE-ABS-KEY ("context concept" AND context-oriented) OR
 TITLE-ABS-KEY ("definition of context" AND context-based) OR
 TITLE-ABS-KEY ("defining context" AND context-based) OR
 TITLE-ABS-KEY ("defining of context" AND context-based) OR
 TITLE-ABS-KEY ("define context" AND context-based) OR

TITLE-ABS-KEY ("defined context" AND context-based) OR
 TITLE-ABS-KEY ("context definition" AND context-based) OR
 TITLE-ABS-KEY ("understanding of context" AND context-based) OR
 TITLE-ABS-KEY ("understanding context" AND context-based) OR
 TITLE-ABS-KEY ("understand context" AND context-based) OR
 TITLE-ABS-KEY ("meaning of context" AND context-based) OR
 TITLE-ABS-KEY ("context means" AND context-based) OR
 TITLE-ABS-KEY ("conceptualization of context" AND context-based) OR
 TITLE-ABS-KEY ("conceptualisation of context" AND context-based) OR
 TITLE-ABS-KEY ("conceptualising context" AND context-based) OR
 TITLE-ABS-KEY ("conceptualizing context" AND context-based) OR
 TITLE-ABS-KEY ("concept of context" AND context-based) OR
 TITLE-ABS-KEY ("conceptualize context" AND context-based) OR
 TITLE-ABS-KEY ("conceptualise context" AND context-based) OR
 TITLE-ABS-KEY ("context concept" AND context-based)

The result of this search, after filtering the irrelevant papers based on their abstract (see section 5.1.1), consists of the papers below.

- Lamsfus, C., Wang, D., Alzua-Sorzabal, A., Xiang, Z. Going Mobile: Defining Context for On-the-Go Travelers (2015) *Journal of Travel Research*, 54 (6), pp. 691-701.
- Zheng, Y. A revisit to the identification of contexts in recommender systems (2015) *International Conference on Intelligent User Interfaces*, Proceedings IUI, 29-March-2015, pp. 133-136.
- Champiri, Z.D., Shahamiri, S.R., Salim, S.S.B. A systematic review of scholar context-aware recommender systems (2015) *Expert Systems with Applications*, 42 (3), pp. 1743-1758.
- Seaver, N. The nice thing about context is that everyone has it (2015) *Media, Culture and Society*, 37 (7), pp. 1101-1109.
- Jaouadi, I., Djemaa, R.B., Abdallah, H.B. A Generic Metamodel for Context-Aware Applications (2015) *Advances in Intelligent Systems and Computing*, 1089, pp. 587-594.
- Nemoto, Y., Uei, K., Sato, K., Shimomura, Y. A context-based requirements analysis method for PSS design (2015) *Procedia CIRP*, 30, pp. 42-47.
- Bauer, J.S., Newman, M.W., Kientz, J.A. What designers talk about when they talk about context (2014) *Human-Computer Interaction*, 29 (5-6), pp. 420-450.
- Hussein, T., Linder, T., Gaulke, W., Ziegler, J. Hybreed: A software framework for developing context-aware hybrid recommender systems (2014) *User Modeling and User-Adapted Interaction*, 24 (1-2), pp. 121-174.
- Pascalau, E., Nalepa, G.J., Kluza, K. Towards a better understanding of context-aware applications (2013) *2013 Federated Conference on Computer Science and Information Systems*, FedCSIS 2013, art. no. 6644129, pp. 959-962.
- Decouchant, D., Mendoza, S., Sánchez, G., Rodríguez, J. Adapting groupware systems to changes in the collaborator's context of use (2013) *Expert Systems with Applications*, 40 (11), pp. 4446-4462.
- Seo, S.-S., Kang, J.-M., Han, Y., Hong, J.W.-K. Context management for user-centric context-aware services over pervasive networks (2012) *14th Asia-Pacific Network Operations and Management Symposium: "Management in the Big Data and IoT Era"*, APNOMS 2012 - Final Program, art. no. 6356050.
- Duggal, A., Misra, M., Srinivasaraghavan, R. Categorising context and using short term contextual information to obtain long term context (2012) *Proc. of the 11th IEEE Int. Conference on Trust, Security and Privacy in Computing and Communications*, TrustCom-2012 - 11th IEEE Int. Conference on
- Dąbrowski, M., Gromada, J., Moustafa, H. Context-awareness for IPTV services personalization (2012) *Proceedings - 6th International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing*, IMIS 2012, art. no. 6296828, pp. 37-44.
- Žontar, R., Heričko, M., Rozman, I. Taxonomy of context-aware systems (2012) *Elektrotehniski*

- Vestnik/Electrotechnical Review, 79 (1-2), pp. 41-46.
- Gómez, N., Cubo, J., Fuentes, L., Pimentel, E. Configuring a context-aware middleware for wireless sensor networks (2012) *Sensors* (Switzerland), 12 (7), pp. 8544-8570.
- Xu, J.Y., Sun, Y., Wang, Z., Kaiser, W.J., Pottie, G.J. Context guided and personalized activity classification system (2011) *Proceedings - Wireless Health 2011, WH'11*, art. no. 12.
- Orsi, G., Tanca, L. Context modelling and context-aware querying: (Can datalog be of help?) (2011) *Lecture Notes in Computer Science* (including subseries *Lecture Notes in Artificial Intelligence* and *Lecture Notes in Bioinformatics*), 6702 LNCS, pp. 225-244.
- Liu, W., Li, X., Huang, D. A survey on context awareness (2011) *2011 International Conference on Computer Science and Service System, CSSS 2011 - Proceedings*, art. no. 5972040, pp. 144-147.
- Etzion, O., Magid, Y., Rabinovich, E., Skarbovsky, I., Zolotorevsky, N. Context-based event processing systems (2011) *Studies in Computational Intelligence*, 347, pp. 257-278.
- Vieira, V., Tedesco, P., Salgado, A.C. Designing context-sensitive systems: An integrated approach (2011) *Expert Systems with Applications*, 38 (2), pp. 1119-1138.
- Lee, S., Chang, J., Lee, S.-G. Survey and trend analysis of context-aware systems (2011) *Information* (Japan), 14 (2), pp. 527-547.
- Arase, Y., Ren, F., Xie, X. User activity understanding from mobile phone sensors (2010) *UbiComp'10 - Proceedings of the 2010 ACM Conference on Ubiquitous Computing*, art. no. 1864452, pp. 391-392.
- Lin, F., Butters, J., Sandkuhl, K., Ciravegna, F. Context-based ontology matching: Concept and application cases (2010) *Proceedings - 10th IEEE International Conference on Computer and Information Technology, CIT-2010, 7th IEEE International Conference on Embedded Software and Systems, ICESS-2010, ScalCom-2010*, art. no. 5577869, pp. 1292-1298.
- Benou, P., Vassilakis, C. The conceptual model of context for mobile commerce applications (2010) *Electronic Commerce Research*, 10 (2), pp. 139-165.
- Taconet, C., Kazi-Aoul, Z. Building context-awareness models for mobile applications (2010) *Journal of Digital Information Management*, 8 (2), pp. 78-87.
- Pal, R. Context-sensitive probabilistic boolean networks: Steady-state properties, reduction, and steady-state approximation (2010) *IEEE Transactions on Signal Processing*, 58 (2), art. no. 5210193, pp. 879-890.
- Tan, E.M.-Y., Foo, S., Goh, D.H.-L., Theng, Y.-L. TILES: Classifying contextual information for mobile tourism applications (2009) *Aslib Proceedings: New Information Perspectives*, 61 (6), pp. 565-586.
- Vale, S., Hammoudi, S. ODM-based architecture for the development of mobile context-aware applications (2009) *ACM International Conference Proceeding Series*, art. no. 9.
- Bolchini, C., Curino, C.A., Orsi, G., Quintarelli, E., Rossato, R., Schreiber, F.A., Tanca, L. And what can context do for data? (2009) *Communications of the ACM*, 52 (11), pp. 136-140.
- Vale, S., Hammoudi, S. COMODE: A framework for the development of context-aware applications in the context of MDE (2009) *Proceedings of the 2009 4th International Conference on Internet and Web Applications and Services, ICIW 2009*, art. no. 5072529, pp. 261-266.
- Vieira, V., Tedesco, P., Salgado, A.C. A process for the design of context-sensitive systems (2009) *Proceedings of the 2009 13th International Conference on Computer Supported Cooperative Work in Design, CSCWD 2009*, art. no. 4968049, pp. 143-148.
- Du, W., Wang, L. Context-aware application programming for mobile devices (2008) *ACM International Conference Proceeding Series*, 273, art. no. 1370292, pp. 215-227.
- Kim, E., Choi, J. A context management system for supporting context-aware applications (2008) *Proceedings of The 5th International Conference on Embedded and Ubiquitous Computing, EUC 2008*, 2, art. no. 4755288, pp. 577-582.
- Vieira, V., Brézillon, P., Salgado, A.C., Tedesco, P. A context-oriented model for domain-independent context management (2008) *Revue d'Intelligence Artificielle*, 22 (5), pp. 609-627.

- Yang, Z., Qilun, Z., Fagui, L. An extended context model in a rfid-based context-aware service system (2008) Proceedings - 2nd 2008 International Symposium on Intelligent Information Technology Application Workshop, IITA 2008 Workshop, art. no. 4732032, pp. 693-697.
- Fernandes, P., Werner, C., Murta, L. Feature modeling for context-aware software product lines (2008) 20th International Conference on Software Engineering and Knowledge Engineering, SEKE 2008, pp. 758-763.
- Li, H., Jyri, S., Jian, M., Kuifei, Y. Research on context-aware mobile computing (2008) Proceedings - International Conference on Advanced Information Networking and Applications, AINA, art. no. 4482885, pp. 24-30.
- Pallapa, G., Das, S.K. Challenges of designing privacy enhanced context-aware middleware for assisted healthcare (2008) Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 5120 LNCS, pp. 200-207.
- Hur, H., Park, H., Kim, J., Chang, H., Lee, M., Cho, S., Choi, E. Context model of user aspect in context-awareness (2008) International Conference on Advanced Communication Technology, ICACT, 3, art. no. 4494098, pp. 1647-1651.
- Mena, T.B., Saoud, N.B.-B., Ahmed, M.B., Pavard, B. Towards a methodology for context sensitive systems development (2007) Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 4635 LNAI, pp. 56-68.
- Zimmermann, A., Lorenz, A., Oppermann, R. An operational definition of context (2007) Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 4635 LNAI, pp. 558-571.
- Wei, E.J.Y., Chan, A.T.S. Towards context-awareness in ubiquitous computing (2007) Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 4808 LNCS, pp. 706-717.
- Lau, S.B.-Y., Lee, C.-S. Context aware reference model: Architecture and implications for adaptation of learning activities (2007) Mobility Conference 2007 - The 4th Int. Conf. Mobile Technology, Applications and Systems, Mobility 2007, Incorporating the 1st Int. Symp. Computer Human Interaction in Mobile Technology, IS-CHI 2007, pp. 623-627.
- Bricon-Souf, N., Newman, C.R. Context awareness in health care: A review (2007) International Journal of Medical Informatics, 76 (1), pp. 2-12.
- Liaqut, H., Iftikhar, N., Qadir, M.A. Context aware information retrieval using role ontology and query schemas (2006) 10th IEEE International Multitopic Conference 2006, INMIC, art. no. 4196414, pp. 244-249.
- O'Brien, P., Abidi, S.S.R. Contextual knowledge sharing in a P2P network (2006) IEEE International Conference on Engineering of Intelligent Systems, ICEIS 2006, art. no. 1703169.
- Reponen, E., Mihalic, K. Model of primary and secondary context (2006) Context in advanced Interfaces International Workshop in Conjunction with AVI 2006, CAI '06, 2006, pp. 37-38.
- Anderson, K.M., Hansen, F.A., Bouvin, N.O. Templates and queries in contextual hypermedia (2006) Proceedings of the Seventeenth ACM Conference on Hypertext and Hypermedia, HT'06, 2006, pp. 99-109.
- Khedo, K.K. Context-aware systems for mobile and ubiquitous networks (2006) Proceedings of the International Conference on Networking, International Conference on Systems and International Conference on Mobile Communications and Learning Technologies, ICN/ICONS/MCL'06, 2006, art. no. 1628369.
- Tanter, É., Gybels, K., Denker, M., Bergel, A. Context-aware aspects (2006) Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 4089 LNCS, pp. 227-242.
- Bucur, O., Beaune, P., Boissier, O. Steps towards making contextualized decisions: How to do what you can, with what you have, where you are (2006) Lecture Notes in Computer

- Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 3946 LNAI, pp. 62-85.
- Christopoulou, E., Goumopoulos, C., Kameas, A. An ontology-based context management and reasoning process for UbiComp applications (2005) ACM International Conference Proceeding Series, 121, pp. 265-270.
- Gong, L. Contextual modeling and applications (2005) Conference Proceedings - IEEE International Conference on Systems, Man and Cybernetics, 1, pp. 381-386.
- Bucur, O., Beaune, P., Boissier, O. Définition et représentation du contexte pour des agents sensibles au context (2005) ACM International Conference Proceeding Series, 120, pp. 13-16.
- Brézillon, P. Task-realization models in contextual graphs (2005) Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 3554 LNAI, pp. 55-68.
- Przybilski, M., Nurmi, P., Floréen, P. A framework for context reasoning systems (2005) Proceedings of the IASTED International Conference on Software Engineering: part of the 23rd IASTED International Multi-Conference on Applied Informatics, SE 2005, pp. 448-452.
- Rukzio, E., Siorpaes, S., Falke, O., Hussmann, H. Policy based adaptive services for mobile commerce (2005) Proceedings - 2005 Second IEEE International Workshop on Mobile Commerce and Services, WMCS'05, 2005, art. no. 1581591, pp. 183-192.
- Go, Y.C., Sohn, J.-C. Context modeling for intelligent robot services using rule and ontology (2005) The 7th International Conference on Advanced Communication Technology, ICACT 2005, 2, art. no. 1462900, pp. 813-816.
- Bazire, M., Brézillon, P. Understanding context before using it (2005) Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 3554 LNAI, pp. 29-40.
- Van Bunningen, A.H., Feng, L., Apers, F.M.G. Context for ubiquitous data management (2005) Proceedings of the International Workshop on Ubiquitous Data Management, UDM 2005, pp. 17-24.
- Huang, W., Tao, T. Adding context-awareness to knowledge management in modern enterprises (2004) 2004 2nd International IEEE Conference 'Intelligent Systems' - Proceedings, 2, pp. 393-398.
- Sato, K. Context-sensitive approach for interactive systems design: Modular scenario-based methods for context representation (2004) Journal of Physiological Anthropology and Applied Human Science, 23 (6), pp. 277-281.
- Wang, Y.-K. Context awareness and adaptation in mobile learning (2004) Proceedings - 2nd IEEE International Workshop on Wireless and Mobile Technologies in Education, pp. 154-158.
- Zimmer, T. Towards a better understanding of context attributes (2004) Proceedings - Second IEEE Annual Conference on Pervasive Computing and Communications, Workshops, PerCom, pp. 23-27.
- Bristow, H.W., Baber, C., Cross, J., Knight, J.F., Woolley, S.I. Defining and evaluating context for wearable computing (2004) International Journal of Human Computer Studies, 60 (5-6), pp. 798-819.
- Calvary, G., Coutaz, J., Thevenin, D., Limbourg, Q., Bouillon, L., Vanderdonckt, J. A Unifying Reference Framework for multi-target user interfaces (2003) Interacting with Computers, 15 (3 SPEC.), pp. 289-308.
- Oulasvirta, A., Kurvinen, E., Kankainen, T. Understanding contexts by being there: Case studies in bodystorming (2003) Personal and Ubiquitous Computing, 7 (2), pp. 125-134.
- Lee, C., Helal, S. Context attributes: An approach to enable context-awareness for service discovery (2003) Proceedings - 2003 Symposium on Applications and the Internet, SAINT 2003, art. no. 1183029, pp. 22-30.
- Lueg, C. Operationalizing context in context-aware artifacts: Benefits and pitfalls (2002)

- Informing Science, 5 (2), pp. 43-47.
- Roman, G.-C., Julien, C., Huang, Q. Network abstractions for context-aware mobile computing (2002) Proceedings - International Conference on Software Engineering, pp. 363-373.
- Dey, A.K., Abowd, G.D., Salber, D. A conceptual framework and a toolkit for supporting the rapid prototyping of context-aware applications (2001) Human-Computer Interaction, 16 (2-4), pp. 97-166.
- Dey, A.K. Understanding and using context (2001) Personal and Ubiquitous Computing, 5 (1), pp. 4-7.
- Salber, D., Dey, A.K., Abowd, G.D. The context toolkit: Aiding the development of context-enabled applications (1999) Conference on Human Factors in Computing Systems - Proceedings, pp. 434-441.
- Abowd, G.D., Dey, A.K., Brown, P.J., Davies, N., Smith, M., Steggles, P. Towards a better understanding of context and context-awareness (1999) Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 1707, pp. 304-307.
- Schilit, Bill, Adams, Norman, Want, Roy Context-aware computing applications (1995) Mobile Computing Systems and Applications - Workshop Proceedings, pp. 85-90.

Appendix B: Overview of context elements and objects

In this appendix, we present an overview of adaptors and sensors, the context elements that they sense or manipulate and the context relationships that connect the context elements to a focus.

Sensor / Adaptor (section)	Context element sensed / manipulated	Context relationship (section)
Information routers (10.1.1)	$hasAccess\left(\begin{matrix} SetofDataElements, \\ Party \end{matrix}\right)$	Not sharing sensitive data (9.1.2.1)
		Encrypt sensitive data (9.1.2.2)
		Connection to shipment (9.1.2.11)
		Filter data (9.1.2.12)
		Shield data from competitors (9.2.3.6)
	$hasAccessSystem\left(\begin{matrix} SetofDataElements, \\ System \end{matrix}\right)$	Not broadcasting sensitive data (9.1.2.6)
		No thick, global pipeline (9.1.2.7)
		Thick international pipeline (9.1.2.8)
		Thin global pipeline (9.1.2.9)
		Check system security (9.1.2.10)
	$canAggregate\left(\begin{matrix} Party, \\ SetofDataElements \end{matrix}\right)$	No aggregating sensitive data (9.1.2.3)
		No aggregating for customs (9.1.2.4)
		Only viewing sensitive data for customs (9.1.2.5)

Table 37: Overview of adaptors and sensors, the context elements that they sense or manipulate and the context relationships that connect the context elements to a focus

Sensor / Adaptor (section)	Context element sensed / manipulated	Context relationship (section)
Information routers (10.1.1)	$routeinFlow\left(\begin{array}{c} ListofSystems, \\ SetofDataElements \end{array}\right)$	Submit documents (9.2.3.1)
		Authorised submitting documents (9.2.3.2)
		No Sharing against agreement (9.2.3.3)
		Protect trade secrets (9.2.3.4)
		Not share competitively sensitive data with competitor (9.2.3.5)
		Not make public competitively sensitive data (9.2.3.7)
		Sharing public data (9.2.3.8)
	$filter\left(\begin{array}{c} SetofDataElements, \\ FilteredSetofDataElements, \\ Business \end{array}\right)$	Filter data (9.1.2.12)
Encryption component (10.1.2.1)	$encrypted(SetofDataElements, Key)$	Encrypt sensitive data (9.1.2.2)
		Shield data from competitors (9.2.3.6)
Thin maker (10.1.2.2)	$thin(SetofDataElements)$	Thin global pipeline (9.1.2.9)
Data viewer (10.1.2.3)	$view(Party, SetofDataElements)$	Only viewing sensitive data for customs (9.1.2.5)
Businesses involved in information sharing (10.2.1)	$connection(Party, Shipment)$	Connection to shipment (9.1.2.11)

Table 37 (continued): Overview of adaptors and sensors, the context elements that they sense or manipulate and the context relationships that connect the context elements to a focus

Sensor / Adaptor (section)	Context element sensed / manipulated	Context relationship (section)
Businesses involved in information sharing (10.2.1)	$hasRole(Party, Shipment, Role)$	Submit documents (9.2.3.1)
		Authorised submitting documents (9.2.3.2)
	$agreement \left(\begin{matrix} Party_1, Party_2, \\ Action, \\ SetofDataElements \end{matrix} \right)$	No Sharing against agreement (9.2.3.3)
	$agreements \left(\begin{matrix} Business, SetofParties, \\ Action, \\ SetofDataElements \end{matrix} \right)$	Protect trade secrets (9.2.3.4)
	$authorized \left(\begin{matrix} Party_1, Party_2, Action, \\ SetofDataElements, \\ Shipment, \\ GovernmentAgency \end{matrix} \right)$	Authorised submitting documents (9.2.3.2)
	$encrypted(SetofDataElements)$	Submit documents (9.2.3.1)
		Authorised submitting documents (9.2.3.2)
	$thick(SetofDataElements)$	No thick, global pipeline (9.1.2.7)
		Thick international pipeline (9.1.2.8)
	$subject \left(\begin{matrix} SetofDataElements, \\ Shipment \end{matrix} \right)$	Connection to shipment (9.1.2.11)

Table 37 (continued): Overview of adaptors and sensors, the context elements that they sense or manipulate and the context relationships that connect the context elements to a focus

Sensor / Adaptor (section)	Context element sensed / manipulated	Context relationship (section)
Businesses involved in information sharing (10.2.1)	$sensitiveTo \left(\begin{matrix} SetofDataElements, \\ Business, Party \end{matrix} \right)$	Not sharing sensitive data (9.1.2.1)
		Encrypt sensitive data (9.1.2.2)
		Not broadcasting sensitive data (9.1.2.6)
		Check system security (9.1.2.10)
		Filter data (9.1.2.12)
	$sensitiveToAggregate \left(\begin{matrix} SetofDataElements, \\ Business, Party \end{matrix} \right)$	No aggregating sensitive data (9.1.2.3)
		No aggregating for customs (9.1.2.4)
		Only viewing sensitive data for customs (9.1.2.5)
	$entitled \left(\begin{matrix} Action, \\ SetofDataElements, \\ Business, Party \end{matrix} \right)$	Encrypt sensitive data (9.1.2.2)
		No aggregating for customs (9.1.2.4)
	$securityRequirement \left(\begin{matrix} Business, \\ SetofDataElements, \\ Level \end{matrix} \right)$	Check system security (9.1.2.10)
	$tradesecrets(SetofDataElements, Business)$	Protect trade secrets (9.2.3.4)

Table 37 (continued): Overview of adaptors and sensors, the context elements that they sense or manipulate and the context relationships that connect the context elements to a focus

Sensor / Adaptor (section)	Context element sensed / manipulated	Context relationship (section)
Independent third parties (10.2.2)	<i>partyType(PartyType, Party)</i>	No aggregating for customs (9.1.2.4)
		Only viewing sensitive data for customs (9.1.2.5)
	<i>broadcasts(System)</i>	Not broadcasting sensitive data (9.1.2.6)
	<i>systemType(System, SystemType)</i>	No thick, global pipeline (9.1.2.7)
		Thick international pipeline (9.1.2.8)
		Thin global pipeline (9.1.2.9)
	<i>geographicalScope(System, Scope)</i>	No thick, global pipeline (9.1.2.7)
		Thick international pipeline (9.1.2.8)
		Thin global pipeline (9.1.2.9)
	<i>controlsDataAccess(System)</i>	Thick international pipeline (9.1.2.8)
	<i>securityProvided(System, Level)</i>	Check system security (9.1.2.10)
	<i>systemOf(System, Party)</i>	Submit documents (9.2.3.1)
		Authorised submitting documents (9.2.3.2)
		No Sharing against agreement (9.2.3.3)
		Not share competitively sensitive data with competitor (9.2.3.5)
	<i>makesPublic($\begin{matrix} System, \\ SetofDataElements \end{matrix}$)</i>	Not make public competitively sensitive data (9.2.3.7)

Table 37 (continued): Overview of adaptors and sensors, the context elements that they sense or manipulate and the context relationships that connect the context elements to a focus

Sensor / Adaptor (section)	Context element sensed / manipulated	Context relationship (section)
Independent third parties (10.2.2)	$commerciallySensitive \left(\begin{matrix} SetofDataElements, \\ Business, Competitor \end{matrix} \right)$	Not share competitively sensitive data with competitor (9.2.3.5)
		Shield data from competitors (9.2.3.6)
	$disturbMarket(SetofDataElements)$	Not make public competitively sensitive data (9.2.3.7)
	$public(SetofDataElements)$	No Sharing against agreement (9.2.3.3)
		Not share competitively sensitive data with competitor (9.2.3.5)
		Sharing public data (9.2.3.8)
	$possibleAccess \left(\begin{matrix} System, \\ SetofDataElements, \\ Competitor \end{matrix} \right)$	Not share competitively sensitive data with competitor (9.2.3.5)
Customs (10.2.3)	$obligation \left(\begin{matrix} Action, \\ SetofDataElements, \\ Shipment, \\ GovernmentAgency, \\ Role \end{matrix} \right)$	Submit documents (9.2.3.1)
		Authorised submitting documents (9.2.3.2)

Table 37 (continued): Overview of adaptors and sensors, the context elements that they sense or manipulate and the context relationships that connect the context elements to a focus

ACKNOWLEDGEMENTS

Writing my dissertation seems like the end of a journey that was often fun and sometimes a bit rocky. Writing a dissertation has been my point on the horizon for a very long time and it almost seems unreal that I actually did it. I feel very lucky with the opportunities that I was provided with that made it possible for me to go on this journey and to pursue my goals, as I do realise that so many others never had the same opportunities.

I had a great time in the last four years. Not only because I really like doing research, but also because of the people I got to work with and learn from. Moreover, having good teachers can be very valuable and I believe that during my PhD research, I had some very good ones.

I want to thank Marijn Janssen for teaching me a great deal on how to be a researcher. I could always count on you and your advice. I am also very grateful for all the opportunities you provided me with that helped me to further develop my skills. I would like to thank Bram Klievink as well for his valuable advice and feedback over the years. Especially your advice on research methods and how to apply them have been very useful to me and to my research. In addition, I appreciate that you both were always very positive and confident that I would do well and that you gave me the freedom to make my own choices and to find my own way when I needed it.

There were many others that contributed to my research as well. Frank Smeele, Frank Stevens, Wouter Verheyen and Aspasia Karampetsou helped me to get the juridical insights that I required for my research. Furthermore, various people at Maersk Line provided very useful feedback on the architecture from a more practical point of view. I would also like to thank the people from the Cassandra and CORE project, as their research and their data have been very useful for my research. The contribution of several anonymous interviewees that volunteered their time to help me obtain the data I needed has been very valuable as well. Furthermore, I would like to thank Raphaël Hannaert and Prince Singh for their help with evaluating the method.

In addition, I am grateful to Yao-Hua Tan and Boriana Rukanova for their feedback and the insights that they provided into the international container-shipping domain. I really liked writing papers together and I hope we can continue to cooperate in the future. I want to thank Jolien Ubacht for her great advice on supervising students and the insightful discussions we had about blockchain technology. In addition to the members of my committee that I already mentioned, I would like to thank Boris Shishkov, Maria Wimmer and Martijn Warnier for being part of my doctoral committee and for their feedback.

The ICT section where I worked the last four years really felt like a home to me where I could be myself. I cannot mention everyone individually here, but I would like to thank all of you for the support, the inspiration and all the kind words and encouragement I received from you. I would like to especially thank Anneke Zuiderwijk, Klara Pigmans, Ali Latifi, Rijk Mercuur, Naci Karkin, Jordi Bieger, Alexia Athanasopoulou, Majid Mohammadi and Agung Wahyudi. Thank you for the interesting conversations during our walks and coffee breaks, and for all the fun we had. Furthermore, I much appreciated the help and support of Laura Bruns and Jo-Ann Karna.

I owe a lot of gratitude to my friends Annouska, Jorien, Nicole and Elisabeth. It is very valuable to have friends like you. Thank you for always being there whenever I needed you and for listening to me. Equally important, thank you for showing me that there is a world outside of work and for all the fun and laughs. In addition, I would like to thank my parents and sisters very much for their genuine interest in my work and their support. I very much appreciate your understanding for my ambitions, even though it sometimes meant that I had less time for you.

Last, but definitely not least, I am very grateful to Bart. You have been closest to me during the whole process and you were there for all the ups and downs. You helped to get my mind off of work and motivated me to persevere when I needed it. You gave me the space I needed to do this, without ever making me feel like I fell short towards you. I am very happy and lucky to have found a partner that understands me as you do and that just accepts me for who I am.

CURRICULUM VITAE

Personal Information

Name: Selinde Helena van Engelenburg
Date of birth: 17 September 1984
Birthplace: Purmerend, the Netherlands

Education

Graduate school (Delft University of Technology, 2015-2018)

Master Artificial Intelligence (Utrecht University, 2009-2012)

Track: Logic and Intelligent Systems

Master's thesis: A temporal argumentation logic for medical diagnosis

GPA: 4

Bachelor Cognitive Artificial Intelligence (Utrecht University, 2006-2009)

Bachelor's thesis: Gödel's incompleteness theorems

Bachelor Psychology (Utrecht University, 2003-2005)

Track: Clinical psychology

Not completed, but obtained all courses for major.

Gymnasium (Anna van Rijn College in Nieuwegein, 1996-2002)

Experience

PhD Candidate (Delft University of Technology, February 2015 – February 2019)

Title of dissertation: Designing context-aware architectures for business-to-government information sharing

Technical coordinator and project leader (IntraQuest, June 2012 – August 2014)

Internship (Alan Turing Institute Almere, April 2011 – December 2011)

Developing a temporal argumentation logic for medical diagnosis as part of master's thesis research.

Internship (TNO, June 2010 – February 2011)

Designing a prototype of an assistant for astronauts on long-duration space missions at The Netherlands Organisation for Applied Scientific Research (TNO) in collaboration with the European Space Agency.

Teaching experience

Daily supervisor (Frank Leeuwenburg, 2017)

Master's thesis: Data Quality in Inter-organizational Product Information Sharing

Internal supervisor (Floor Seuren, 2018)

Master's thesis: Introducing Blockchain to Commercial Real Estate

Advisor (Raphael Hannaert, 2018)

Master's thesis: Designing a Context-aware Decentralized Marketplace for Sensor Data

Advisor (Tanya Tapaneya-Olarn, 2018)

Master's thesis: The impact of a context-aware architecture for B2G information sharing on data quality in the container-shipping domain

Course: Fundamentals of Data Analytics (Delft University of Technology, 2018)

Preparing and providing lab sessions

Course: Integrated Design of ICT Architectures (Delft University of Technology, 2018)

Preparing and providing a lecture

Publications

van Engelenburg, S., Janssen, M., & Klievink, B. (2015). Design of a Business-to-Government Information Sharing Architecture Using Business Rules. In *Software Engineering and Formal Methods* (Vol. 9509, pp. 124–138). Springer. <http://doi.org/10.1007/978-3-319-15201-1>

van Engelenburg, S., Janssen, M., & Klievink, B. (2017a). Design of a software architecture supporting business-to-government information sharing to improve public safety and security. *Journal of Intelligent Information Systems*. <http://doi.org/10.1007/s10844-017-0478-z>

van Engelenburg, S., Janssen, M., & Klievink, B. (2017b). What belongs to context? A definition, a criterion and a method for deciding on what context-aware systems should sense and adapt to. In A. Cerone & M. Roveri (Eds.), *Software Engineering and Formal Methods 2017* (Vol. 10729 LNCS, pp. 101–116). Springer International Publishing AG. http://doi.org/10.1007/978-3-319-74781-1_8

van Engelenburg, S., Janssen, M., Klievink, B., & Tan, Y. (2017). Comparing a Shipping Information Pipeline with a Thick Flow and a Thin Flow. In *Electronic Government, 16th IFIP WG 8.5 International Conference, EGOV2017, St. Petersburg, Russia, September 4-7, 2017, Proceedings* (Vol. 10428, pp. 228–239). Springer International Publishing AG. <http://doi.org/10.1007/978-3-319-64677-0>

van Engelenburg, S., Janssen, M., & Klievink, B. (2018). A Blockchain Architecture for Reducing the Bullwhip Effect. In B. Shishkov (Ed.), *BMSD 2018, LNBIP 319* (pp. 69–82). Springer International Publishing AG. <http://doi.org/10.1007/978-3-319-94214-8>

van Engelenburg, S., Janssen, M., Klievink, B., Tan, Y., & Rukanova, B. (2018). Comparing the Openness of Archetypical Business-to-Government Information Sharing Architectures. In A. Zuiderwijk & C. C. Hinnant (Eds.), *Proceedings of 19th Annual International Conference on Digital Government Research (dg.o'18)*. ACM, New York, NY, USA. <http://doi.org/10.1145/3209281.3209350>

Klievink, B., Janssen, M., Voort, H. Van Der, & van Engelenburg, S. (2018). Regulatory Compliance and Over-Compliant Information Sharing – Changes in the B2G Landscape. In EGOV 2018, LNCS 11020 (pp. 249–260). <http://doi.org/10.1007/978-3-319-98690-6>

van Engelenburg, S., Janssen, M., Klievink, B., Engelenburg, S. Van, Janssen, M., & Klievink, B. (2019). Designing context-aware systems: a method for understanding and analysing

context in practice. *Journal of Logical and Algebraic Methods in Programming*, 103, 79–104. <http://doi.org/10.1016/J.JLAMP.2018.11.003>

Academic conferences

Attending and presenting

13th International Conference on Software Engineering and Formal Methods 2015 (SEFM2015)

15th International Conference on Software Engineering and Formal Methods 2017 (SEFM2017)

19th Annual International Conference on Digital Government Research 2018 (dg.o'18)

8th International Symposium on Business Modeling and Software Design 2018 (BMSD2018)

Reviewer

The 19th Annual International Conference on Digital Government Research 2018 (dg.o'18)

IFIP Electronic Government 2018 (EGOV 2018)

Session chair

Session on Data Analytics at the 14th IFIP Conference on e-Business, e-Services and e-Society (I3E 2015)

Session on Internet of Things and Method Engineering at the Eight International Symposium on Business Modeling and Software Design (BMSD2018)

Other activities

Presenting and participating as an invited expert on blockchain technology

8th meeting of the European Commission Customs 2020 Project Group to study a possible framework to develop the EU Single Window environment for customs (EU-SW) including the legal context (12-14 March 2018, Rotterdam/the Hague, the Netherlands).

The European Commission workshop on “Blockchain and Distributed Ledger Technology for Taxation and Customs IT Systems (29-30 May 2018, Valetta, Malta).

Other workshops and presentations

Participation in the Lorentz Workshop on “Normware: Modeling Norms and Automated Norm Application” (14-18 November 2016, Leiden University)

Invited presenter at Regioraad Zuid-West of EvoFenedex on blockchain technology and the context-aware architecture
(19 September 2018, Naaldwijk, the Netherlands)

