Universität Hamburg
DER FORSCHUNG | DER LEHRE | DER BILDUNG

University of Hamburg
Faculty for Mathematics, Informatics und Natural Sciences
Department of Computer Science

Master's Thesis

IoT Extensions of the Enterprise Architecture Exemplified by the smartPORT Hamburg

In Cooperation with

Hamburg Port Authority

Jöran Tesse

3tesse@informatik.uni-hamburg.de, joeran@riuna.de

Studiengang:	IT-Management & -Consulting
Matrikelnr.:	6591462
Fachsemester:	06
Erstgutachter:	Prof. Dr. Ingrid Schirmer
Zweitgutachter:	Prof. Dr. Tilo Böhmann
Betreuer:	Ulrich Baldauf (Hamburg Port Authority, AöR)

Es ist, wie es ist.

Ein jegliches hat seine Zeit.

Abstract

The influence of the *Internet of Things* (IoT) on IT-related and connected layers of *enterprise architecture management* (EAM) is investigated from both a theoretical and a practical viewpoint. The close cooperation with the *Hamburg port authority* (HPA) allowed for this approach and led to several IoT-specific extensions to EAM. These extensions are captured in a meta model and applied to collected architectural data of the practitioner gathered through interviews. This iterative process was guided by a growing list of architectural concerns refining both the meta model as well as an altered view on the investigated layers, which aims to strategically describe IoT-specific aspects of IT- or infrastructure-related architectural artifacts as well as making suggestions to form a meaningful level of granularity for them.

Contents

1 Introduction and Motivation **1**
1.1 Internet of Things 1
1.2 Enterprise Architecture 3
1.3 Concerns, Views and Meta Models 4
1.4 Architecture Integration 5
1.5 Research Questions 6
1.6 Structure of this Thesis 6

2 Related Literature **9**
2.1 Enterprise Architecture Layers 10
2.2 Challenges in Enterprise Architecture Management 11
2.3 IoT-related Architecture Aspects 12
2.4 Fog Computing 14
2.5 CMDB and Sensor Locality View 15
2.6 Cloud Computing 16
2.7 Information Streams 16

3 Organizational Context **19**
3.1 smartPORT Initiative 19
3.2 smartPORT Projects 19
3.2.1 Traffic Simulation System 20
3.2.2 Port Monitor 21
3.2.3 Smart Parking 21
3.2.4 PrePORT Parking 22
3.2.5 Smart Delineator 22
3.2.6 Smart Switch 23
3.2.7 Smart Road 24
3.3 Organizational Context 24
3.4 Consolidation Phase 25

4 Action Design Research **27**
4.1 Artifact: Concerns and Vision 30
4.2 Artifact: Layers 30
4.3 Artifact: Meta Model 30
4.4 Artifact: Instantiated Architecture 31
4.5 Interviews 31

5 Capturing IoT-Aspects in EA Meta Model Extensions **33**
5.1 Concerns 33
5.2 Physical View 36
5.3 Class and Instance Logic 37
5.4 Layers: Smart Brick, Fog, Cloud 38
5.4.1 Cloud 40
5.4.2 Fog 42
5.4.3 smart brick 44
5.5 Flow of Information 47
5.5.1 Communication Attributes 48
5.5.2 Content Attributes 49
5.5.3 Relationships between Information Flows 49
5.6 Metamodell 51
5.7 Mapping to EAM Tool (iteraplan) 52
5.7.1 Layers: Cloud, Fog and smart brick 52
5.7.2 Information Flows 53
5.7.3 smart bricks 53

6 Derived Models and Evaluation **55**

7 Discussion, Conclusion and Outlook **57**
7.1 Service Application Layer 58
7.2 Fog Computing 59
7.3 Stage of IoT and Future Architectures 60
7.4 Further Influencing Factors 63
7.5 Relation to CMDB 64

7.6 Emerging Views . . . 65
7.6.1 Storage View . . . 65
7.6.2 Maintenance View . . . 66
7.6.3 Security View . . . 66

A Appendix **69**
A.1 Interview Questionnaire . . . 69
A.2 Conducted Interviews . . . 77
A.3 Interview Results . . . 79
A.4 Chapter 6: Applying EA-extensions to smartPORT projects and Evaluation . . . 80

References **81**

Bibliothekseinstellung **89**

Eidesstattliche Erklärung **91**

List of Figures

1.1 Global Growth of Smart Mobile Devices and Connections, as prognosticated by Cisco, Inc. 2
1.2 IoT application domains and related applications 3
1.3 Enterprise architecture as cross-layer view of aggregate artifacts . . 5

2.1 A framework for literature reviewing 9
2.2 Taxonomy of literature reviews (following [Cooper, 1988]) 10
2.3 Importance of enterprise architecture management goals according to a questionnaire conducted by [Winter et al., 2010]. 12
2.4 Results of non-representative German questionnaire about what artifact types are considered / implemented in the respondent's EA. 13
2.5 Illustration of different cloud system service models commonly used in IT. 16

4.1 Stages and Principles of Action Design Research 27
4.2 IT-dominant second stage of action design research 28

5.1 Relation of the proposed smart brick, fog and cloud layer to the essential layers of enterprise architecture management [Winter and Fischer, 2006]. 39
5.2 smart bricks are composed of a hierarchy of bricks (physical objects) and attached sensors. 44
5.3 An IoT-enhanced (fatigue-sensitive) bridge modeled according to the smart brick logic. 44
5.4 Flow of Information: Example for a raw data stream flowing from an induction loop to a section controller. 47
5.5 The meta model created for enterprise architecture management. . 51
5.6 The enterprise architecture model used in iteraplan. 52

7.1 Project prePORT parking: Enrollment is captured both as a service application and as an actuator in form of a smart brick. 59

7.2 A possible future enterprise architecture featuring a dedicated information hub. 61
7.3 A to-be enterprise architecture with unifying data acquisition and information provisioning. 62
7.4 An privacy-extension applicable to all types of flowing information. 67
7.5 A data privacy-concerned visualization of the HPA's derived enterprise architecture model. 67
7.6 A data confidentiality- and integrity-concerned visualization of the HPA's derived enterprise architecture model. 68

List of Tables

5.1 Evaluation of what concerns are answered throughout this thesis. . 36

Glossary

AIM .... Application Integration Management

BIE .... Building, Intervention and Evaluation; Stage 2 in Action Design Research as proposed by [Sein et al., 2011]

CMDB .... Configuration Management DataBase

EA .... Enterprise Architecture

EAM .... Enterprise Architecture Management

HPA .... Hamburg Port Authority

IoT .... Internet of Things

IS .... Information System

IT .... Information Technology

QOS .... Quality of Service

TAS .... Traffic Analysis System

TSS .... Traffic Simulation System

1 Introduction and Motivation

The dawning of the so called fourth industrial revolution [bitkom, 2016] poses significant challenges to a variety of industries. One of these challenges is the adoption of *internet of things* (IoT) technologies, which others also expect to be an influential driver of innovation [Gartner, 2015]. Today, some industries are already facing changing markets due to the internet of things, such as the automotive industry, where new, IoT-enhanced business models appeared in form of car sharing [Steininger et al., 1996]. The logistics industry is currently at a stage, where new technologies are evaluated and new business models are emerging on the basis of automatic sensor evaluation and increasingly interconnected devices, as the cooperation with the *Hamburg port authority* (HPA) during the course of this thesis suggests. While many projects were already launched at the HPA prior to this thesis, the influence of increasing variety in the types of employed systems is not yet sufficiently documented at a strategic level. Being an important tool for strategic IT-management, the *enterprise architecture* (EA) was chosen to be the main subject of research for this thesis. Since enterprise architecture management is strongly influenced by practitioner's needs and commonly built around specific concerns [Winter et al., 2010], the implications posed to an enterprise architecture by an IoT-enriched IT-landscape are evaluated in close cooperation with a practitioner from the logistics industry, the Hamburg port authority.

1.1 Internet of Things

The *Internet of Things* (IoT) is not a well-defined term, but rather an "umbrella term" [Bassi et al., 2013, page 13] merging different aspects of technological systems where physical objects are connected by means of IT. Some define the IoT paradigm to necessarily incorporate the Internet [Bassi et al., 2013, page 14] and make a clear distinction between "Internet of Things" and "Intranet of Things". Another perspective is to acknowledge different visions of the subject, while defining the IoT paradigm as the intersection between an "Internet-oriented vision", a "Things-oriented vision" and a "semantics-oriented vision" [Atzori et al., 2010]. Following Atzori et al. to call these perspectives as well as the paradigm itself "visions", the term will hereafter be used to refer to IT landscapes where "a high number of different objects - univocally addressable - constitute an underlying IoT fabric" [Borgia, 2014, page 3].

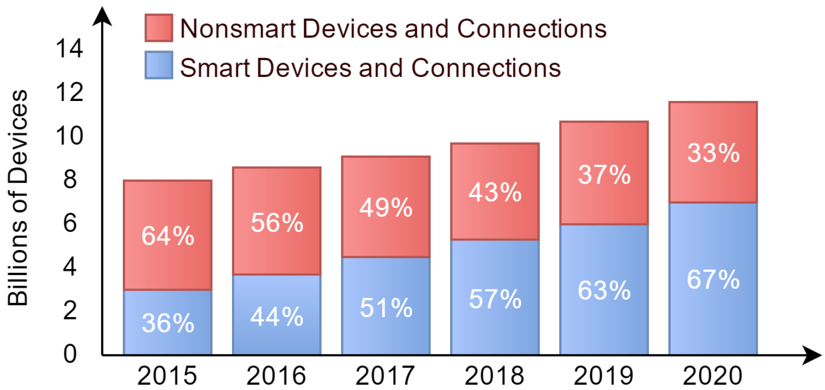

Figure 1.1: Global Growth of Smart Mobile Devices and Connections, as prognosticated by Cisco, Inc.

***Source**: [Cisco, 2016b]*

Accordingly, there are no specific criteria for IoT systems such as wireless communication, sensors or a connection to the Internet. Instead, IoT systems follow the idea to incorporate addressable physical objects. This implies that observed attributes of an object have to be perceptible and need to be uniquely identified and perceived at the desired level of detail, commonly by means of sensors. The number of devices grows differently in different areas, but devices "smartness" is expected to increase significantly [Cisco, 2016b], as seen in figure 1.1.

As the IoT paradigm is not purely a technological paradigm, it is rather referring to situations where heterogeneous IT-landscapes, automation and increasing numbers of sensor types are involved. This is exemplified in [Borgia, 2014] by showing different applications of the IoT paradigm and linking them to IoT-specific domains, as seen in figure 1.2. Still, the paradigm induces certain developments in IT landscapes, as it commonly involves the utilization of various sensors to either "sense" physical objects or to allow physical objects to "sense" their environment. This results in an increasing complexity for an organizations enterprise architecture.

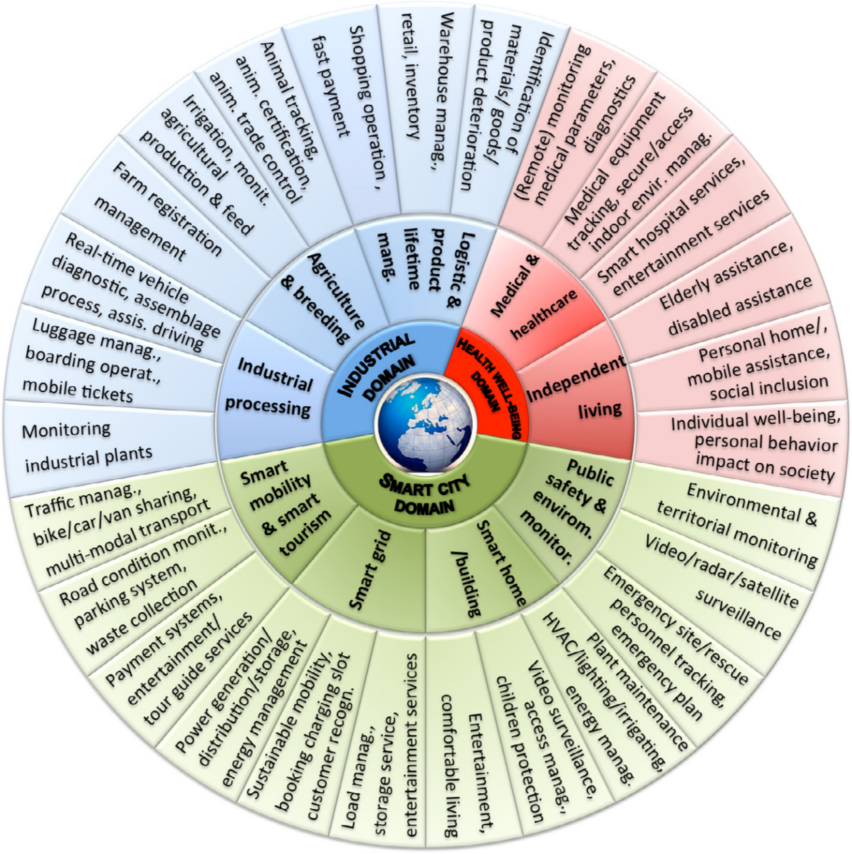

Figure 1.2: IoT application domains and related applications
***Source**: [Borgia, 2014, page 9]*

1.2 Enterprise Architecture

An enterprise architecture (EA) is a big-picture view of an organization's various business units, projects and other components of organizational importance, as well as details about how information is networked and what dependencies exist [Hanschke, 2009]. Accordingly, the enterprise architecture can be seen as a depiction of the actual organization, attempting to document the current, planned and target state. In a more formal way, the enterprise architecture can be defined as the "fundamental concepts or properties of a system in its environment embodied in its elements, relationships, and in the principles of its design and evolution" [ISO, 2011]. In detail, an enterprise architecture describes both the "the structures in business and IT and the links that exist between them" [Hanschke, 2009], meaning that an enterprise architecture focuses on the interface between business and IT, describing the business structures relevant for the IT and the IT structures relevant to the business as well as the links in-between.

Although a complete depiction of the enterprise architecture seems desirable, the effort in documenting every detail about an organization contradicts this in practice. Therefore, an enterprise architecture is always a trade-off between effort and detail. Accordingly, one premise of enterprise architecture management is to find a subset of the IT-landscape that provides sufficient information at reasonable expenditure of human labor. The definition of "sufficient information" is challenging, as EAM is utilized to solve manifold goals. Generally, it is most commonly part of strategic IT management [Hanschke, 2009] and is a tool to align business and IT as well as providing transparency [Winter et al., 2010].

Conclusively, the goal of enterprise architecture is to help understand patterns of systems and to reveal links between IT-related and business-related artifacts, e.g. explaining the link between a certain sensor and the service of displaying traffic-related advices on video boards. Therefore, less details modeled can provide more information, as data without strategic concern may be relocated to dedicated architectures.

1.3 Concerns, Views and Meta Models

One of the first publications about enterprise architectures already states that "what you think architecture is depends on what you are doing" [Zachman, 1987], which probably still holds true today. For different purposes, multiple viewpoints of the same data are necessary to transport the required information in a graspable way. In order to understand a specific model, it is not only important to display relevant data, but also to reduce the amount of irrelevant data as well. Enterprise architectures are growing increasingly complex, and the complexity can only be tackled by providing *architectural viewpoints* (views), that promote certain information or address specific concerns. The provided information is targeted to answer particular concerns, which play an important role in determining any particular view. Similar to other methods related to requirements

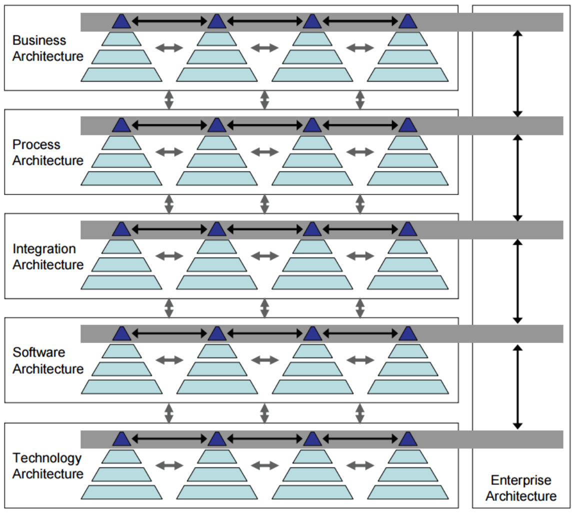

Figure 1.3: Enterprise architecture as cross-layer view of aggregate artifacts
Source: *[Winter and Fischer, 2006, page 3]*

engineering, many enterprise architecture frameworks start out by decomposing complex architectures into different "pieces" ("perspectives" in [Zachman, 2016], extended by "layers" as will be explained in section 2.1), which are commonly related to specific concerns. As depicted in figure 1.3, an important role of enterprise architecture management is to provide views integrating architectural artifacts of different layers, thus combining distinct disciplines of architectural management.

1.4 Architecture Integration

Prior to this thesis, another master's thesis was conducted in direct predecession to form a theoretical basis. Therefore, this thesis will be related to [Windelen, 2015] as both the preceding thesis and most of the referenced literature were reviewed before any other research was conducted. While sharing a similar theoretic basis, the specific focus of this thesis is quite different, as it builds on top of Windelen's findings. From the practitioner's perspective, the two main results of [Windelen, 2015] were the creation of a *locality viewpoint*, which was introduced to link IT components to physical locations at instance level, and a first attempt to model the emerged architecture in an architecture modeling tool.

The questions arising from this work were if and how the proposed sensor locality view could be integrated into enterprise architecture management, and how the emerged architecture could be sorted, aggregated and/or grouped to better reflect the specific roles of different information systems, rather then positioning and grouping them by project affinity. Additionally, it is investigated what implications the internet of things poses to an enterprise architecture, and what level of detail is required for different artifacts in enterprise architecture management to support its goals, most importantly business-IT-alignment and transparency.

1.5 Research Questions

The main research questions are what IT- and infrastructure-related aspects of enterprise architecture need to be refined when IoT-technologies are adopted, what aspects of the IoT paradigm are relevant to strategic goals of enterprise architecture management, what new concerns emerge around IoT-related projects and their resulting EA models, and how this can be incorporated in both a meta model and resulting enterprise architecture models created following the derived meta model.

1.6 Structure of this Thesis

Following this introduction, the thesis is structured into six additional parts. Firstly, chapter 2 provides a literature-based overview of the concerned topics and outlines progress made in these areas. Then, a detailed description of the organizational context is given in chapter 3 to explain the purpose and goal of the investigated projects. Following this, a description of the applied methodology is given in chapter 4. The theoretical results of this thesis are captured in chapter 5, while the application of these results is discussed int chapter A.4. Finally, the last chapter provides discussions and conclusions as well as an outlook for possible future developments in the area of subject.

2 Related Literature

This chapter will summarize prior research by means of a literature research process, which was guided by [Brettle and Gambling, 2003], [Brocke et al., 2009] and [Blaxter, 2010]. To circumvent researching subjects already practiced or dealt

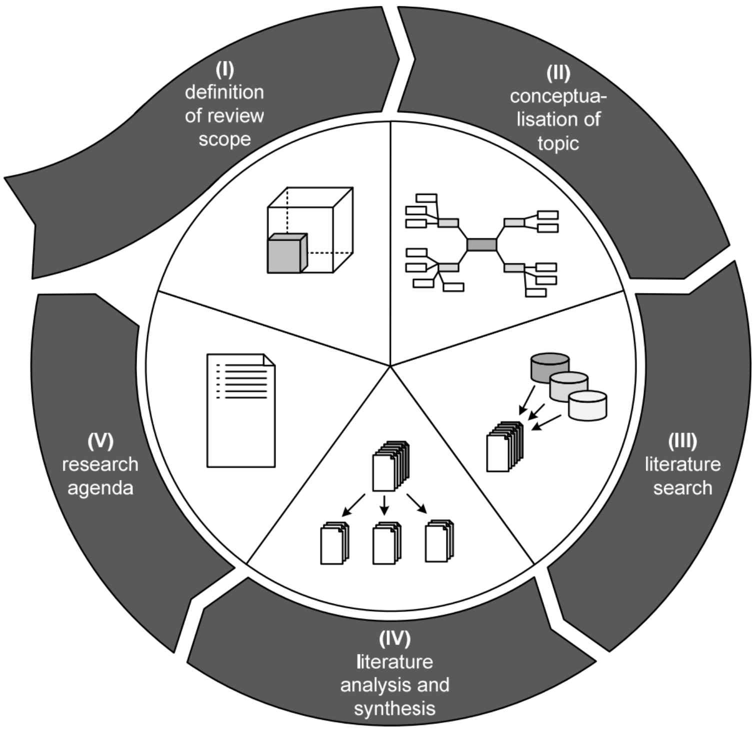

Figure 2.1: A framework for literature reviewing
***Source**: [Brocke et al., 2009]*

with in science, and to circumvent inappropriate or ineffective decisions along the path, an extensive literature review was conducted [Brettle and Gambling, 2003]. To organize the research process, the structure depicted in figure 2.1 was applied. A methodological focus was put on the depicted phase (I) as suggested by [Brocke et al., 2009], following their advice to create a taxonomy as depicted in figure 2.2. A broad variety of literature was reviewed, while an essential first step was reviewing meta-studies summarizing other literature (e.g. [Winter et al., 2010]). Throughout the literature review process, several gaps between posed architectural concerns (see section 4.1) and capabilities of current and proposed architectural principles were discovered.

<table>
<tr><th colspan="2">Characteristic</th><th colspan="12">Categories</th></tr>
<tr><td>(1)</td><td>focus</td><td colspan="3">research outcomes</td><td colspan="3">research methods</td><td colspan="3">theories</td><td colspan="3">applications</td></tr>
<tr><td>(2)</td><td>goal</td><td colspan="4">integration</td><td colspan="4">criticism</td><td colspan="4">central issues</td></tr>
<tr><td>(3)</td><td>organisation</td><td colspan="4">historical</td><td colspan="4">conceptual</td><td colspan="4">methodological</td></tr>
<tr><td>(4)</td><td>perspective</td><td colspan="6">neutral representation</td><td colspan="6">espousal of position</td></tr>
<tr><td>(5)</td><td>audience</td><td colspan="3">specialised scholars</td><td colspan="3">general scholars</td><td colspan="3">practitioners/politicians</td><td colspan="3">general public</td></tr>
<tr><td>(6)</td><td>coverage</td><td colspan="3">exhaustive</td><td colspan="3">exhaustive and selective</td><td colspan="3">representative</td><td colspan="3">central/pivotal</td></tr>
</table>

Figure 2.2: Taxonomy of literature reviews (following [Cooper, 1988])
Source*: [Brocke et al., 2009, Cooper, 1988]*

Another important aspect to reviewing literature is to define the leveraged sources of literature. In the case of this thesis, a non-public source of information was available in forms of internal documentation repositories at the Hamburg port authority. While these documents were mostly confidential, interesting aspects outlining the viability of the designed meta model were found, and dependencies to other architectural domains, such as software architectures and use cases, were reviewed.

2.1 Enterprise Architecture Layers

The enterprise architecture management is a discipline separable into distinct layers. These span across artifacts of different domains and can be commonly assigned to the following five essential layers, as described in [Aier et al., 2008] and [Winter and Fischer, 2006]:

Business Architecture: This layer is dedicated to business strategy and contains information about *what* is needed, wanted, used or important to pursue business, namely organizational goals and success factors, services, market segments, strategic goals and interaction within the business ecosystem [Aier et al., 2008] [Winter and Fischer, 2006].

Process Architecture: The process layer is responsible for capturing organizational information related to *how* business is pursued, e.g. distribution channels, business processes, performance indicators, organizational units, roles, responsibilities, flow of information and sites (locations) [Aier et al., 2008] [Winter and Fischer, 2006].

Integration Architecture: This layer "presents the fundamental organization of information system components in the relevant enterprise context" [Winter and Fischer, 2006]. In other words, the integration architecture aims to emphasize the collaboration and dependencies of information system components.

Software Architecture: This layer contains template information about applications, declaring their inner components at coarse- and fine-grained levels as well as the logical data structure of any interfaces within the software or interfaces to other applications. Similar to the integration layer, links between application components and applications can be depicted, but the focus of this layer is not to depict relationships between applications, but rather to depict an application's inner structure in detail, e.g. for labor planning during implementation.

Technology Architecture: The technology layer "represents the fundamental organization of computing / telecommunications hardware and networks" [Winter and Fischer, 2006] and is therefore strongly related to a *configuration management database* (CMDB) as presented in ITIL® (e.g. [Van Haren, 2011]). In contrast to ITIL's® CMDB, the technology architecture is intended to support IT strategy, rather than IT operation. Accordingly, the technology architecture is not composed of information about deployed hardware instances as a CMDB would be, but holds information about classes of hardware, or "technology patterns" [Correia et al., 2009].

2.2 Challenges in Enterprise Architecture Management

According to [Winter et al., 2010], there are several reasons to implement *enterprise architecture management* (EAM), most importantly to align business and IT, as seen in figure 2.3. Most of the ascertained goals are not directly related to one

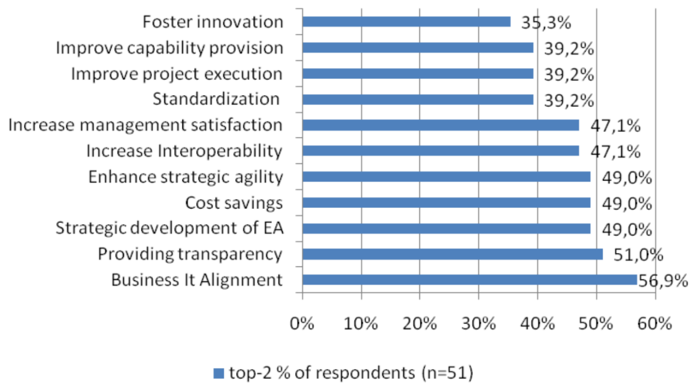

Figure 2.3: Importance of enterprise architecture management goals according to a questionnaire conducted by [Winter et al., 2010].
***Source**: [Winter et al., 2010]*

specific architectural layer (see section 2.1), but require linking artifacts of at least two layers. Therefore, a prevalent challenge of enterprise architecture management is to meaningfully depict relevant links between artifacts of different layers. This not only implies the need to meaningfully depict logically complex details of such links, but is further exacerbated by the tremendous number of possible links. It is essential to only model links between carefully selected components with each other, as generic linking would impose an amount of effort for understanding such models that the aforementioned EA management goals could hardly be met.

Regarding artifacts within the proposed architectural layers, a non-representative questionnaire summarized subjective estimates shown in figure 2.4. According to the depicted results, actual technological objects such as hardware and network components are understood as good as they need to be. However section 1.1 reasons that the internet of things paradigm might pose new challenges to this area of enterprise architecture management. It is not certain whether the current models suffice for architectural concerns posed by IoT-related IT-landscapes.

2.3 IoT-related Architecture Aspects

Essential to the internet of things is the "things" aspect. These "things" are most commonly but not necessarily physical objects. While the concept is generally

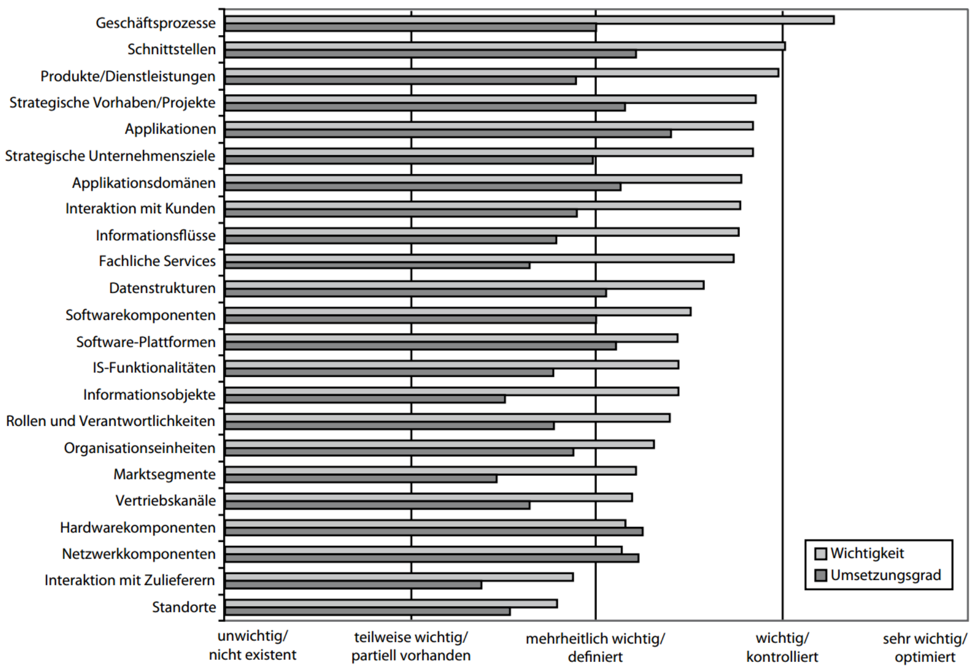

Figure 2.4: Results of non-representative German questionnaire about what artifact types are considered / implemented in the respondent's EA.
Source: *[Aier et al., 2008]*

to encompass non-technological entities linked to services or processes [Meyer et al., 2015], the naming is inconsistent in literature, e.g. "physical entity", "entity of interest" or "object" [Haller, 2010]. "Things" are directly relevant to supply chain management and unsurprisingly found in dedicated tools such as SAP. As a specialized "thing"-management already exists in a different context, it is still interesting to investigate the relevance of "thing"-aspects to enterprise architecture management.

For clarifying the relationship between "things" and other aspects such as devices or resources, Haller states that devices are related to things in two ways. They are either attached to an entity of interest or form an "environmental sensor" monitoring the entity/entities of interest. Still, "things" are seen as actual instances of objects (tangible). This is in accordance to literature on software architecture where a relation between actual objects and technological devices is seen [Bassi et al., 2013]. However, these entities are not only relevant to software architecture, but also to other layers of an enterprise architecture, e.g. for the process layer [Meyer et al., 2015]. In contrast to software architecture or other technological viewpoints, an instance-based view is not feasible for all areas in enterprise architecture management. Processes, for example, are not always linkable to specific instances of objects, which is sometimes even acknowledged but circumvented in other literature, e.g. by introducing an attribute "isMultiInstance" (Boolean) in [Meyer et al., 2015].

As "things" are present in business architecture, introducing "things" to software architecture is strongly advised [Bassi et al., 2013] and including them to process modeling is deemed vital for a wider adoption of IoT [Haller and Magerkurth, 2011], the presence of "things" seems to appear in different forms on several layers of enterprise architecture management. Therefore, the practical relevance of "things" in layers where they do not exist yet, as well as in enterprise architecture management generally, should be investigated.

2.4 Fog Computing

The term is probably related or even adopted from the fog computing paradigm (original source unknown), which, in the IoT context, “extends Cloud computing and services to the edge of the network” [Stojmenovic and Wen, 2014, page 1]. This can be due to several reasons, but is mainly related to "managing latency and scalability through localization of computation" [Breivold et al., 2015]. If data is interpreted closer to a sensor, then only the results need to be transmitted over the network, thus reducing the overall network load in many cases. A temperature sensor, for example, might measure the temperature every second. If said sensor is installed outside, a fog system could aggregate the data and forward the temperature average every other minute, which would be sufficient for most applications while reducing the theoretical network load for this sensor by 99%.

Additionally, the predictability in latency improves as the computing system moves closer to the data source. An off-site sensor connected via the internet would have considerably more delay jitter than if a processing unit would be installed at the sensor's location. These are important metrics for *quality of service* (QOS) applications, especially for real-time applications.

Fog computing is also relevant for privacy and security issues, e.g. an anonymizing fog system might be critical for enabling services such as video-based road traffic measurements. However, deploying fog systems is not exclusively beneficial, as deployed fog systems themselves need to be secured both physically and virtually. A multi-purpose single-board computer with linux functioning as a fog system might be more easy to exploit than a specialized sensor relying on non-flashable firmware. Therefore, both hardware and software aspects can be relevant for certain concerns.

2.5 CMDB and Sensor Locality View

ITIL®propagates the usage of a configuration management database [Van Haren, 2011], which is distinct from enterprise architecture management in that it handles instances of systems, not technological patterns, as already mentioned in section 2.1. However, an enterprise architecture is concerned with connecting relevant parts of different aspects of an enterprise architecture. Although the CMDB is not seen to be part of EAM [Winter and Fischer, 2006, Correia et al., 2009], a merging context exists because the EA comprises technological patterns derived from information present in the CMDB [Correia et al., 2009]. Similarly, a *sensor locality view* conceptually links instances of systems to specific locations [Windelen, 2015]. Following the distinction between CMDB and EAM [Correia et al., 2009], the proposed sensor locality view does not yet sufficiently transport the concept of locality to enterprise architecture management. Furthermore, the link between instances of systems and specific locations does not imply that types of systems should be linked to types of locations.

2.6 Cloud Computing

The term cloud computing is generally defined on a technical level and is constituted of different characteristics. On a technological level, characteristics such variable, automated resource pooling [Armbrust et al., 2010], broad network accessibility and elastic scalability [Mell and Grance, 2011] are mentioned. Another characteristic is the automatic measuring of service execution and resulting system adaption [Mell and Grance, 2011]. From a strategic point-of-view, the cloud system's broad accessibility renders them location-independent to some extent, and their specific computational capabilities are of subordinate strategical relevance for service execution, as they can be elastically adapted to meet the requirements in service execution. Different service models for cloud systems exist

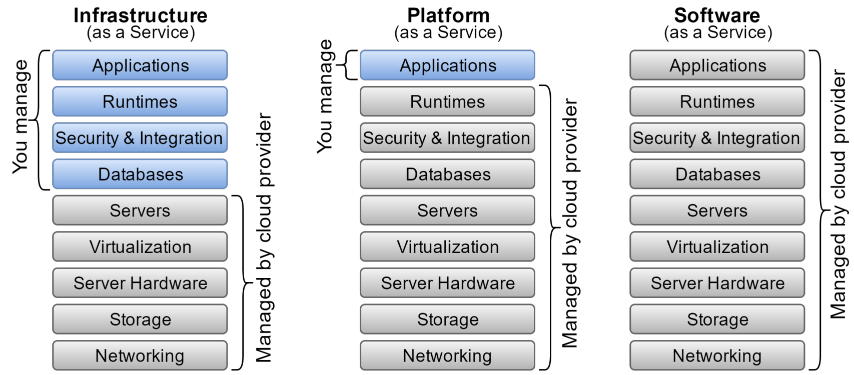

Figure 2.5: Illustration of different cloud system service models commonly used in IT. ***Source:*** *Slightly modified version of [Vilaplana et al., 2013, figure 1]*

[Mell and Grance, 2011, Vilaplana et al., 2013], as depicted in figure 2.5. These service models illustrate the separation of underlying hardware and service execution, although partial knowledge about the software is required even for the "Software as a Service"-scenario.

2.7 Information Streams

Flowing information flows between instances of systems, while being mostly homogeneous among equal types of systems. Therefore, flowing information has previously been attributed to EAM (e.g. [Matthes et al., 2016]). However, literature on specific attributes is scarce and publications attempting to standardize communication at a company-internal level suggest that a common description of flowing information is highly relevant [Kumar and Sowjanya, 2015]. Apparently, enterprise architecture management settled with depicting dependencies, while detailed descriptions of interfaces and dependencies were subject to *application integration management* (AIM). Literature on AIM is scarce whatsoever, and due to its similarity to software architecture management, where interfaces and transmitted data may also be described in detail, AIM is expected to be carried out at project level in a non-uniform way, i.e. with varying notions and levels of detail across different projects.

3 Organizational Context

The contents of this thesis and acquired insights are at least partially influenced by the specific context of the observed and contributing company, the port of Hamburg's local port authority (HPA). Therefore, this chapter will elaborate the setting in terms of both the tasks and responsibilities carried out by the Hamburg Port Authority, as well as describing characteristics of investigated projects.

3.1 smartPORT Initiative

The harbor of Hamburg is challenged by having to handle increasing freight volumes [HPA, 2016] within a confined space. Due to the surrounding city of Hamburg, the port's industrial area cannot be expanded to allow for additional infrastructure elements such as roads or railroad tracks. Therefore, the local port authority strives to improve the existing infrastucture's efficiency in order to increase the overall freight volume handled by the port of Hamburg. This can be achieved in multiple ways, for example by efficient traffic guidance, by reducing the number of vehicles within the harbor area and by reducing the downtime of infrastructure elements.

In order to explore capabilities of new technologies, the Hamburg Port Authority launched the *smartPORT initiative* [HPA, 2015b] aiming to identify, implement, test and evaluate IoT-related scenarios. Many projects within the smartPORT initiative introduced new sources of data by introducing either new types of sensors or by new means of interpreting the output of already employed sensor types.

3.2 smartPORT Projects

This section will briefly introduce those projects of the smartPORT initiative that were investigated in the context of this thesis. For choosing projects to investigate, there were no fixed criteria, but all of the picked projects are smartPORT projects and they are mostly related to road traffic and to applications of sensor types that were either not yet used routinely or common sensor types used in innovative ways.

3.2.1 Traffic Simulation System

Previously, the only system to analyze traffic flowing through the harbor area, was a commercially available system. It was bought as standard software and accordingly was externally developed but is internally operated. Based on best practice, the *Traffic Analysis System* (TAS) utilized only double inductive loops, as standardized under the *"Technische Lieferbedingungen für Streckenstationen"* (TLS) [BASt, 2012]. The elaborate TLS standard is the result of governmental regulation and harmonization, developed and refined by traffic engineers over a large timespan (decades). The TAS' level of standardization allowed for a considerable level of predictability regarding the accuracy of created traffic flow estimates, yet it proved to be quite inflexible in terms of incorporating new types of sensors or attempting to produce traffic forecasts. This existing system is still used and valuable to the HPA, as it forms the central gathering point for a hierarchical structure of sensor connectivity and data aggregation, and implements a database for short- and longterm data storage. Therefore, the existing system was only marginally changed to expose certain interfaces to an added traffic simulation system, e.g. for accessing induction loop data.

The new, enhanced system for traffic simulation (TSS) has the unique points of representing an integrated approach by incorporating data gathered by a diverse yet unfinalized variety of sensors, and simulating actual traffic flow based on the gathered data in order to generate traffic forecasts. Accordingly, the TSS extends the existing system both on a qualitative and on a functional level, as it leverages more data to create a more accurate estimation of the current traffic situation, and generates new outputs in form of traffic forecasts, which may be used for automated strategic decisions and planning, for example when computing traffic routes.

3.2.2 Port Monitor

Satisfying experience with a monitoring system for nautical routing and coordination led to this integrated monitoring system, an attempt to extend the existing system by incorporating the road and railway traffic situation as well as information about traffic-related events, e.g. from movable bridges, thus creating an integrated view of the overall traffic situation throughout all traffic routes within the port area. The goal was to accelerate decision making by increasing transparency and displaying the infrastructure status and events in a single, integrated system.

3.2.3 Smart Parking

A significant portion of the traffic within the port area is related to finding a parking lot, as the overall traffic in cities related to finding parking lots is estimated to be 30% [APCOA, 2013]. The Smart Parking project is an explorative project to evaluate technologies for accounted parking lots. Such parking lots communicate their degree of occupancy, which is not as trivial as it would seem. Given the common practice of trucks disconnecting their trailer, for example to deposit an empty cargo container, which is either picked up later or collected by a colleague driving a different truck, the task of counting the amount of unoccupied space on the parking lot is more complicated than counting how often a boom gate opens. The resulting operational system of the Smart Parking project uses several strategically placed induction loops to generate “vehicle footprints”. Since an inductive loop measurement roughly corresponds to the amount of metal above the sensor, a vehicle footprint may be interpreted as a vehicle profile from the front to the back of the vehicle. Thus, if a leaving vehicle’s “footprint” only matches the first part of an entering vehicles “footprint”, it can be assumed that the leaving vehicle abandoned its trailer on the parking lot.

3.2.4 PrePORT Parking

Since the space within the harbor limits is confined, it is beneficial to park as many trucks as possible outside the port area. This contradicts the truckers' general preference to park inside the port area, because they need to reach their destined terminal reliably at fixed dates. To circumvent delays due to congestion or closed bridges, truckers currently try to park as close to the terminal as possible.

Building upon the traffic simulation system, the prePORT parking project introduces a new scheduling scheme built around notifying the drivers about their ideal time of departure. To give further incentive to the truckers, prePORT parking allows them to place reservations, reducing their expended time on finding a suitable parking lot. Since placing a reservation would also help routing traffic more efficiently, the overall traffic load could be reduced, thus increasing the infrastructure's efficiency and directly fulfilling a strategic goal of the HPA. Also, prePORT parking is planned to be part of an integrated system for planning the logistical chain within the port of Hamburg.

Another aspect of the project is the unique way that the available space is used. As part of the project, trucks parking on parking lots enhanced with PrePORT Parking are mandated to specify their projected time of departure either manually or by disclosing their desired time of arrival, leveraging the TSS to estimate a reasonable time of departure. Then, trucks are assigned parking lanes, where all trucks park bumper-to-bumper as they are all scheduled to leave the parking lot approximately at the same time. This allows for an increased vehicle density of almost 100% (according to HPA employee Hermann Grünfeld).

3.2.5 Smart Delineator

Not addressing traffic management directly, the smart delineator project aims to reduce the expenditure on maintenance, both in terms of cost and personnel. Delineators are used to guide traffic in road construction sites and are required to have a shining light at their tops. As part of governmental regulation, any infrastructure operator is required to replace the batteries every 3 weeks to assure that every single light is glowing at all times, although this specific type of battery has shown to last several months, according to an HPA employee. Also, the delineators need to be checked regularly to ensure they were not hit by a car, turned, displaced or otherwise functionally constrained.

Since the attached battery was sufficiently powerful, a small circuit board was integrated into the delineator's head, transmitting the battery current and interpreted values of an accelerometer, indicating both the battery status as well as whether the delineator was tipped, hit, turned or otherwise shocked. Considering the cost of long-range data transmission, both in regard to hardware cost as well as energy cost from the battery, each smart delineator sends its data to a local gathering unit, which then forwards the aggregated data to the dedicated processing system.

3.2.6 Smart Switch

Railway switches are naturally subject to attrition, mainly adhesive wear. They need to be greased regularly, because a failing switch might not be bypassed on other rails, thus impacting many other railway sections and trains. To reduce the expenditure on excess greasing while still reducing the risk of switch failure, the Smart Switch project introduces a system to estimate the condition of a railway switch. Several strategically important switches were chosen to pilot the technology and allow operational evaluation of the employed technology. Generally, the condition is estimated by gauging the force needed to slide the switch into the opposite position, which is specifically achieved by both a dynamometer and an ampere meter at the electromotor. This allows maintenance staff to service switches on demand and to schedule switch maintenance to a convenient time of day, when the worn out switch is least trafficked.

3.2.7 Smart Road

The Smart Road project was introduced to implement and evaluate several innovative (IoT-) technologies related to smart infrastructure. Following the overall goal to reduce maintenance expenditure by making maintenance more predictable, several sensors were attached to a movable bridge. Whenever a ship passes the observed bridge, the roadway has to be lifted vertically, introducing a certain amount of material stress to the affected bridge elements. The impact of bridge movement as well as the weather's impact are measured by gauging the material strain and tilt angle of essential bridge parts.

Another aspect of the Smart Road project was to reduce energy consumption by controlling the road illumination in an on-demand fashion. In order to save energy, the deployed lighting posts sense the presence of pedestrians and switch their lights accordingly. Additionally, cyclists are detected and multiple lighting posts collaborate to illuminate the way that a cyclist takes, while saving energy behind the cyclist.

3.3 Organizational Context

Being the local port authority of the port of Hamburg, the Hamburg port authority is responsible for maintaining the infrastructure and assuring the infrastructure to be fit for use. Accordingly, traffic management is a key operational activity for the Hamburg Port Authority to maximize the infrastructure's efficiency. As roads are not the only traffic route within the port area, these tasks are also applied to railway and waterway.

The resulting challenge to coordinate the distinct traffic routes originates both from the unique characteristics of each route and also from the fact that the logistical interfaces between these routes are managed of other stakeholders, e.g. terminal operators. When a container enters the port by ship and leaves it by truck, then the HPA's responsibility is only to guide the ship, though the HPA cannot influence container terminal operation or truck coordination.

Currently, the HPA is exploring possibilities to integrate these activities and recent projects among the HPA and involved stakeholders may be linked to this higher level goal. The aforementioned projects create a basis for a large scale innovation process, shifting the HPA's focus from infrastructure maintenance to logistics coordination. Traffic ways within the harbor area connect the different stakeholders in logistics, and so might the HPA, eventually.

In addition, it is worth noting that the HPA is an "Anstalt des öffentlichen Rechts", a German organizational form strongly related to government authorities. Accordingly, the HPA is obliged to publish invitations to tender, resulting in a large variety of external companies carrying out their projects.

3.4 Consolidation Phase

Projects within the smartPORT Initiative were launched preceeding the international port conference IAPH 2015, fostering two major directions: smartPORT Logistics [HPA, 2015c] and smartPORT Energy [HPA, 2015a]. Scheduling these projects, the Hamburg Port Authority focused on delivering showcase scenarios for named conference instead of following their regular project management routine. Therefore, many of these innovational projects were not created in guidance of a thoroughly developed architecture or founded with detailed documentation, but rather in a way that would yield visible results at the fixed date of the conference, often in form of explorative prototypes or even mock-ups. As the research was initiated shortly after the conference, the Hamburg Port Authority now is in a phase of evaluating and consolidating their projects. The prior project origination was scarcely controlled and is known to enterprise architecture management by the term "bricolage" [Ciborra, 1992].

Currently, the Hamburg Port Authority tries to document the emerged enterprise architecture in order to increase business and IT alignment as well as providing transparency for personnel and external contractors. Following this, the documentation provides a basis for evaluating and further integrating project results. Accordingly, the the HPA surpassed the explorative phase and entered the consolidation phase, where the strategic view of enterprise architecture management helps in analyzing project results by providing a uniform view across projects and professional domains.

4 Action Design Research

This section outlines the applied methodology. Succeeding [Windelen, 2015] contextually, the focus of this thesis was to further investigate how the internet of things poses new concerns and requirements to enterprise architecture management. During the work on this thesis, the following four main tasks can be named. Firstly, an intensive literature research was conducted, to gather knowledge and form a theoretical basis of the concerning subjects. Secondly, interviews about explorative projects were conducted to gain insight about the specifics of these projects and possible architectural pitfalls. Over the span of this thesis, many documents of the Hamburg port authority and relevant entities of the HPA's business ecosystem were reviewed, including project documentations, use-cases and software architectures.

The employed action design research method is a combination of two research methods, action research and design research. For this thesis, constructing and

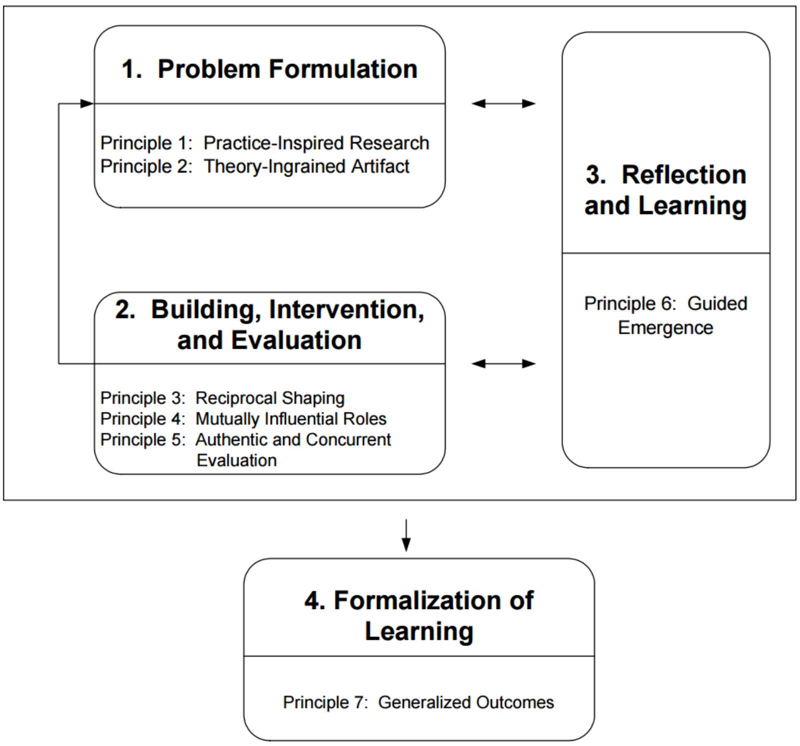

Figure 4.1: Stages and Principles of Action Design Research
Source: *[Sein et al., 2011, page 41]*

evaluating technology artifacts [Cole et al., 2005], as propagated in design research, was aided by practitioner-leveraging, change-oriented artifact shaping from action research [Avison et al., 2001]. This approach was enabled by the extensive collaboration with the Hamburg port authority and responds to the twofold need of making scientific contribution while solving specific organizational concerns posed by practitioners [Sein et al., 2011]. Following the stages and principles introduced in [Sein et al., 2011], ensemble artifacts were created which are listed separately in the succeeding sections, while this thesis addressed each stage of action design research depicted in figure 4.1 as follows.

Initiating this thesis, a vague problem was formulated by practitioners which is an anticipated trigger for initiating the first stage of action design research, the problem formulation stage. Coming from the Hamburg port authority, a practitioner, the posed problem was initially seen as sufficiently inspired by practice, therefore the focus was set on principle 2, creating theory-informed artifacts by conducting an intensive literature review on relevant topics.

The second stage leverages the narrowed problems formulated in stage 1. Iteratively, partial solutions were formulated, visualized by means of mock-ups and evaluated, while the overall process followed a rather IT-dominant [Sein et al., 2011] routine, which is depicted in figure 4.2. During this *Building, Intervention,*

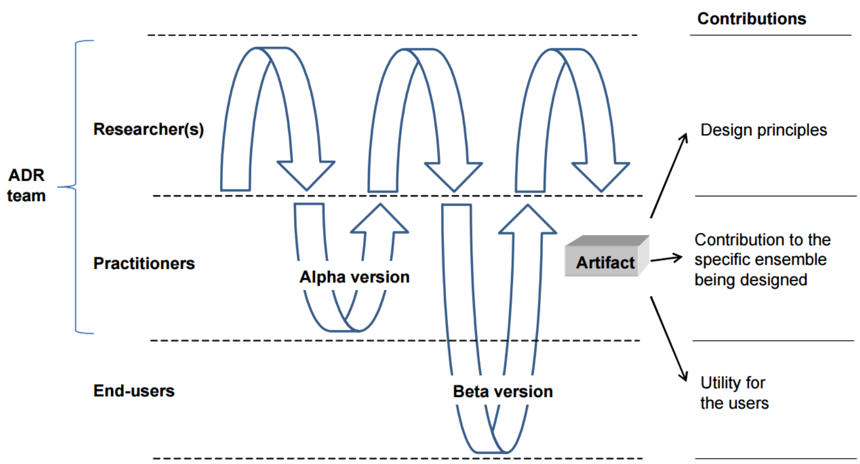

Figure 4.2: IT-dominant second stage of action design research
Source: [Sein et al., 2011, page 42]

and Evaluation (BIE) stage, the artifacts were reviewed and evaluated regularly in meetings with practitioners as well as researchers. It is worth noting that a routinely participating member of the HPA was ranked head of IT strategy, and multiple meetings included the HPA's CIO, which demonstrates the practitioner's commitment. Regarding the depicted end-user tests ("beta version") of artifacts, summative tests were conducted by means of both a mock-up of proposed artifacts was presented during later interviews, and an implementation project constructing a mock-up of a possible future application which simulatively leverages the data provided by instantiations of the proposed artifacts was presented.

According to the definition of stage three, "reflection and learning", is it necessary to apply the concepts and gathered knowledge to "a broader class of problems" [Sein et al., 2011]. This was approached by investigating multiple projects from varying professional domains implemented by distinct contractors/partners. Howevery, this allowed only for a small amount of generalization, so the proposed artifacts were also matched with IT landscapes found in IoT-related software architectures and grey literature, e.g. literature depicting network structures propagated by suppliers of IoT-related components.

To formalize the learning, "three levels of generalization" are proposed: "(1) generalization of the problem instance", "(2) generalization of the solution instance", and "(3) derivation of design principles from the design research outcomes" [Sein et al., 2011]. Accordingly, (1) the problem instance was partially generalized by continually matching the artifacts to different IoT-related projects from distinct professional domains, (2) the solution instance was abstracted from the specific needs of the HPA to a generalized solution relevant for practitioners facing the problem of the previously generalized class of problems, and (3) design principles were derived following the iterative "building, intervention and evaluation" cycles, forming a meta model, design guidelines and visualization proposals exemplified by visualization mock-ups, allowing the application of the gathered results to enterprise architecture management in other companies.

4.1 Artifact: Concerns and Vision

The initially formulated problem triggered the creation of this minor artifact. It comprises of concerns for an IoT-related enterprise architecture and a vision for a desired enterprise architecture. This approach is schematically following the architecture development method in TOGAF [Haren, 2011]. To comply with the principles of action design research, this artifact was not generated prior to architecture planning, but was revised and extended throughout the course of this thesis and formed both a basis for the other intended artifacts as well as providing means of assessing utility of the other created artifacts.

4.2 Artifact: Layers

To determine insufficiencies of prior approaches to enterprise architecture, a key activity was to model and survey the existing architecture using best-practice approaches. Early stages of assessing and modeling the the emerged enterprise architecture revealed that the number of types of *information systems* (IS) had increased in the observed IoT-related architecture, and other IoT-related projects were expected to further increase this number. Additionally, after creating layers (or classes) of architectural components, it was investigated whether different classes of components would benefit from specific levels of detail or whether gathered concerns (see section 4.1) needed unique attributes, e.g. references to other types objects. Therefore, the later on proposed smart brick paradigm was designed within this artifact.

4.3 Artifact: Meta Model

The emerging opportunities for change were collected in the meta model artifact. Being one of the main artifacts of this thesis, the meta model for enterprise architecture management aims to incorporate the specific needs of IoT-enhanced IT landscapes. The meta model can be seen as a design principle for other architecture instantiations, thus being an immediate outcome of the applied action design research method [Sein et al., 2011]. Accordingly, the meta model is expected to be the main theoretical contribution of this thesis. An early version of a meta model was proposed quite early and iteratively revised, adapted, extended and consolidated throughout the course of this work.

4.4 Artifact: Instantiated Architecture

In parallel to the other artifacts, a concrete architecture model for the reviewed IoT-related projects was implemented, forming the main output for the involved practitioner. While being part of the previously introduced artifacts, this artifact was separated to indicate its encompassing role. Following the action design research method, the created architecture description, as designed and shaped by the proposed meta model, does itself meet the requirements of forming an artifact, as it was directly triggered by a concretely formulated problem, underwent the proposed stages of artifact shaping and provides design principles for instantiating the created architecture description, which itself is an instantiation of the design principles posed by means of a meta model.

4.5 Interviews

During the course of this thesis, 15 informal interviews were conducted (for a list of interviews see appendix A.2), which can be separated into two groups of interviews. The first group is comprised of interviews in a semi-structured way where five students working on their master theses asked questions related to their thesis' specific focus. Forming a common link between these distinct theses, each interview examined one specific project. The second group of interviews leveraged results of prior findings and the emerging artifacts from the action design research process described in the preceding sections. For later interviews, the goal was to gather data to be modeled according to the respective development state of the meta model, as well as allowing formative evaluation of the development state of the meta model by matching utility and assessed fitness for purpose.

Since "experts" and employees in higher positions were interviewed, an informal interview style was chosen [Kornmeier, 2007]. Further dividing this style of interview, the conducted interviews can be separated into a first group of semi-structured interviews and a second group of in-depth interviews, both of which are types of qualitative research [Britten, 1995].

The semi-structured interviews were outlined by a questionnaire (see appendix A.1), while detailed questions turned out to be necessary as the interviewed employees did not have detailed knowledge of ever area appearing in the questionnaire.

The second group of interviews were structured around gathering architectural data for the proposed models. Since the designed meta model was continually refined, a significant portion of the data remains incomplete, as interviews were held at stages of the thesis where not all necessary attributes were yet identified. Therefore, this group of interviews had the very specific desired result of capturing architectural data for a concrete model. To outline the required data, mock-ups of architecture instantiations were shown, while the then following interview could not be structured in advance, as the contentual scope was scarcely known.

5 Capturing IoT-Aspects in EA Meta Model Extensions

The results section of this thesis offers results, their reasoning and points out differences to practiced or suggested approaches. Regarding the Enterprise Architecture Management introduces means of documentation and analysis. Yet, with the introduction of smart technologies the observed infrastructure landscapes appear to become increasingly heterogeneous and diverse, especially when considering the number of sensor types and innovative ways of leveraging sensor data. To address this complexity, this master thesis attempts to focus on infrastructural and integrating layers of the enterprise architecture and constructing a meaningful view of the gathered and relevant data.

Preceding this thesis, some architectural data was already gathered, though it proved to be less intuitively comprehensible than desired while still not depicting the necessary amount of information. The required levels of detail for different parts of the enterprise architecture were refined in accordance with requirements posed by influences of IoT-technologies, increasing device numbers and further changes of the IT-landscape during project advancements.

5.1 Concerns

During the conducted interviews, several concerns were gathered, formulated, reviewed and aligned. These concerns were posed by different persons representing a variety of different positions, departments or functions within the organization. While operative traffic managers preferred focusing on the different sources of information and their currentness, IT-strategists posed more concerns related to actual data requests of potential future applications. The latter is an interesting aspect, as the Hamburg Port Authority did launch projects aimed at increasing the (internal) visibility of newly introduced technologies, especially for less technology-based departments of the company. Mostly, these concerns can be attributed to high-level goals of enterprise architecture management, such as aligning business and IT, handling growing application landscapes [Matthes et al., 2008], identifying gaps and redundancies [Winter et al., 2010].

(C−1) What Services are supported by any specific information system, sensor or flow of information?

(C−2) Who decided to buy a specific sensor or infrastructure element and knows why it was chosen?

(C−3) Does a sensor need to be regularly recalibrated? When did the last recalibration take place?

(C−4) What information streams does any specific raw data stream contribute to?

(C−5) How important is any specific raw data stream for the services it contributes to? Is it more or less important than another raw data stream?

(C−6) Where is flowing information buffered or stored? How long is it stored? Where is it stored in terms of a geographic location or country?

(C−7) What information systems process certain flows of information?

(C−8) Is some transmitted or otherwise processed data related to persons or is it otherwise subject to governmental regulation in terms of privacy?

(C−9) Where is the source of any transmitted or processed data?

(C−10) Who is responsible for any specific transmitted or processed data, who can be consulted or needs to be informed in case of a disruptive incident?

(C−11) What is the data quality for any specific flow of information, or how accurately is it measured or derived?

(C−12) What information do certain services or systems yield?

(C−13) Is the physical object a sensor is attached to operative? (Counterexamples: In maintenance, broken or removed)

(C−14) Are physical objects that sensors are attached to subject to quality management, i.e. is it regularly maintained, does it have an expected lifetime or is it a mobile object?

(C−15) If a project uses data or information generated within other projects, then what projects is a certain project dependant on, and what projects depend on a certain project?

(C−16) What types of sensors are attached to certain types of physical objects?

(C−17) What types of physical objects is a certain type of sensor attached to?

(C−18) What sensors can be found within a specific geographic region?

(C−19) What data or information do I have about a certain area? (Or: Which raw data originates from a specific area, and what information is generated utilizing that data?)

(C−20) What project is any specific architectural artefact used in?

(C−21) When was an enterprise architecture artefact modelled?

(C−22) When was the currentness of an architectural artefact last checked?

(C−23) Who is responsible for maintaining any specific architectural artefact?

(C−24) Which are the 'important' information systems?

(C−25) Where does the simulation take place?

Some of these concerns are related to maintenance and project exploration, while others arose from discussions of possible future projects. Additionally, several showcases were discussed regarding visualization of modeled EA data to create a vision of possible future needs, i.e. what architectural data might be needed to support future visualizations of enterprise architectures.

Since these concerns were collected and iteratively extended, the created list can be matched against the proposed architecture. As part of a summative evaluation, the current state for each concern can be determined. Interestingly, as table 5.1 shows, 40% of the worded concerns are directly addressed by the proposed architecture extensions, while another 44% of the concerns were partially addressed or could be addressed by findings not incorporated into the designed meta model. The 4 remaining concerns, which are supported neither by the proposed exten-

Table 5.1: Evaluation of what concerns are answered throughout this thesis.

10 of 25	**Concerns directly addressed**		
	C–4, C–7, C–9, C–11, C–12, C–15, C–16, C–17, C–24, C–25		
6 of 25	**Concerns partially addressed**		
	C–5, C–8, C–10, C–13, C–14, C–19		
7 of 25	**Concerns answered within mentioned architecture extensions**		
	C–1	Service Application Layer	see section 7.1
	C–2, C–3, C–14	Maintenance View	see section 7.6.2
	C–6	Storage View	see section 7.6.1
	C–18, C–19	Sensor Locality View	[Windelen, 2015]

sions nor by mentioned possible future extensions, appear to be tool-specific and would supposedly be implemented as a feature in future versions of the modeling tool as they require attributes related to the management process rather than to managed artifacts (e.g. C–22).

5.2 Physical View

Prior to this thesis, the Hamburg port authority already developed a first architectural overview. The purpose of the work carried out was to gather an information basis and to explore and evaluate a specific enterprise architecture tool called iteraplan©by iteratec GmbH. During the course of this thesis, the new concerns were continuously matched with the existing landscape, refining the used models and aligning them. Throughout this process, it became apparent that the desired amount of information was difficult to display in one view as the information spans over many different scenarios, use-cases or architectures. The used modeling tool was based on a best-practice meta model and, by extension, on the essential layers of enterprise architecture management introduced in section 2.1.

The derived view for this thesis comprises information from multiple layers and was labeled “physical view”, as it focuses on physical components from sensor to data center and actual content of transmitted information, while enforcing a viewpoint of tracing flowing information from sensor to processing application. Additionally, the view separates the depicted objects into three specific classes of objects, ranging from technical components to software applications as needed, and sorting these groups into three layers. The layer dedicated to sensors introduces means of capturing the thing-centric IoT-paradigm (see section 2.3) by linking sensor types to physical object types. Additionally, the IT-infrastructure necessary to transmit data and information is also captured in a proposed fog layer, described and distinguished from other components. Eventually, flowing information reaches a proposed cloud layer, which typically contains software applications operated in data centers to perform complex computation with the gathered information to produce a form of information that is easily usable by other systems or, by extension, services.

5.3 Class and Instance Logic

Enterprise architecture management could document information on instance level in a senseful way, i.e. documenting every single existing component rather than aggregating them to classes of components by only documenting component types. The advantage on pursuing the instance logic is the exact basis of information which captures every possible aspect of the actual system, because it is an exact reflection of the enterprise's IT-landscape. Class logic, on the other hand, still finds an answer to most concerns, while requiring less expenditure on gathering, displaying and maintaining the architectural documentation. In the same way, it also allows for different yet more conclusive views, since the condensed view on the landscape enables the person looking at the information to see relevant correlations while not overtaxing their perception.

In this thesis, a mixed approach was found to be most reasonable to the subject. An instance logic is required at some level, but the vast number of components, especially in regard to the increasing numbers of sensors in internet of things projects, strains the applicability of the instance logic. Still, many concerns demand information only providable when documenting instances of objects. In the course of this thesis, a decision was made to capture every instance of data center level applications (cloud layer), while condensing systems in lower layers of the architecture to system types. To some degree, instances are required to resolve specific concerns, such as concern C-5 in section 5.1, where a quality-wise comparison between sensors of the same type is intended. Although data quality may vary, the data that each sensor emits was found not to differ between sensors of the same type (note: contradicting scenarios do seem plausible). In conclusion, the required view in regard to instance- or class-based notion depends both on the considered concerns and on the IT-landscape of an enterprise, and the mixed approach taken in this thesis still needs to be further refined.

As mentioned in sections 2.1 and 2.3, mixed approaches and missing clarifications of the applied logic do occur in research. Commonly, the enterprise architecture chooses a class-based view on the different architectures and leaves detailed descriptions to domain-specific architectures, dedicated databases or other means of documentation.

5.4 Layers: Smart Brick, Fog, Cloud

The existing landscape of enterprise architecture modeling tools provides an unstructured view of information systems in terms of each system's role. Mostly, the distinction of systems is based on their type (hardware, software, process, service, ...), and they are grouped by their projects. During the examination of the IT landscape at the Hamburg Port Authority it became apparent that a visualization should group systems by role rather than by project-affiliation. For the proposed view, this thesis suggests three modeling layers, or three roles for information systems. This logic not only allows for a more structured diagram, but also benefits consistency in modeling within each group of information systems and fosters computer assisted analysis.

During the conducted interviews, concerns were mentioned such as "Which are the 'important' information systems?", "Where does the simulation take place?". To answer concerns such as these, criteria for categorizing information systems were developed and applied. When checking the applied categorization with different employees at the Hamburg Port Authority, the proposed separation into three layers proved to be comprehensible, although cases emerged where distinguishing between cloud and fog layer or fog and smart brick layer seemed difficult. Therefore, a necessity to find detailed criteria for differentiating the information system's roles exists presumably. These criteria may not always match the opinion of every employee, but will probably foster a well-developed architecture by restraining inconsistencies among the documentation of different projects.

The following sub-sections will outline each proposed layer by defining its purpose, a designated level of detail, finding criteria to decide whether or not an information system is suitable to be displayed in this layer, defines responsibilities and common activities carried out by systems on the regarded layer and concludes by outlining prevalent characteristics of systems within the architectural

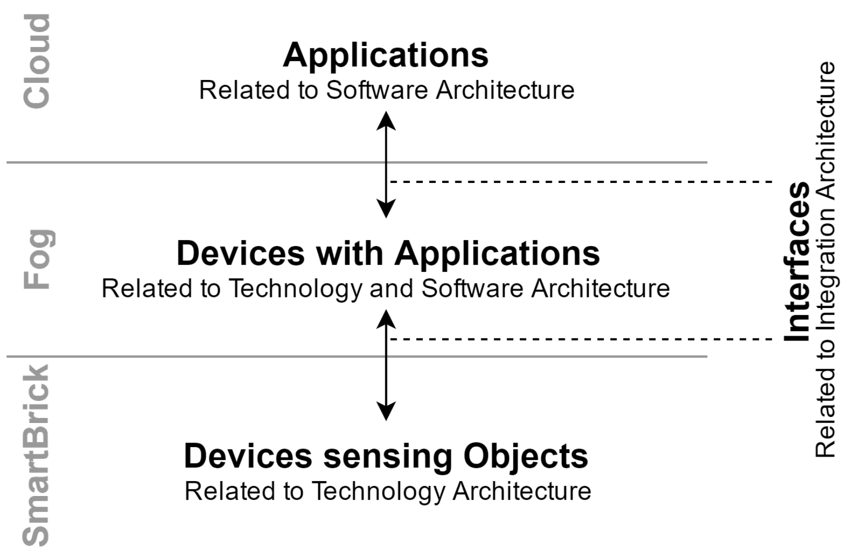

Figure 5.1: Relation of the proposed smart brick, fog and cloud layer to the essential layers of enterprise architecture management [Winter and Fischer, 2006].
Source*: Own work*

layer. Additionally, a connection to the essential architectural layers introduced in section 2.1 is explained, because each proposed layer addresses different types of architectural artifacts, as outlined in figure 5.1.

5.4.1 Cloud

The Cloud layer aims to resemble the group of information systems that are essentially generating and offering a generic type of information applicable to different scenarios. Generally, the cloud term used in this thesis applies to applications running in data centers, and the underlying hardware or physical location is of minor importance. This is supposedly not a redefinition of the cloud paradigm commonly used in IT, but a specific aspect of cloud systems. Using different classifications of cloud systems as discussed in section 2.6, it becomes apparent that a key aspect of the cloud paradigm stems from the separability of resource allocation and service provisioning. Following figure 2.5 (page 16), the aspect of a cloud system that would be captured by the proposed cloud layer is the interfacing application at the top of the depicted stack. Additionally, separately running software applications would be generalized to one cloud system if they inhere tight couplings and unifying means of interfacing the outside IT landscape. Conclusively, the proposed cloud layer is strongly linked to software architecture (see section 2.1) and resembles a partial yet unifying view of the employed software applications.

Level of Detail

For the cloud layer, the desired level of detail is to capture every instance of a cloud system without showing any details of the inner workings or the number of involved physical components. In this layer, multiple systems of the same project are only modeled separately if they are loosely coupled, show little dependencies towards each other, and both systems apply means of sophisticated information transformation or simulation. The goal is to have as few individual cloud systems as possible in the cloud layer while still showing most of the strategically relevant information. As an example, many cloud applications consist of different components with easily distinguishable roles running on different servers. This is quite interesting from a software architects point-of-view, but from a strategic perspective it is sufficient to depict one combining cloud system. Nonetheless, an interactive visualization could zoom into such an application and depict the main components of a cloud system. As an example, it would be helpful to zoom into a traffic analysis system and find a simulation-component, but a detailed description of dozens of components, classes and interfaces is probably not strategically relevant.

Criteria and Differentiation

Cloud systems are strongly aggregated systems, commonly composed of multiple applications targeted to support a unifying set of services. A cloud system is in many cases the most sophisticated information system within a project (exceptions exist, see the fog computing paradigm in section 2.4), while there is no strict coupling between project and cloud system - i.e. projects may comprise any number of cloud systems or no cloud system whatsoever. The task of a cloud system is generally to evaluate data, aggregated data and information to generate new information by means of intensive computation or simulation. It rarely matters where a system of the proposed cloud layer is physically located, there is even little impact to be expected if the concerned application runs on outsourced hardware hosted by an external provider. Accordingly, there exists a significant relation between cloud systems, applications of cloud systems and software architecture, while there are only loose couplings to the underlying hardware.

Responsibilities and Characteristics

Cloud systems are intended to gather information, generate information services and provide them at cloud level for use in other applications. Any project that provides information generated by means of computation would commonly offer the computed data through a cloud system providing an information service. Typically, communication between projects is implemented at cloud level, as most projects only leverage interpreted data rather than gathering raw data from sensors employed in other projects. This is especially true for the HPA, where the projects were implemented not too long ago and knowledge exchange between projects is in its beginnings.

5.4.2 Fog

The chosen terminology of the fog layer was originally inspired by publications of Cisco [Cisco, 2015], who introduced it not as a layer for enterprise architectures but as a component class in IoT-related network diagrams. It is strongly related to fog computing (see section 2.4) where aspects of cloud systems are sourced to the network layer. Therefore, the fog layer incorporates some aspects of cloud systems and is charged with networking-related tasks. Systems in the fog layer are typically customized networking components that either route incoming information to other systems or reduce the network load by buffering or aggregating the incoming data before it is forwarded. As the fog layer is seen in the enterprise architecture management context, there is no intent to require every single type of networking component to be modeled.

Level of Detail

Within the fog layer, it is desirable to see every type of system that does anything to the forwarded data, i.e. data aggregation or buffering, or that fulfills an accentuated role, such as being a gateway to wireless or public networks, applying a new addressing scheme to forwarded data, or implementing means of sensor management logic. On a strategic level, it is not necessary to see instances of fog system types in the enterprise architecture. Being related to networking, fog system types are closely related to types of technical components implemented in the field. This contrasts the desired level of detail on the cloud layer, where the number of modeled systems is kept as small as possible. Since fog systems are commonly connecting few types of components with scarcely occurring links to fog systems in other projects, large numbers of fog systems would not overly diminish comprehensibility. Inner components of fog systems do not need to be displayed in a full-landscape view and are typically very specialized, as most fog systems are dedicated technical components with specifically tailored computational capacities. Still, some innovative fog system types were found that run on standardized hardware, for example a single-board computer anonymizing video streams, so in some situations it is more relevant to document software aspects. Therefore, the fog layer is related to both the technology and software architecture, as the incorporated role of a fog system may stem from either its technological capabilities or from the employed software.

Criteria and Differentiation

Fog systems can be described as components that qualify neither as a cloud system nor as a sensor. They commonly perform tasks that do not require large amounts of computational capabilities, for example data routing, detecting failures of attached sensors, buffering incoming data or information, or aggregating information, though this may change over time, as discussed in 2.4. A common goal of computation in fog systems is to reduce the network load by decreasing the amount of forwarded data. System can also be modeled at the fog layer if they are of strategic importance, even if they do not directly modify the routed data. As an example, it may be strategically relevant to model a wireless router due to the implications on latency, reliability and privacy, although such a device does little more than unfiltered data forwarding.

Responsibilities and Characteristics

Most essentially, fog systems are necessary technical components to ensure connectivity. Additionally, they also incorporate activities related to reducing the network load, which typically includes routing, buffering and aggregation. In addition, they are often assigned tasks of sensor management, i.e. noticing failure of attached sensors. Commonly, it is more efficient to supply fog systems with necessary computational resources than to equip every single sensor with additional hardware. Accordingly, increasing smartness of sensory devices and increasing transmission bandwidths may eventually reduce the need of fog systems.

5.4.3 smart brick

The smart brick layer is intended to depict implemented sensor types [Schirmer et al., 2016]. Yet, only depicting classes of sensors did not provide a sufficient data basis for the ascertained concerns. On the other hand, depicting instances

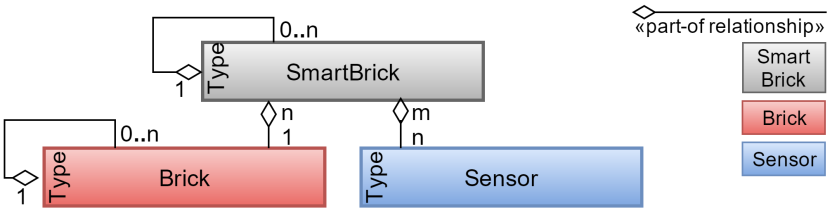

Figure 5.2: smart bricks are composed of a hierarchy of bricks (physical objects) and attached sensors.
Source: *[Schirmer et al., 2016]*

of sensors, like the cloud layer contains instances of cloud systems, is not feasible as the vast number of sensors would profoundly hinder comprehensibility. Throughout this thesis, a smart brick logic was shaped, which expresses the relation of sensors and physical objects ("bricks"). Existing systems for managing

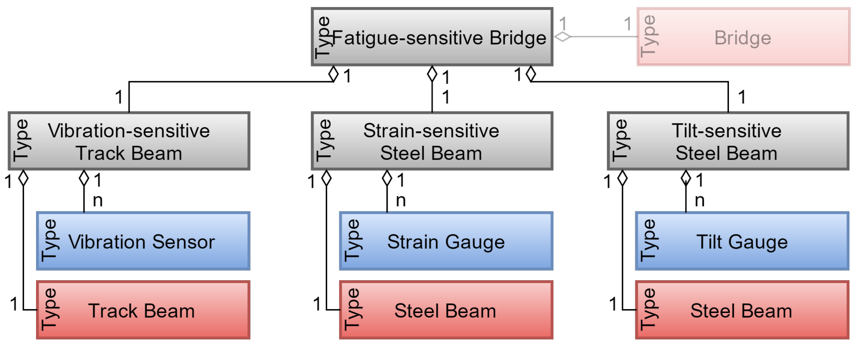

Figure 5.3: An IoT-enhanced (fatigue-sensitive) bridge modeled according to the smart brick logic.
Source: *[Schirmer et al., 2016]*

bricks, or physical objects, commonly depict them as a hierarchy of physical objects, e.g. a bridge is comprised of bridge posts, which are composed of a concrete socket and steel beams. In the same way, a smart bridge having multiple sensors could be broken down into smart bridge parts comprising a subset of attached sensors. As depicted in figure 5.2, this is modeled by allowing smart bricks to be composed of other smart bricks, while the hierarchical characteristics are captured in integrity conditions. If a smart brick is part of another smart brick, then the attached sensors must be a subset of the sensors attached to the parent smart brick, and the linked brick must be a sub-brick of the parent smart brick's brick. This principle is visualized in figure 5.3, showing how the comprising smart brick "Fatigue-sensitive Bridge" is an extension of the brick (physical object) "bridge", while the subordinate smart bricks are "attached" to parts of this bridge.

However, "attaching" model-parts is not related to the physical attachment of a sensor instance to an object. Instead, the relationship is intended to reflect the logical relation of the generated data, i.e. the type of "thing" or physical object that generated data is related to. As an example, a camera for counting passing vehicles may be physically attached to a lamppost or other infrastructural objects, but the modeled smart brick would "attach" this camera to a road segment, as link between smart brick and brick models the logical locality reference of the employed sensors.

Regarding other architectural layers, the smart brick logic is related to the technology architecture, but also to classes of physical objects. The latter is not yet captured in any form of database, as it was not yet part of strategic or operative management. Types of things, from a thing-centric vision of an IoT-landscape, were formulated to be relevant to other parts of strategic management, as described in section 2.3. This is specifically referring to [Meyer et al., 2015], where "things" in process modeling are attributed with "isMultiInstance" (multiple instances of a system can be assumed to refer to a system type in this context).

Level of Detail

The smart brick layer is intended to view every type of smart brick, i.e. the same sensor type can be used in multiple smart brick types and thus be modeled multiple times. The intention is to depict how a sensor is used instead of displaying the plain sensor, which is motivated by two developments.

Firstly, following the internet of things paradigm, new types of sensors are installed and employed sensors may be used in novel ways, resulting in increasing sensor numbers, sensor roles and especially in multiple sensors observing the same objects or parts of the same object. Smart bricks capture this by forming hierarchies of smart bricks with descriptive names, thus hiding complexity by condensing the displayed information.

Secondly, the same sensor can be used with different strategies in mind, i.e. using a video camera to identify license plates at a parking lot entrance or using video cameras similarly to 3d-scanners to survey area occupation on the same parking lot. This aspect is captured by letting multiple smart brick types leverage the same sensor type while describing the logical object relation of generated data.

To foster comprehensibility of such projects, multiple smart brick types comprising the same sensor and brick types can be modeled with descriptive names. Therefore, the desired level of detail on the smart brick layer is to be as detailed as possible and offers ways to be considerably more detailed than existing architectural models that only depict sensor types. If certain areas of the enterprise architecture become too detailed, the complexity can be hidden in smart brick-hierarchies, i.e. several related smart brick types can be condensed to a root smart brick comprising the previously modeled smart bricks.

Criteria / Differentiation

All systems that posses at least one type of sensor can be modeled using smart bricks. It is possible that a system that incorporates a sensor also acts as a network node for another smart brick type, thus also qualifying to be a fog system. In these cases, both aspects of the same system are modeled separately to indicate their strategic application. Accordingly, all modeled sensor types are always a component of a smart brick (1..n to 1..n relationship), while some information systems are modeled both as a fog system and as a smart brick, thus introducing a small amount of redundancy on the technological level.

Responsibilities and Characteristics

Smart bricks are "smart objects" and accordingly generate raw data or information, depending on the computational capabilities of the employed (sensory-) systems. They form a level of abstraction from the technical components to foster comprehensibility for stakeholders that do not have detailed knowledge of the related project. In terms of the internet of things, the goal of smart bricks is to constitute the relation between sensor and "thing" (see section 2.3), thus depicting the role, purpose, intention or coverage of a sensor type.

5.5 Flow of Information

Information flows between different systems and applications, which is an aspect that is already modeled in existing enterprise architecture frameworks and is part of the integration architecture. Still, flows of information have not been

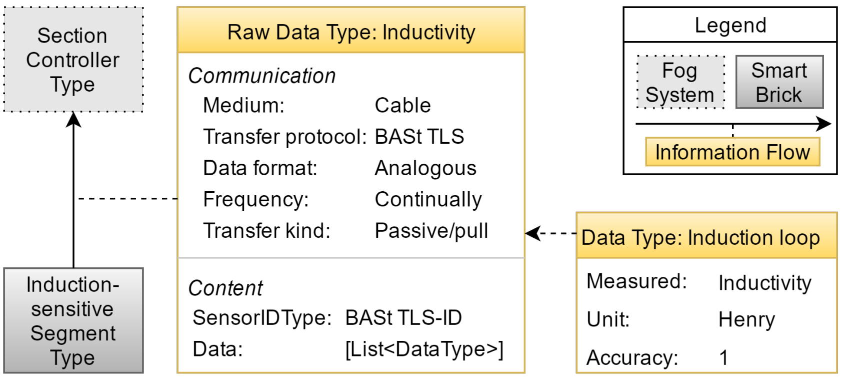

Figure 5.4: Flow of Information: Example for a raw data stream flowing from an induction loop to a section controller.
Source: *Own work*

standardized in terms of attributes, i.e. the only information modeled was the two linked information systems, the direction of the flow of information and possibly other generic information [Hanschke, 2009, page 121]. The transferred data

(*data type* in figure 5.4) is often called business object [Matthes et al., 2016] or information object [Hanschke, 2009, page 123]. Although enterprise architecture management tools allow attribute creation for interfaces and information or business objects, the nature and content of these objects still needs to be captured in a uniform way. This basic information about the flowing information is sometimes documented for specific systems or projects in the related documentation, i.e. for cloud systems this information can commonly be found in the software architecture description. Especially systems that stem from specialized, non-IT related projects, such as induction loops installed in roads, this information is not captured within any software architecture description. Therefore, no centralized and detailed source of information for these artifacts exists yet. Since information flows are closely linked to types of sensors or types of fog systems, i.e. the type of information transmitted scarcely differs among the same type of sensor, the enterprise architecture seems to be a good fit for a slightly more detailed description of flows of information.

As depicted in figure 5.4, the proposed attributes for information flows can be separated into two groups. The first group is concerned to *how* the information is transferred (communication), while the second group comprises details about *what* is transferred (content). Additionally, information flows transport data types (previously information objects or business objects), which are further described by attributes identifying how to interpret the transferred data. They were named "data types" to clarify their intention, as business objects are also used in other contexts.

5.5.1 Communication Attributes

Regarding the communication aspect, it can be relevant to know what medium is used. Depending on the scenario, a distinction between types of cables (i.e. internal, contracted, external or public/internet) can be made, or between cable-bound and wireless communication. The transferred data is formatted both on the transport and the data level. The transport protocol refers to the OSI transport-layer and below [Zimmermann, 1980], while the data format refers to the protocol used at the application layer. Additionally, the frequency of communication is documented, while the proposed transfer kind is a simplification of a mechanism already present in current architectural models. A passive (pull) request transfers data from the source to the target but is initiated by the target, commonly by means of a request being sent from the supposed target to the source. This could also be modeled in existing frameworks by attributing the transferred business object a custom direction value [Matthes et al., 2016].

5.5.2 Content Attributes

The content section of the information flow is essentially a collection of data types, or information objects (business objects) as mentioned above. In addition, the generic attribute *SensorIDType* is introduced to document how the reference between data and sensor is identified and managed. Each data type attached to an information flow is composed of an identifying name, a description of what is measured, the unit of the measurement and an optional attribute indicating the sensor accuracy. The significance of these attributes can be motivated with an example from the surveyed projects. Within a system calculating the travel times for road segments, the returned travel time was stored as a floating point number with double precision. It seemed obvious for the personnel of other projects that the travel time was measured in fractions of minutes, although it was actually the raw output of a simulation model computing in fractions of a second.

5.5.3 Relationships between Information Flows

There is a relationship between different flows of information. Information generated by systems above the smart brick layer always leverages sensory data. In other words, every flow of information either originates from a smart brick, or is generated through computing or aggregating other flows of information. This dependency was implicitly stated in section 5.4.2, where fog systems were denoted to forward data. The semantics of such links could partially be captured by noting the project affinity of information streams, which was actually the case in early iterations of the information flow artifact.

Information system dependencies also suggests a distinction between different types of information flows, which are divided into raw data streams, information streams and information services in this thesis. A raw data stream stems from a smart brick, thus not depending on any other flow of information. Accordingly, Information streams necessarily depend on other information streams or raw data streams. Additionally, the information service type was introduced, which was intended to distinguish between forwarded, buffered and aggregated information streams on the one hand, and information that was created by means of sophistication algorithms in a cloud system on the other hand. Information streams present a class of information stream that can be described with the same attributes but plays a highlighted role in the enterprise context.

Dependencies between information flows proved to be highly relevant for certain concerns, e.g. concern C-5 in section 5.1, where the contributing raw data streams for a specified information service need to be found. Such concerns cannot be answered by any means until this relationship between information flows is documented.

Additionally, tracing information flow dependencies allows to automatically create meaningful cut-outs of large enterprise architecture models. Currently, enterprise architecture management tools can trace dependencies between information systems. Due to the increasing numbers of connections between systems, this no longer suffices for visualizing partial architectures. If an information stream about parking lot occupancy is regarded, then systems related to railroad switches do not need to be depicted, even though an interface between the railroad system and the traffic management system may exist. If a modeling tool has knowledge about dependencies between information flows, then it is enabled to automatically discard irrelevant dependencies / interfaces.

5.6 Metamodell

Forming the basic design principles of the before mentioned aspects, the meta model depicted in figure 5.5 is the template for modeling any specific enterprise

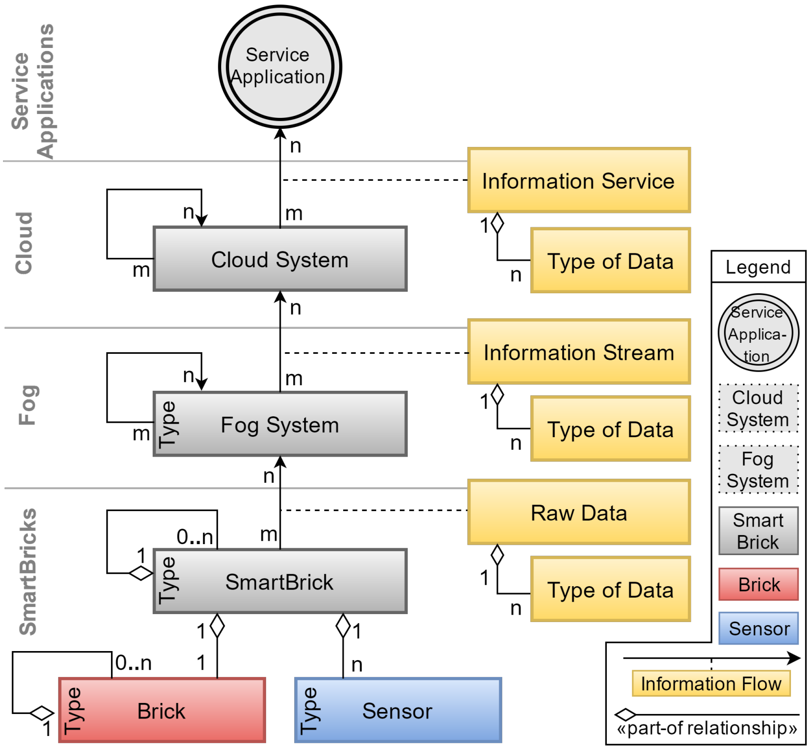

Figure 5.5: The meta model created for enterprise architecture management.
Source: *[Schirmer et al., 2016]*

architecture. The proposed layers cloud, fog and smart brick are grouped vertically with an additional layer at the top, named service applications. This layer is depicted in figure 5.5 but is subject to further research and will not be covered within this thesis. It is merely a hint about what information is to be expected at layers above the proposed cloud layer.

At the smart brick layer, the aforementioned model (see section 5.4.3) is shown in an expanded view, displaying the part-of relationship between the components. Originating from smart bricks, the raw data flows to connected fog systems. Similarly, information streams flow between fog and cloud systems, and information streams have its source in cloud systems, as explained in section 5.5.3.

5.7 Mapping to EAM Tool (iteraplan)

The architectural models introduced in this chapter are intended to be used in practice at the Hamburg port authority. Therefore, a mapping between the proposed models and the available mechanisms in the HPA's modeling tool were needed. As the HPA uses iteraplan, an enterprise architecture management tool

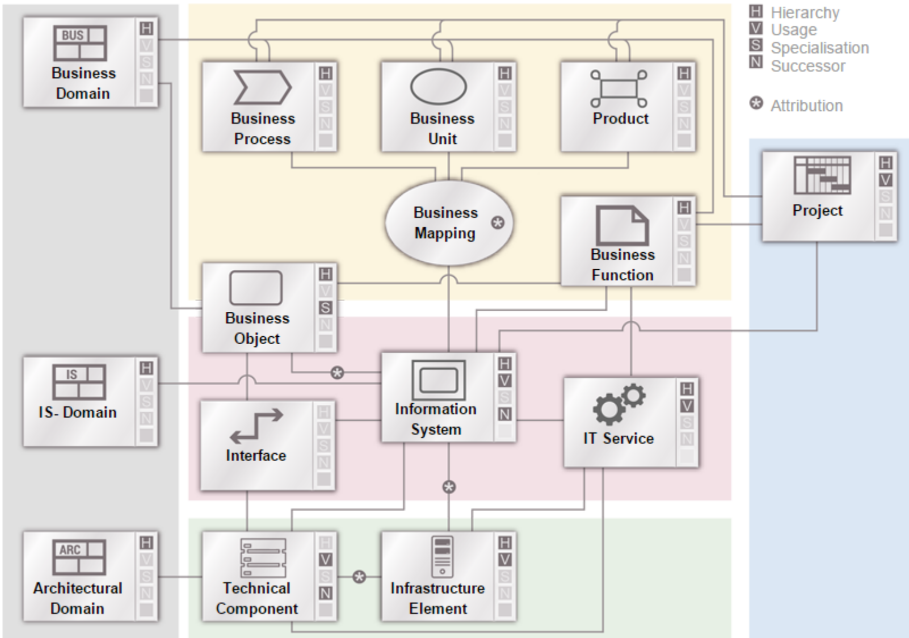

Figure 5.6: The enterprise architecture model used in iteraplan.
Source: *©iteratec GmbH 2016, Build ID: Corporate Edition v5.2.0-r25486 Public Demo System https://www.iteraplan.de/iteraplan%5Fee%5Frelease/, accessed 2016-05-30*

created iteratec GmbH, the model to map against is the "best practice enterprise architecture" [Hanschke, 2009, page 65]. Due to the limitations of the available enterprise architecture model depicted in figure 5.6, the extensions proposed in this thesis focus on "Information Systems", "Interfaces" and "Business Objects". Although these were not intended to be used in such a way, they do provide the mechanisms necessary for resembling the introduced meta model.

5.7.1 Layers: Cloud, Fog and smart brick

All systems from smart brick to cloud system are modeled as information systems. This allows the use of interfaces (and business objects) to consistently model connections between systems and applications. To locate each system at the appropriate layer, an enumerating attribute can be used to place any information system at a specific layer. By not setting this attribute, systems are not directly placed at the specific layer, which is relevant for hierarchical smart bricks or components of cloud and fog systems. Not every modeled information system is supposed to be directly placed on one of the layers in a visualization.

5.7.2 Information Flows

In iteraplan, an interface can connect two information systems in a directed way. Therefore, informations flows are represented by interfaces. Additionally, the model requires additional data objects to be linked to information flows. In iteraplan, business objects can be modeled to be transported by an interface. Therefore, transmitted data is modeled as business objects. Yet, the desired link between different information flows cannot be modeled directly. The "is based on"-relationship between two information flows was chosen by the practitioner to be modeled by inserting a text field attribute to interfaces named "based on", which contains the assigned iteraplan-IDs of contributing information flows. A more suitable representation of the relationship between interfaces was not found in iteraplan.

5.7.3 smart bricks

In accordance with section 5.7.1, smart bricks are modeled as information systems in the practitioner's modeling tool. This decision was made due to the distinct usage of technical components at the HPA, which would generally be the target architectural layer for smart bricks. In iteraplan, the underlying meta model for technical components is similar to that for information systems, but is slightly more limited. Information systems can be assigned business objects to indicate their utilization, and can also be linked to higher layers of the enterprise architecture, i.e. business functions and business mappings. These features are not yet required by smart bricks, so the only practical difference is that the current form of visualization, an information flow diagram, would not display technical components.

A smart brick is modeled as an Information System containing all attached sensors in a "used by"-relationship. A clear distinction to a "part of"-relationship can be made, as this type of relationship would prohibit the use of leveraged sensors in other smart bricks (1..n relationship instead of m..n). Although a sensor instance is always a part of a smart brick instance, the type-level (class-) logic intentionally used in enterprise architecture management requires sensors to be represented as a contributing system, not as an exclusively incorporated system.

The smart brick's unique relationship to bricks ("things", physical objects) is represented by a simple text attribute. This is due to the lack of object-type databases or strategic management ob object types in other systems. Therefore, no references to other systems can be made, and object relations are formed by a natural language description of the linked brick. Regarding the current use of smart bricks, this logical object reference suffices to fulfill the formulated goal of describing the "thing"-aspect of internet of things systems.

6 Derived Models and Evaluation

Due to the confidential nature of the detailed project discussion in this chapter, the contents were moved to the non-public appendix of this thesis.

Da dieses Kapitel vertrauliche Daten enthält, befindet es sich im nicht-öffentlichen Anhang der Arbeit.

7 Discussion, Conclusion and Outlook

The impact of the *Internet of Things* (IoT) to *enterprise architecture management* (EAM) appears to stem mostly from the increasing number of devices and their connections (see section 1.1), as well as from the necessity of incorporating relationships to physical objects into strategic management (see section 2.3). This twofold requirement was both mentioned in literature and observed at the practitioner.

The link to physical objects was addressed by depicting the "thing"-relation of generated data in form of smart bricks (see section 5.4.3). However, some aspects are still difficult to depict, such as "environmental sensors" [Haller, 2010]. Also, it is worth noting that "things" were seen in a very physical way at the practitioners business environment, hence the name "brick" for physical objects. Other companies may find the concept of smart bricks helpful, but may shift the logic to virtual entities of interest, following the "thing"-definition in [Haller, 2010]. Future research may identify the need to incorporate physical objects to other layers of the enterprise architecture management, as it may also be relevant to outline object-relationships in information flows (integration architecture), services and processes. Similar to how smart bricks link sensors to sensed entities, it might be relevant to link flowing information (interface) or data (business/information object) to logically related objects, thus representing the logical object relation of transmitted information.

The aspect of increasing device numbers and accordingly increasing complexity in IT-infrastructure is addressed by choosing different levels of detail in previously used EAM concepts. A partially novel view has been proposed (see section 5.2) which attempts to create a more strategically helpful link between technology, software and integration architecture by neglecting some details from both the technology and software architecture (e.g. hiding the technological basis of cloud systems and the software running on sensors). Hidden details are expected to be visualized on demand, i.e. allowing users of enterprise architecture management tools to show details of depicted systems interactively. This mechanism was consistently transported to the integration architecture, which is required to incorporate more information than previously necessary.

Although only one practitioner was observed, the results can be considered partially generalizable. The investigated projects were led by different project managers and from different professional domains, such as traffic management, infrastructure maintenance and railway management. Partial architectural models were presented at later interviews and the feedback as well as gathered project data were used to check the fitness for purpose and to further adjust the underlying model. Seeing that several distinct projects fit into the proposed models and each benefit in terms of comprehensibility, it does seem feasible to utilize the model in other enterprises. Due to the lack of concrete architectural proposals in the literature, the emerged architecture model should be seen as a basis for discussion and further research is needed, especially by applying the model to other entrepreneurial environments and evaluating the results.

7.1 Service Application Layer

Although the service application layer is intentionally not conceptualized in this thesis, the prePORT parking project indicates that there is a relation between the smart brick layer and layers above the cloud layer. As depicted in figure 7.1, the touch screen used for enrolling in a parking lane is modeled by means of a smart brick, while the corresponding service is linked to the connected car platform on a functional level, even though it is performed using the touch display. This logical connection between the functional target of a service and the input point for service interaction could be one of the key aspects of a layer somewhere above the proposed cloud layer. Albeit meaningful, a layered visualization of the architecture as depicted throughout this thesis inhibits a meaningful visualization

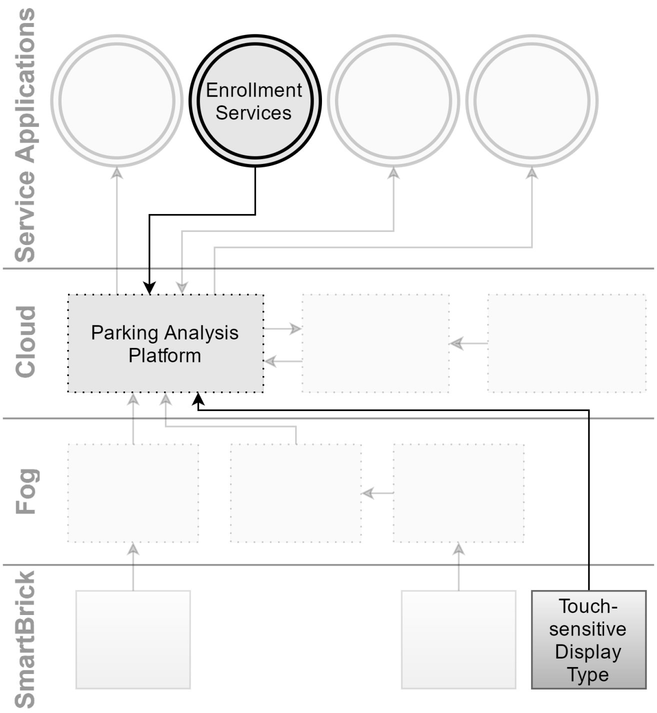

Figure 7.1: Project prePORT parking: Enrollment is captured both as a service application and as an actuator in form of a smart brick.
***Source**: Own work*

of the aforementioned connection between a service-related layer and the smart brick layer at the bottom of the created visualization. This could only be implemented leveraging interactive visualization components, which might prove to be an important area of interest within enterprise architecture management.

7.2 Fog Computing

The fog computing paradigm discussed in section 2.4 appears to be an emerging research discipline and was found to pose unique requirements to enterprise architecture management. For most fog systems, it is strategically relevant to outline their unifying role of bringing aspects of cloud systems physically closer to employed sensors. The future of this aspect is unclear, as single-board computers merge fog system and sensors, whereas increasing capacities of communication hardware diminishes the necessity to deploy additional systems outside of data centers. For managing currently emerging IoT-enhanced enterprise architectures, the fog-aspect was found to be relevant. Discussions with interview partners outlines their importance not only for understanding the current IT-infrastructure, but also for identifying unused potential by documenting customizable firmware and software in fog systems. Currently, the observed practitioner did only leverage inter-project communication at cloud-level, as connections at lower levels require detailed knowledge about the fog layer.

7.3 Stage of IoT and Future Architectures

Studies suggest that the amounts of data handled by enterprises grows exponentially [Gantz and Reinsel, 2011], and following the internet of things paradigm, the numbers of sensors and sensor types increases [Gartner, 2015]. Although the discussed fog computing paradigm attempts to reduce the network load (see section 2.4), the amounts of data, data types and attempts of storing said data for statistical evaluation grow [Chen et al., 2012]. Necessarily, the underlying infrastructure for data transport also increases. Some other, unrelated technologies facing growing capabilities in data transport did at some point of time converge to standardized interfaces incorporating most of the emerged device classes. The most prominent advocates of such developments are USB and PCIe, both of which introduced a certain overhead by introducing an additional layer in communication. USB for example provides a standardized set of device classifications and mechanisms for automatic integration of unknown devices. PCIe introduced a generalized internal link, and current CPU architectures can be observed to offer fewer types of interfaces, yet increasing numbers of PCIe-lanes [Intel, 2015].

Although it is unclear when this will occur, IoT-releated system landscapes may at some point be subject to similar developments, where communication between the systems is standardized, classes of sensor types are standardized and introducing new sensors to the infrastructure is little more effort than plugging it in. Currently, such solutions are isolated applications, an example could be multiroom sound systems with mobile app integration. In addition, a convergence or standardization in data transport might be seen in increasing numbers of devices directly connected to TCP/IP-networks (e.g. IP-cameras, IP-enabled weather stations, ...).

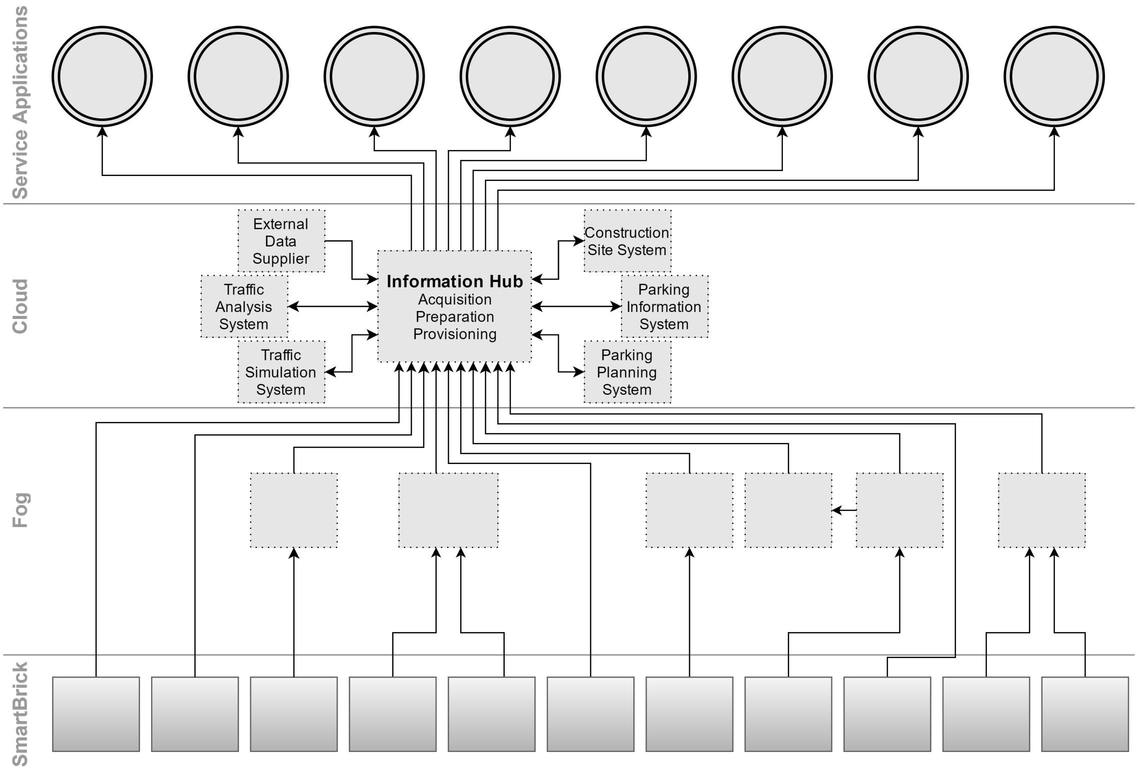

Figure 7.2: A possible future enterprise architecture featuring a dedicated information hub.

***Source**: Own work*

When viewing current company take-overs of major companies engaged in networking technologies with interest in IoT-related topics, it appears that such developments are already in their beginnings. Cisco acquired Jasper [Cisco, 2016a], a cloud platform for centralized control of IoT devices. SAP's strategy to build a single platform gathering all data [SAP, 2016] and the HPA's emergence of a hub-system at cloud level (see section **??**) foster the vision of an architecture where all (IoT-) data is gathered and distributed from a central system, as sketched in figure 7.2.

Regarding data acquisition, this concept is not new, as the "separation of concerns" plays a major role in software architecture [Soni et al., 1995] and some current micro-service based software architectures specifically include components

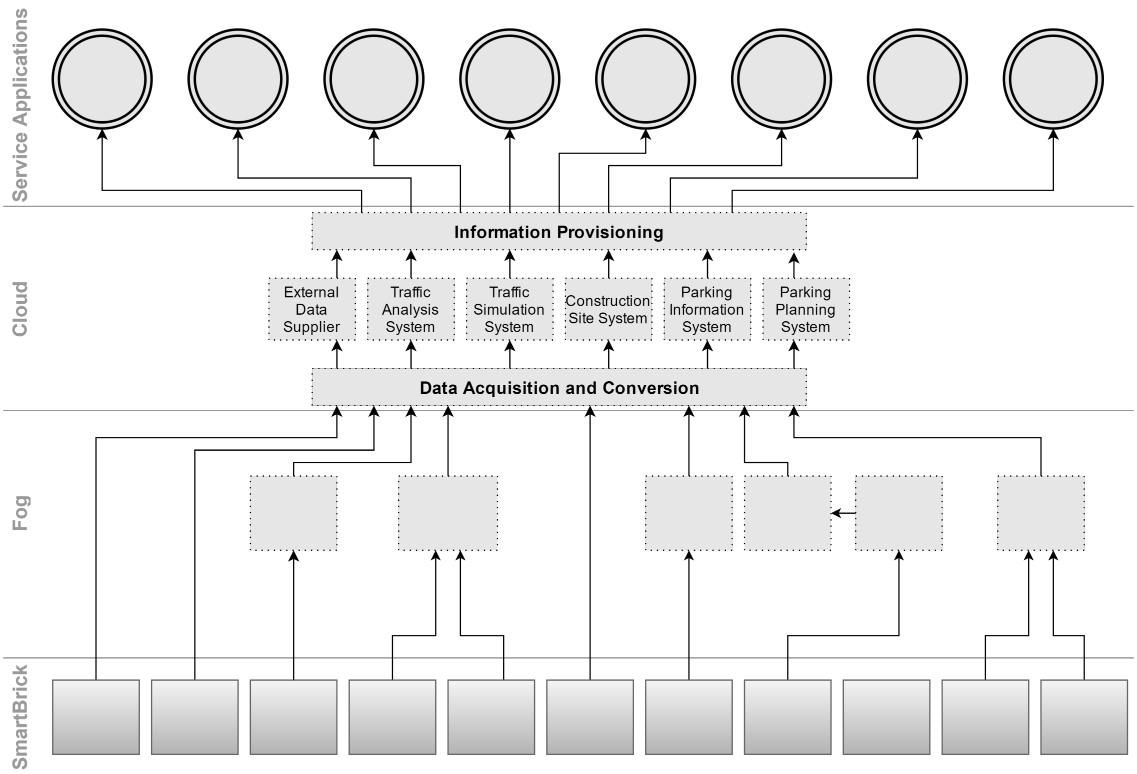

Figure 7.3: A to-be enterprise architecture with unifying data acquisition and information provisioning.
***Source**: Modeled following [Schirmer et al., 2016]*

for data acquisition and data preparation [Namiot and Sneps-Sneppe, 2014]. Still, these concepts were found to be implemented only at project level at the HPA and probably many other companies, while the aforementioned line of argument suggests that an enterprise-wide solution would be beneficial.

Another way of formalizing information flow at cloud level would be to create common interfaces for information entering and leaving the cloud layer [Schirmer et al., 2016], as depicted in figure 7.3. This is a relatively direct generalization of common component structures within many projects. Still, this might be less viable for other companies, as it is focused around the HPA's IT-landscape, where communication between cloud systems is yet at its beginnings. Possibly, some more developed companies might face IT-landscapes where data is refined in a process through two or more successive cloud systems, creating links from the depicted information provisioning component to the data acquisition component.

Accordingly, a necessity exists for cloud systems to convert their data in formats known to other systems. The other apparent necessity exists in offering a centralized platform for information provisioning - not necessarily for providing information to services, but for "publishing" information to any other system that is interested. Such a platform has the advantage of allowing hardware abstraction as well as generic hardware decisions applicable for all information available in an enterprise, e.g. saving said information in fast but volatile memory (in-memory database) or managing data archiving and data safety topics such as geographic redundancy in a centralized fashion. Another aspect of centralization is the ability to utilize data more easily, fostering project agility, and allowing for data-subscriptions and payment models, thus offering new business models.

Following this line of argument, a future architecture might feature one capital information hub where a centralized data management handles subscriptions, payment models, types of storage (disk vs. in-memory) per data type, archiving, risk management (data safety) and possibly security or privacy related concerns. Additionally, many small information hubs may emerge due to the effects of fog computing (localized computation) and increasing agility in creating virtualized hardware capabilities [Felter et al., 2015]. These small information hubs may over time merge with each other or be incorporated into the main information hub.

7.4 Further Influencing Factors

Shortening life cycles and "volatile markets" call for more agility in many different enterprise domains [Christopher, 2000]. The expected impact on IoT-related sensor deployment is that agile project teams introduce new sensors with scarce IT-strategy-alignment, which was already observed at the HPA and is strongly related to the term bricolage [Fuglsang and Sørensen, 2011]. Additionally, agile organizational structures intentionally reduce the dependencies between project teams [Kniberg and Ivarsson, 2012], thus encouraging more independent and accordingly more rapid deployment of systems, especially systems with little inter-project dependencies such as sensors. Therefore, enterprise architecture management can be expected to eventually adapt to shorter system lifecycles and less centralized system deployment.

A possible answer to these requirements could be automation of enterprise architecture management processes [Farwick et al., 2012], i.e. automated integration of new devices into enterprise architecture tools. As a first step, such systems should be automatically added to systems capturing actual instances such as the Configuration Management Database from ITIL®. While standardized protocols do not yet exist, solutions are already available (e.g. [ServiceNow, 2016]), and such developments could be accelerated by improvements in artificial intelligence (e.g. [SAP and IBM, 2016]). Accordingly, a possible key task for enterprise architecture management could be to incorporate abstracted data from an automatically populated CMDB, although it is questionable if a technological separation between EA-tool and CMDB would persist when automated system discovery was introduced.

7.5 Relation to CMDB

Although the CMDB is not directly a part of enterprise architecture management [Correia et al., 2009], there is a relation between the two, as explained in section 2.1. Since the technological architecture layer possesses patterns of systems managed in a CMDB, it is easily comprehensible that the EA holds an abstracted and/or aggregated view of the CMDB. Accordingly, if this master thesis proposes a link between sensors and physical objects, forming smart bricks (see section 5.4.3), then this conceptual link is also relevant for a possible CMDB. Regarding the proposed smart brick logic, the equivalent data in a CMDB would hold information about instances, i.e. what sensor (instance, not type) is attached to what physical object (again instance, not type). Since the EA is supposed to capture patterns from a CMDB, the CMDB could also add additional information, e.g. where a specific sensor is attached on a given physical object or where an observed object is located. Feedback to enterprise architecture management is plausible, for example deriving information about whether or not certain sensor types are commonly attached below the water surface. As said in the beginning, a clear distinction can be made regarding the responsibilities of EA and CMDB, but it seems plausible that stronger couplings between them could benefit both of them in terms of consistency and expressiveness.

7.6 Emerging Views

At later stages of this thesis, it became apparent that the suggested layered architecture description as well as the gathered information about information streams could be leveraged as a basis for other views. When grouping the collected concerns (see section 5.1), three possible views were identified.

7.6.1 Storage View

A storage view would display information about where certain information is buffered or stored. It would provide an overview of major databases and buffering components. Such a view is partially already available through the proposed meta model, as prolonged transmission intervals imply buffering. Still, it is not yet possibly to easily identify where data is stored for longer amounts of time. In the age of cloud-computing this is highly relevant, as cloud applications are not necessarily running inside an owned data center, but could be operated in other countries with differing governmental regulation regarding data privacy. Building on top of the already available data on information flow, such a view could be implemented as an extension of the proposed architecture.

7.6.2 Maintenance View

A maintenance view would hold information about untapped business potentials as well as information relevant to operative personnel. During the interviews with operative traffic managers, a desire was expressed to not only visualize actual communication, but also unused interfaces, unused configurations of used interfaces or systems' dependencies in correlation with configurations of information streams. While the configuration of specific systems would be captured on instance level, the configuration of flowing information is largely consistent across instances of the same type (which is the initial reasoning for modeling information flows within the enterprise architecture). Such information could comprise references to systems handling address management (e.g. IP-address allocation). Additionally, a maintenance view could also provide information about the type of power supply leveraged to run systems. As a maintenance view would be related to operative management, it could also provide information about the extensibility of systems. Many fog systems, for example, could be extended by means of additional software or patched firmware, as discussed exemplarily in section **??** (Smart Parking).

7.6.3 Security View

Security is a topic with varying importance subject to environmental factors [Federrath, 2014], e.g. the interest in information security recently increased following the Snowden affair [Heise, 2016]. Still, an ongoing long-term trend in growth of importance can be assumed, as the number of incidents declines [Eurostat, 2016]. As most of these issues are related to information sent and the proposed distinction between smart bricks, fog and cloud systems do pose implications to security issues, a security view would fit well into the proposed architectural viewpoint. Every raw data stream, information stream and information service could be extended as depicted in figure 7.4 to incorporate security-related information. The proposed attributes are related to strategic IT-security management, as enterprise architecture management itself is a part of strategic IT-management.

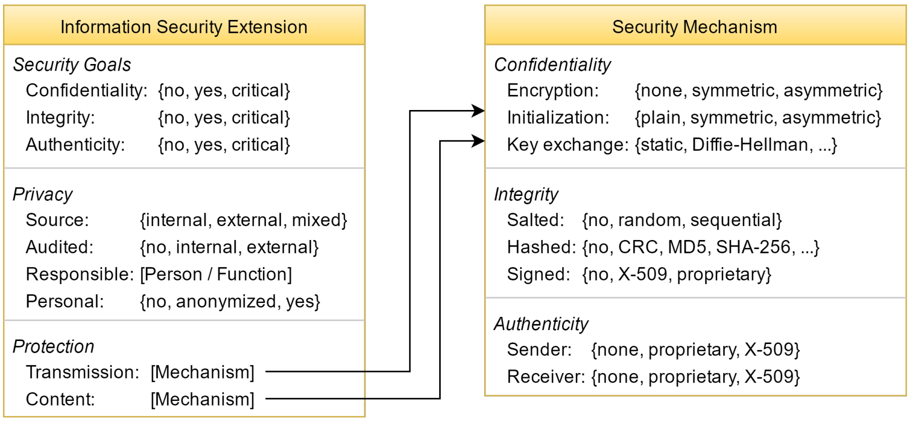

Figure 7.4: An privacy-extension applicable to all types of flowing information.
Source: *Own work*

Generally, the attribute selection was guided by the ISO 2700x series (specifically ISO 27002 [ISO, 2013]). In contrast to approaches like [Narman et al., 2008], the attributes are specifically tailored for flowing information and accordingly stress the aspect of authenticity, which is a part of integrity management. However, the aspect of availability is neglected, because this is related to information safety, not the regarded information security. Instead, the proposed categories are extended to reflect information privacy and strategic goals of managing privacy and security. The relevance of additional attributes, e.g. attributes related to possibilities of eavesdropping or reactive security, could not be tested for relevance

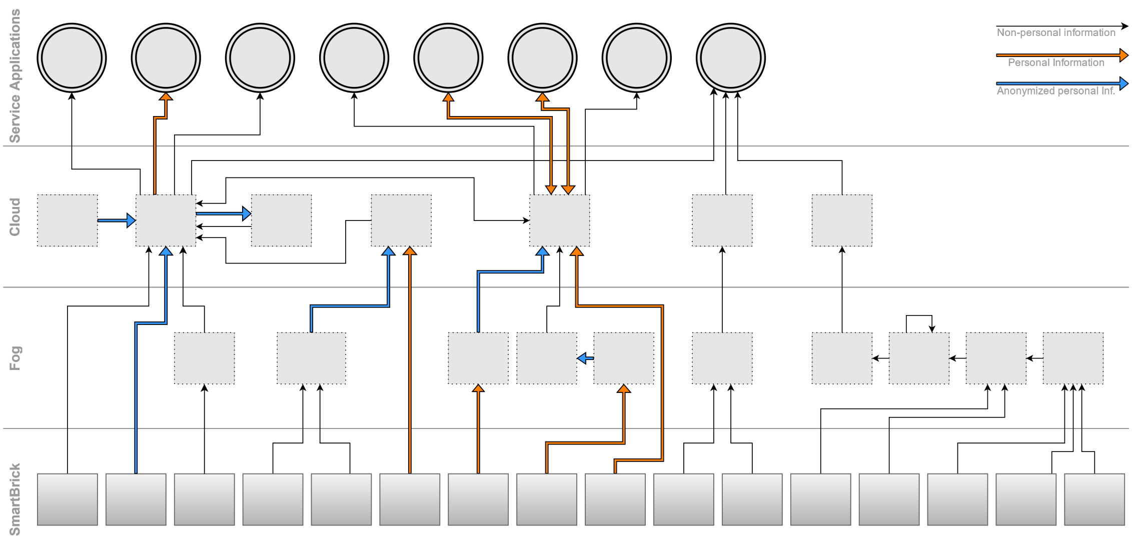

Figure 7.5: A data privacy-concerned visualization of the HPA's derived enterprise architecture model.
Source: *Own work*

in the investigated environments. The proposed security-extensions would need to be checked in operation and will not be incorporated into the EAM-tool employed at the HPA. To demonstrate the attribute's purpose, a visualization of privacy-related attributes was created in form of a mock-up seen in figure 7.5. The displayed part of the architecture shows how personalized information is

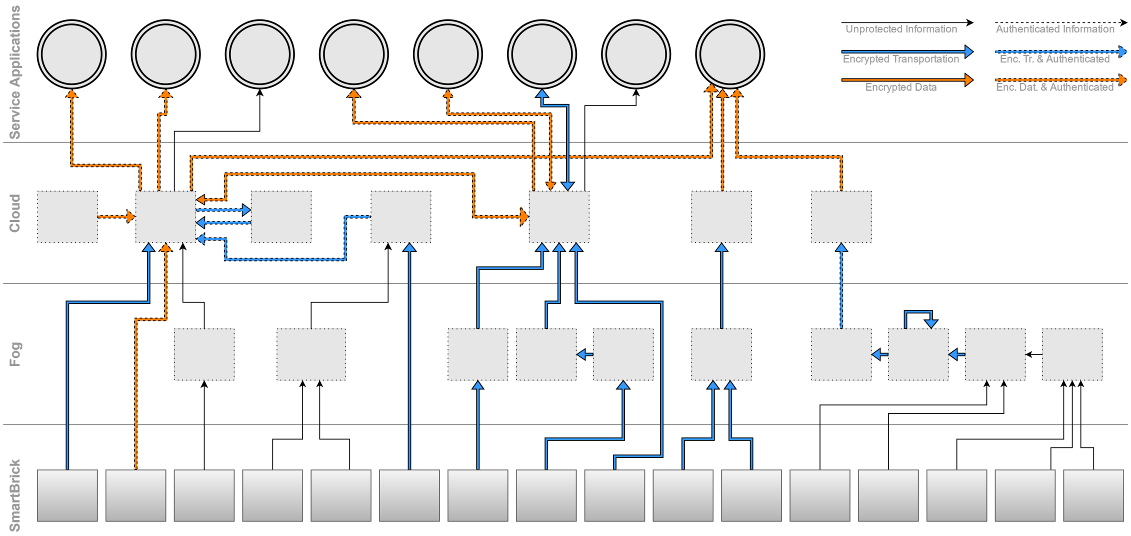

Figure 7.6: A data confidentiality- and integrity-concerned visualization of the HPA's derived enterprise architecture model.
Source: *Own work*

sent between systems and that several systems are able to anonymize information. Additionally, a second mock-up was created to visualize the current state of information security (confidentiality and authenticity) and can be seen in figure 7.6. Seeing the created model visualization, it becomes apparent that encrypted transmission is the preferred encryption below the cloud layer, while at cloud layer and above the data is encrypted directly. The latter is probably due to the usage of standardized transmission (mostly TCP/IP), and outlines that encrypted transmission is preferred (whenever applicable). It is worth noting that the data which figures 7.5 and 7.6 are based on is probably incomplete and possibly not entirely correct, as the necessary attributes will not be used by the HPA and the acquisition of required data was not actively supported.

A Appendix

A.1 Interview Questionnaire

1. Einleitung

   a) Vorstellung der Interviewer

   b) Erläuterung der Ziele der Masterarbeiten (kurz, evtl. durch UB,IS)

      - Darstellung eines Architekturmodells des (Business Ecosystems) SmartPort mit Grobmodellierung/Beschreibung der verschiedenen Ebenen und Komponenten aus unterschiedlichen Perspektiven sowie innovativer Visualisierung, mit dem Ziel
      - weitere darauf aufbauende Projekte zu informieren
      - Managementaufgaben (Strategie, Betrieb, Maintenance etc) zu unterstützen
      - Erfassung der Schritte des Projektes (der Intervention in das Business Ecosystem SmartPort)
      - Herausarbeitung von Besonderheiten bei Projekten/Architekturen mit SmartTechnologies
      - Validierung der entstandenen Dokumentation aus Projekten in bezug auf IT-Bebauungsplanung etc.
      - Unterlagen: Informationsflußgraphik (pro Projekt), Bebauungsplangraphik

   c) Angenommener Schwerpunkt des Interviews

      - Beschreibung einer entwickelten Solution bzw. des sie realisierenden Projektes aus Projektmanagement- sowie Fachbereichsperspektive

   d) Ausarbeitung des Interviews

      - Rückkopplung der skizzierten Antworten (textbasiert)

- Einarbeitung in Architekturbeschreibung, Visualisierung und Interventionsbeschreibung
- Validierung und evtl. Ergänzung/Überarbeitung der bestehenden HPA-Dokus (Informationsflußgraphik (pro Projekt), Bebauungsplangraphik)

2. Rolle des Projektes und des Interviewpartners

   a) Kernziele/aufgaben des Projektes

      - Was waren die (Kern-) Ziele des Projektes? Welche neuen Erkenntnisse kommen aus der Umsetzung? Wie fließen diese in die Konzeption darauf aufbauender Projekte oder Erweiterungen ein? Welche Erweiterungen sind bereits in Planung?

   b) Rolle des Interviewpartners (kurz)

      - Was genau war/en Ihre Aufgabe/n im Projekt?
      - Welche Positionen mit welchen Verantwortlichkeiten/Befugnissen haben Sie im Projekt bekleidet? Wie lange? Welche Verbindungen und Aktivitäten bestehen weiterhin?
      - Mit wem haben Sie im Projekt zusammengearbeitet (Personen / Organisationen)?

3. Architektur

   a) Solution und Services

      - Solution:
      - Bitte beschreiben Sie kurz die durch das Projekt entwickelte Solution, welchen Umfang, welche fachlichen Services bietet sie an? Welche Geschäftsmodelle sind angedacht (hier: nicht öffentliche Auswertung)?
      - Wer sind/wären die Adressaten/Nutzergruppen in welchen Nutzungsszenarien? Wie sind letztere entwickelt/designed worden?
      - Fachliche Services und Apps:

- Aus welchen fachliche Services/Teilservices und Apps besteht eine Solution? Nach welchen Gesichtspunkten kann/sollte man diese Services und Apps schneiden/formen? (Wie) Sollte man fachliche Services und Apps unterscheiden? Sind diese Services und Apps beschrieben?
- Wie sehen Funktionalität und fachlichen Schnittstellen hierzu aus? Existiert hierüber eine Dokumentation?
- Welche fachlichen Services und Apps werden intern, welche extern bereitgestellt? Beziehen sich dies auf Entwicklung und Betrieb gleichermaßen?
- Status Entwicklung:
- In welchem Umfang wurde die Solution umgesetzt, was ist ausschnitthaft, prototypisch?
- Welche Pilotnutzer gibt es? Wie wurden diese gewonnen? Wie wurde getestet?
- Welche Ausrollszenarien sind angedacht? Welche Veränderung bedeutet dies bei den Nutzergruppen/-organisationen?
- Erweiterbarkeit:
- Welche zusätzlichen, neuen Services und Apps wären in diesem Kontext sinnvoll?
- Zu welchen erweiterten oder neuen Solutions ließen sich die im Projekt oder auch aus anderen Projekten realisierten neuen bzw. auch extern verfügbaren Services und Apps konfigurieren? Welche Informationen wären hierbei hilfreich?
- Welche nächsten Schritte sind zu erwarten?

b) Prozesse

- Zusammenhang Prozesse – Services
- Welche Prozesse verändern sich durch die in der Solution eingeführten fachlichen Services?

- Welche davon sind extern, welche intern?
- Wurde dies bei der Entwicklung der Services mit betrachtet? Oder zu welchem Zeitpunkt wurde dies relevant?
- Läßt sich die Größe der Organisationsveränderung (Changemanagement) abschätzen? Wodurch? Wie werden interne und externe Veränderungen angeleitet?
- Wie werden Einigungen bzgl. veränderter/neuer Verantwortlichkeiten verschiedener Partner erzielt?
- Welche Informationssysteme (intern, extern) waren aus diesem Grund zu verändern?
- Wie aufwändig/lokal waren diese Veränderungen?

c) Informationsströme und „Informationsservices"

- Was sind Rohdatenströme (wie werden diese unterschieden, nach Sensortypen, nach physischen Objekttpyen (Brücke, Parkplatz, Kaimauer. . .))? Wie werden sie zu welchen Informationsströmen gebündelt (z.B. Parkplatzbelegungsgrad, Slotbezogene Bündelung)? In welcher Frequenz werden sie gesendet?
- Welche Informationsströme wurden realisiert und fließen in welche Services ein? Welche werden nach außen gegeben, welche von außen bezogen? (Wer sind die Partner)? Würden Sie in diesem Zusammenhang von Informationsservices sprechen und diese von fachlichen Services unterscheiden?
- Nach welchen Gesichtspunkten wurden sie festgelegt/entwickelt?
- Welche Rolle spielen/spielten Datenschutzgesichtspunkte? Wo stellt sich die Frage: „Wem gehören die Daten/Informationen?" Welche Services und Apps werden individuell zugeschnitten? (hier evtl. nicht öffentliche Auswertung)?
- Wo befinden sich sicherheitskritische Informationsströme und welche Maßnahmen sind zu ihrem Schutz ergriffen (hier evtl.

nicht öffentliche Auswertung)?

- Werden Rohdatenströme und Informationsströme nach Qualität unterschieden? Und was bedeutet Qualität?

d) Informationssysteme

- Verarbeitung der Informationsströme
- In welchen Informationssystemen werden die Rohdatenströme empfangen, und in Informationsströme gebündelt?
- Welche sind neu entwickelt?
- Welche sind erweitert worden?
- Welche Komponenten zur Rohdaten-/Informationsstromverarbeitung sind eingeführt worden? Lassen sich einheitlichen Komponenten projektübergreifend einführen, bzw. Architekturmuster entwickeln?
- Wie unterscheiden sich diese Schnittstellen von herkömmlichen Schnittstellen zwischen Informationssystemen (z.B. zu Ordersystemen)?
- Ist eine extra Data-Analytics bzw. Informationstromanalyseschicht einzuführen – oder pro System (s.o.)?
- Wo werden Daten/Informationen jeweils in welcher Form wie lange zur Verarbeitung (und aus anderen Gründen) gespeichert?
- Funktionalität der Informationssysteme
- Welche Funktionalität/Dienste stellen die Informationssysteme bereits?
- Welche externen Informationssysteme sind angebunden (bei welchen Partnern), was muss von ihnen bekannt sein bzw. wie sollten diese dokumentiert werden, reichen hier standardisierte Schnittstellen, welche?
- Stellt die Integration der herkömmlichen prozessorientierten Informationssysteme (z.B. Ordersysteme) mit neuen Rohdaten/in-

formationsstromverarbeitenden Informationssystemen eine besondere Herausforderung dar? Welche?

e) SmartBricks (Begriff wird erläutert: physische Objekte, ausgestattet mit Sensoren und Tags)

- Welche Brickarten (phyische Objektarten) wurden mit Sensoren und Tags ausgestattet? Wieviele wurden jeweils realisiert?
- Wurden Bricks der gleichen Brickart möglichst mit gleichen Sensoren/Tags (Anzahl/Art, Herstellertyp etc) zu SmartBrickarten ausgestattet? Sollte eine „Standardausstattung" und Abweichungsmodi sowie projektbezogene Ausbaustufen von Brick- zu Smartbrickarten eingeführt werden?
- Welche Sensorarten wurden eingebaut?
- Welche Tags?
- Wie würden Sie SmartBricks kategorisieren? Nach Art (z.B. Brücke ...), nach Kombination Brick und Sensor/Tag, nach Nutzung (z.B. Verkehrsdichtemessung)?
- Welche Fragen wollen Sie an die realisierte SmartBrick-Landschaft aus welchen Perspektiven stellen?

4. Projekt und Verbesserungsvorschläge

a) Projektverlauf

- Welche Aufgaben (Design, Entwicklung, Bewerbung, Ausrollen/-Transformation, Auswahl externer Partner, Beratung, Betrieb...) wurden projektintern, HPA-intern, in Kooperation mit externen Partnern durchgeführt?
- Welche Veränderungen von Geschäftsprozessen/der Organisation wurde in welchen Fachbereichen und bei welchen externen Partnern im Rahmen des Projektes geplant und durchgeführt? Wie aufwändig war dies?

b) Verbesserungsvorschläge

- Wo würden Sie im Nachhinein Verbesserungen im Projekt vorschlagen? Wo gab es Abweichungen vom Plan oder Probleme in der Kommunikation . . . ?
- Welche Rolle könnte hierbei eine für alle Beteiligten nutzbare SmartPort Architektur als Informations- und Kommunikationsplattform spielen?

5. Zukünftiges Architekturmanagement und Governance

   a) SmartPort Architekturaufbau und Management

   - Aus Projektperspektive
   - Welche Informationen über Dienste, Prozesse Informationsströme, Informationssysteme benötigen Sie als Projektmanager beim Start/zur Planung des Projektes, in seinem Verlauf?
   - Welche Informationen sollten (externe) Partner erhalten können (z.B. auch zum Changemanagement)? Welche Partner/externe Verantwortliche?
   - Welche Informationen benötigen Sie dabei von weiteren parallel laufenden oder in Planung befindlichen Projekten?
   - Welche Informationen sollten aus dem Projekt in die SmartPort Architektur fließen?
   - Aus Fachbereichsperspektive
   - Welche Informationen/Anfragen über die dokumentierten Architekturelemente und ihre Beziehungen möchten Sie aus Fachbereichsperspektive erhalten können?
   - für strategische Planung
   - für den Betrieb (Verkehr, Infrastruktur etc) . . .
   - In welchen weiteren Aufgaben/Verantwortlichkeiten/Concerns/Bedarfen wären ihnen diese Informationen von Nutzen?
   - Haben Sie Vorstellungen für ein entsprechendes unterstützendes Tool?

b) Governance der Projekte

- Wie gestaltete sich die Zusammenarbeit mit anderen Projekten? Wer war (inhaltlich) federführend? Wer hat final entschieden?
- In Bezug auf die Entwicklung der Software und der Infrastruktur?
- In Bezug auf die Entwicklung der Services?
- Welche Gremien und Entscheidungsstrukturen gibt/gab es?
- Welche Gremien wurden eingeführt und in welchen Gremien und in welcher Rolle war der Projektleiter im Projektverlauf involviert?
- Wer hat über die Kommunikations- und Entscheidungsstrukturen in den Phasen entschieden und diese kontrolliert?
- Wer hat Aufgaben der Kommunikation mit Behörden, weiteren externen Entwicklungspartnern und zukünftigen Nutzern (Unternehmen, Stadt, Bürgern, Bürgerinitiativen, Öffentlichkeit. . .) übernommen?

c) Governance für Ausbau und Betrieb

- Welche Verantwortlichkeiten für den Betrieb neuer Solutions sind einzuführen?
- Mit welchen Managementaufgaben?
- Mit welchen Monitoringaufgaben?
- Schnittstelle zu Projekten: Nach Projektabschluss werden welche Aufgaben von wem übernommen, um die entwickelten Dienste/-Solutions auszurollen, in den Betrieb zu überführen?
- Wer sollte für den Ausbau und die Kombination der Solutions verantwortlich sein?
- Welche Anforderungen und Verantwortlichkeiten haben die Fachbereiche?

A.2 Conducted Interviews

Interviewee:	Name Anonymized
	IT-strategist at Hamburg Port Authority
Related Project:	Enterprise Architecture
Interviewee:	Name Anonymized
	IT-strategist at Hamburg Port Authority
Related Project:	iteraplan, an EAM-Tool
Interviewee:	Name Anonymized
	Project manager at Hamburg Port Authority
Related Project:	Smart Area Parking
Interviewee:	Name Anonymized
	Project manager at Hamburg Port Authority
Related Project:	EVE, SPL
Interviewee:	Name Anonymized
	Operative traffic manager at Hamburg Port Authority
Related Project:	EVE
Interviewee:	Name Anonymized
	Project manager at Hamburg Port Authority
Related Project:	SPL
Interviewee:	Name Anonymized
	Traffic analyst / manager at Hamburg Port Authority
Related Project:	EVE (traffic detection)
Interviewee:	Name Anonymized

	CDO at Hamburg Port Authority
Related Project:	Enterprise Architecture
Interviewee:	Name Anonymized
	Project manager at Hamburg Port Authority
Related Project:	smartTAG
Interviewee:	Name Anonymized
	External supplier
Related Project:	SmartSwitch
Interviewee:	Name Anonymized
	CDO at Hamburg Port Authority
Related Project:	Enterprise Architecture
Interviewee:	Name Anonymized
	Project manager at Hamburg Port Authority
Related Project:	Smart Road
Interviewee:	Name Anonymized
	Name Anonymized
	Name Anonymized
	Project manager / operative traffic managers at HPA
Related Project:	Traffic-related enterprise architecture
Interviewee:	Name Anonymized
	Software architects / developer at Name Anonymized
Related Project:	PrePORT Parking
Interviewee:	Name Anonymized
	IT-strategist at Hamburg Port Authority

Related Project: Enterprise Architecture

Interviewee: Name Anonymized
Senior consultant (EAM) at Name Anonymized
Related Project: Enterprise Architecture Management

A.3 Interview Results

- The interview results are classified and not disclosed in this version of the thesis -

A.4 Chapter 6: Applying EA-extensions to smartPORT projects and Evaluation

- The project analysis is classified and not disclosed in this version of the thesis -

References

[Aier et al., 2008] Aier, I. S., Riege, D.-W.-I. C., and Winter, R. (2008). Unternehmensarchitektur – literaturüberblick und stand der praxis. *Wirtschaftsinformatik*, 50(4):292–304.

[APCOA, 2013] APCOA (2013). Survey on finding parking lots in cities (German).
http://www.apcoa.de/nachrichten/artikel/apcoa-parking-studie-die-parkplatz-suche-verschlingt-viel-zeit-und-geld.html, accessed 2016-05-20.

[Armbrust et al., 2010] Armbrust, M., Fox, A., Griffith, R., Joseph, A. D., Katz, R., Konwinski, A., Lee, G., Patterson, D., Rabkin, A., Stoica, I., et al. (2010). A view of cloud computing. *Communications of the ACM*, 53(4):50–58.

[Atzori et al., 2010] Atzori, L., Iera, A., and Morabito, G. (2010). The internet of things: A survey. *Computer networks*, 54(15):2787–2805.

[Avison et al., 2001] Avison, D., Baskerville, R., and Myers, M. (2001). Controlling action research projects. *Information technology & people*, 14(1):28–45.

[Bassi et al., 2013] Bassi, A., Bauer, M., Fiedler, M., Kramp, T., Van Kranenburg, R., Lange, S., and Meissner, S. (2013). Enabling things to talk. *Designing IoT Solutions With the IoT Architectural Reference Model*, pages 163–211.

[BASt, 2012] BASt (2012). Technische lieferbedingungen für streckenstationen, ausgabe 2012. Technische Regelwerke für das Straßenwesen in Deutschland TLS2012, Bundesanstalt für Straßenwesen, Bergisch Gladbach, Germany. German.
http://www.bast.de/DE/Verkehrstechnik/Fachthemen/v5-tls/tls-streckenstationen.html, accessed 2016-05-20.

[bitkom, 2016] bitkom (2016). Implementation strategy industrie 4.0. bitkom, Berlin, Germany. https://www.bitkom.org/Publikationen/2016/Sonstiges/Implementation-Strategy-Industrie-40/2016-01-Implementation-Strategy-Industrie40.pdf, accessed 2016-05-17.

[Blaxter, 2010] Blaxter, L. (2010). *How to research*. McGraw-Hill Education (UK).

[Borgia, 2014] Borgia, E. (2014). The internet of things vision: Key features, applications and open issues. *Computer Communications*, 54:1–31.

[Breivold et al., 2015] Breivold, H. P., Sandstr, K., et al. (2015). Internet of things for industrial automation–challenges and technical solutions. In *2015 IEEE International Conference on Data Science and Data Intensive Systems*, pages 532–539. IEEE.

[Brettle and Gambling, 2003] Brettle, A. and Gambling, T. (2003). Needle in a haystack? effective literature searching for research. *Radiography*, 9(3):229–236.

[Britten, 1995] Britten, N. (1995). Qualitative research: qualitative interviews in medical research. *Bmj*, 311(6999):251–253.

[Brocke et al., 2009] Brocke, J. V., Simons, A., Niehaves, B., Riemer, K., Plattfaut, R., Cleven, A., et al. (2009). Reconstructing the giant: On the importance of rigour in documenting the literature search process. In *ECIS*, volume 9, pages 2206–2217.

[Chen et al., 2012] Chen, H., Chiang, R. H., and Storey, V. C. (2012). Business intelligence and analytics: From big data to big impact. *MIS quarterly*, 36(4):1165–1188.

[Christopher, 2000] Christopher, M. (2000). The agile supply chain: competing in volatile markets. *Industrial marketing management*, 29(1):37–44.

[Ciborra, 1992] Ciborra, C. U. (1992). From thinking to tinkering: The grassroots of strategic information systems. *The information society*, 8(4):297–309.

[Cisco, 2016a] Cisco (2016a). Cisco completes jasper acquisition. Public notice on Jasper acquisition.
http://www.cisco.com/c/en/us/about/corporate-strategy-office/acquisitions/jasper.html, accessed 2016-05-12.

[Cisco, 2016b] Cisco (2016b). Cisco visual networking index: Global mobile data traffic forecast update, 2015–2020. http://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/mobile-white-paper-c11-520862.pdf, accessed 2016-05-24.

[Cisco, 2015] Cisco, I. (2015). Enabling the internet of everything: Cisco's iot architecture. Cisco Live Milan breakout session.

http://www.slideshare.net/Cisco/enabling-the-internet-of-everything-ciscos-iot-architecture, accessed 2016-05-02.

[Cole et al., 2005] Cole, R., Purao, S., Rossi, M., and Sein, M. (2005). Being proactive: where action research meets design research. *ICIS 2005 Proceedings*, page 27.

[Cooper, 1988] Cooper, H. M. (1988). The structure of knowledge synthesis. *Knowledge in society*, 1(1):104–126.

[Correia et al., 2009] Correia, A. et al. (2009). Integrating it service management within the enterprise architecture. In *Software Engineering Advances, 2009. ICSEA'09. Fourth International Conference on*, pages 553–558. IEEE.

[Eurostat, 2016] Eurostat (2016). Jeder vierte internetnutzer in der eu hatte probleme mit internetsicherheit im jahr 2015. Press release from Eurostat, the statistical office of the European Communities.
http://ec.europa.eu/eurostat/documents/2995521/7151123/4-08022016-AP-DE.pdf/d73a1134-763c-483d-b9ba-aeef5fb18e50, accessed 2016-05-13.

[Farwick et al., 2012] Farwick, M., Pasquazzo, W., Breu, R., Schweda, C. M., Voges, K., and Hanschke, I. (2012). A meta-model for automated enterprise architecture model maintenance. In *Enterprise Distributed Object Computing Conference (EDOC), 2012 IEEE 16th International*, pages 1–10. IEEE.

[Federrath, 2014] Federrath, H. (2014). Sicherheitsmanagement. lecture notes. Lecture notes on information security management.

[Felter et al., 2015] Felter, W., Ferreira, A., Rajamony, R., and Rubio, J. (2015). An updated performance comparison of virtual machines and linux containers. In *Performance Analysis of Systems and Software (ISPASS), 2015 IEEE International Symposium On*, pages 171–172. IEEE.

[Fuglsang and Sørensen, 2011] Fuglsang, L. and Sørensen, F. (2011). The balance between bricolage and innovation: Management dilemmas in sustainable public innovation. *The service industries journal*, 31(4):581–595.

[Gantz and Reinsel, 2011] Gantz, J. and Reinsel, D. (2011). Extracting value from chaos. *IDC iview*, 1142:1–12.

[Gartner, 2015] Gartner (2015). Gartner's 2015 hype cycle for emerging technologies. Gartner, Inc.

[Haller, 2010] Haller, S. (2010). The things in the internet of things. *Poster at the IoT 2010, Tokyo, Japan, November*, 5:26.

[Haller and Magerkurth, 2011] Haller, S. and Magerkurth, C. (2011). The real-time enterprise: Iot-enabled business processes. In *IETF IAB Workshop on Interconnecting Smart Objects with the Internet*. Citeseer.

[Hanschke, 2009] Hanschke, I. (2009). *Strategic IT Management: A Toolkit for Enterprise Architecture Management*. Springer Science & Business Media.

[Haren, 2011] Haren, V. (2011). *TOGAF Version 9.1*. Van Haren Publishing.

[Heise, 2016] Heise, M. G. (2016). Amazon-technikchef: Snowden-enthüllungen treiben verschlüsselung. Online media from Heise Medien GmbH (dpa, anw). http://heise.de/-3175353, accessed 2016-05-13.

[HPA, 2015a] HPA (2015a). Energy cooperation at the port of hamburg. Informational leaflet, published by the Hamburg Port Authority, AöR (German). http://www.hamburg-port-authority.de/de/presse/broschueren-und-publikationen/Documents/broschuere_smartportenergy_web.pdf, accessed 2016-05-18.

[HPA, 2015b] HPA (2015b). Hpa goes smartport. Informational leaflet, published by the Hamburg Port Authority, AöR (German). http://www.hamburg-port-authority.de/de/presse/broschueren-und-publikationen/Documents/150422_tl_messe_lowres.pdf, accessed 2016-05-20.

[HPA, 2015c] HPA (2015c). Intelligent networks and smart sensors for a more efficient port management. Informational leaflet, published by the Hamburg Port Authority, AöR (German). http://www.hamburg-port-authority.de/de/presse/broschueren-und-publikationen/Documents/140401_HPA_Broschuere_spl_web.pdf, accessed 2016-05-18.

[HPA, 2016] HPA (2016). Gesamtumschlag im hamburger hafen von 1990 bis 2015 (in millionen tonnen). Found in Statista, data provided by the Hamburg Port Authority, AöR.

http://de.statista.com/statistik/daten/studie/29803/umfrage/gesamtumschlag-im-hamburger-hafen/, accessed 2016-05-20.

[Intel, 2015] Intel (2015). 6th generation intel® core™processor family datasheet, vol. 1. CPU-Datasheet. http://www.intel.de/content/dam/www/public/us/en/documents/datasheets/desktop-6th-gen-core-family-datasheet-vol-1.pdf, accessed 2016-05-12.

[ISO, 2011] ISO (2011). Systems and software engineering - architecture description. ISO 42010:2011, International Organization for Standardization, Geneva, Switzerland.

[ISO, 2013] ISO (2013). Information security management code of practice. ISO 27002:2013, International Organization for Standardization, Geneva, Switzerland.

[Kniberg and Ivarsson, 2012] Kniberg, H. and Ivarsson, A. (2012). Scaling agile @ spotify with tribes, squads, chapters & guilds. Spotify on the topic of agile scaling. https://creativeheldstab.com/wp-content/uploads/2014/09/scaling-agile-spotify-11.pdf, accessed 2016-05-12.

[Kornmeier, 2007] Kornmeier, M. (2007). *Wissenschaftstheorie und wissenschaftliches Arbeiten: eine Einführung für Wirtschaftswissenschaftler*. Springer-Verlag.

[Kumar and Sowjanya, 2015] Kumar, P. S. and Sowjanya, S. (2015). Developing an enterprise environment by using wireless sensor network system architecture. *International Journal of Computer Engineering in Research Trends*, 2(10):902–908.

[Matthes et al., 2008] Matthes, F., Buckl, S., Leitel, J., and Schweda, C. M. (2008). *Enterprise architecture management tool survey 2008*. Techn. Univ. München.

[Matthes et al., 2016] Matthes, F., Hauder, M., Schneider, A. W., and Khosproshahi, P. A. (2016). Enterprise architecture management pattern catalog v2. https://wwwmatthes.in.tum.de/pages/3b4t6l34g936/EAM-Pattern-Catalog-Wiki, accessed 2016-04-29.

[Mell and Grance, 2011] Mell, P. and Grance, T. (2011). The nist definition of cloud computing.

[Meyer et al., 2015] Meyer, S., Ruppen, A., and Hilty, L. (2015). The things of the internet of things in bpmn. In Persson, A. and Stirna, J., editors, *Advanced Information Systems Engineering Workshops*, pages 285–297. Springer International Publishing. Proceedings of CAiSE 2015 International Workshops, Stockholm, Sweden, June 8-9, 2015.

[Namiot and Sneps-Sneppe, 2014] Namiot, D. and Sneps-Sneppe, M. (2014). On micro-services architecture. *International Journal of Open Information Technologies*, 2(9).

[Narman et al., 2008] Narman, P., Schonherr, M., Johnson, P., Ekstedt, M., and Chenine, M. (2008). Using enterprise architecture models for system quality analysis. In *Enterprise Distributed Object Computing Conference, 2008. EDOC'08. 12th International IEEE*, pages 14–23. IEEE.

[SAP, 2016] SAP (2016). Sap hana platform: Data integration, data access and data quality. Goal description of the SAP HANA platform. https://hana.sap.com/capabilities/eim.html, accessed 2016-05-12.

[SAP and IBM, 2016] SAP and IBM (2016). Ibm and sap announce plans for major investments to drive clients' digital transformations. IBM press release. http://ibm.biz/Bd4qqL, found through http://heise.de/-3163547, both accessed 2016-05-13.

[Schirmer et al., 2016] Schirmer, I., Drews, P., Saxe, S., Baldauf, U., and Tesse, J. (2016). Extending enterprise architectures for adopting the internet of things – lessons learned from the smartport projects in hamburg. 19th International Conference on Business Information Systems, Leipzig, Germany.

[Sein et al., 2011] Sein, M., Henfridsson, O., Purao, S., Rossi, M., and Lindgren, R. (2011). Action design research.

[ServiceNow, 2016] ServiceNow (2016). Populating the cmdb. CMDB-population through automated discovery. http://wiki.servicenow.com/index.php?title=Populating%5fthe%5fCMDB, accessed 2016-05-12.

[Soni et al., 1995] Soni, D., Nord, R. L., and Hofmeister, C. (1995). Software architecture in industrial applications. In *Software Engineering, 1995. ICSE 1995. 17th International Conference on*, pages 196–196. IEEE.

[Steininger et al., 1996] Steininger, K., Vogl, C., and Zettl, R. (1996). Car-sharing organizations: The size of the market segment and revealed change in mobility behavior. *Transport Policy*, 3(4):177–185.

[Stojmenovic and Wen, 2014] Stojmenovic, I. and Wen, S. (2014). The fog computing paradigm: Scenarios and security issues. In *Computer Science and Information Systems (FedCSIS), 2014 Federated Conference on*, pages 1–8. IEEE.

[Van Haren, 2011] Van Haren, P. (2011). *Itil®2011 Edition – A Pocket Guide*. Van Haren Publishing.

[Vilaplana et al., 2013] Vilaplana, J., Solsona, F., Abella, F., Filgueira, R., and Rius, J. (2013). The cloud paradigm applied to e-health. *BMC medical informatics and decision making*, 13(1):35.

[Windelen, 2015] Windelen, M. (2015). Integration von internet of things systemen in das enterprise architecture management.

[Winter et al., 2010] Winter, K., Buckl, S., Matthes, F., and Schweda, C. M. (2010). Investigating the state-of-the-art in enterprise architecture management methods in literature and practice. *MCIS*, 90.

[Winter and Fischer, 2006] Winter, R. and Fischer, R. (2006). Essential layers, artifacts, and dependencies of enterprise architecture. In *Enterprise Distributed Object Computing Conference Workshops, 2006. EDOCW'06. 10th IEEE International*, pages 30–30. IEEE.

[Zachman, 2016] Zachman, J. (2016). The zachman framework ™. Apparent successor of the 1987 Zachman Framework. https://www.zachman.com/about-the-zachman-framework, accessed 2016-04-26.

[Zachman, 1987] Zachman, J. A. (1987). A framework for information systems architecture. *IBM systems journal*, 26(3):276–292.

[Zimmermann, 1980] Zimmermann, H. (1980). Osi reference model–the iso model of architecture for open systems interconnection. *Communications, IEEE Transactions on*, 28(4):425–432.

Bibliothekseinstellung

Der Autor ist mit der Einstellung der Arbeit in die Bibliothek einverstanden.

Hamburg, den __________ Unterschrift: __________________________

Eidesstattliche Erklärung

Ich versichere, dass ich die vorstehende Arbeit selbstständig und ohne fremde Hilfe angefertigt und mich anderer als der im beigefügten Verzeichnis angegebenen Hilfsmittel nicht bedient habe. Alle Stellen, die wörtlich oder sinngemäß aus Veröffentlichungen entnommen wurden, sind als solche kenntlich gemacht.

Hamburg, den __________ Unterschrift: _________________________