Hochschule für Angewandte Wissenschaften Hamburg
Hamburg University of Applied Sciences

Bachelor Thesis

Christian Gläser

Developing an ICT blueprint for a mid-size build-to-order corporation

Fakultät Technik und Informatik
Studiendepartment Informatik

Faculty of Engineering and Computer Science
Department of Computer Science

Christian Gläser

Developing an ICT blueprint for a mid-size build-to-order corporation

Bachelor Thesis eingereicht im Rahmen der Bachelorprüfung

im Studiengang Bachelor of Science Wirtschaftsinformatik
am Department Informatik
der Fakultät Technik und Informatik
der Hochschule für Angewandte Wissenschaften Hamburg

Betreuender Prüfer: Prof. Dr. Ulrike Steffens
Zweitgutachter: Prof. Dr. Rüdiger Weißbach

Eingereicht am: 18. November 2015

Christian Gläser

Title of the paper

Developing an ICT blueprint for a mid-size build-to-order corporation

Keywords

IT-Management, Enterprise Architecture Management, EAM, EA, Service-Oriented Architecture, SOA, Integration, EAI, ICT

Abstract

Structuring and documenting the IT landscape is increasingly important for IT management. By giving an example for a mid-size built-to-order corporation, a formal and documented ICT blueprint is created based on an informal reference model. The implementation uses Enterprise Architecture Methods and considers integration aspects. The thesis also illustrates the dependencies between Business-, Enterprise- and IT-Architecture and evaluates the current state of research regarding mass customization, Enterprise Architecture Management and integration frameworks with a focus on service-oriented architecture (SOA).

Christian Gläser

Thema der Arbeit

Developing an ICT blueprint for a mid-size build-to-order corporation

Stichworte

IT-Management, Enterprise Architecture Management, EAM, EA, serviceorientierte Architektur, SOA, Integration, EAI, ICT

Kurzzusammenfassung

Strukturierung und Dokumentation der IT-Infrastruktur ist für IT-Management zunehmend wichtig. Exemplarisch wird für einen mittelständischen Auftragseinzelfertiger ein informelles Referenzmodell softwaregestützt in ein formales und dokumentiertes Modell der IT-Landschaft überführt. Für die Implementierung sind Enterprise Architecture sowie IT-Integrationsaspekte (serviceorientierte Architektur) wichtig. Die Thesis verdeutlicht die Zusammenhänge zwischen Geschäfts-, Enterprise- und IT-Architektur.

Contents

List of Figures

1 Introduction

The term Information Technology (IT) was coined in 1958 [LW58]. Since then disruptive changes related to IT have occured at least once a decade: In the 1960s the Microprocessors was invented, followed by Ethernet in the 1970s. The 1980s followed with PCs (IBM 5150: 1981 [Old13] [Int03]) and Microsoft Windows (1985). In the last decade of the 20th century the internet - invented in 1989 - experienced rapid growth. Today, these pervasive technologies weave so much into everyday life that they become indistinguishable from it (Mark Weiser) - this trend was accelerated by another disruptive change - the invention of the smartphone in the 2000s.

The smartphone is also a good example for the combination of two formerly separate technologies: IT and Communication Technology (CT) have converged to form **I**nformation and **C**ommunication **T**echnology (ICT). Today the terms IT and ICT are used interchangeably [Hua12] and the pure communication technology has become outdated as a result of technological progress.

The progress outlined before leads to a necessity to update outdated elements of the business infrastructure which is nothing new. In the past the IT in corporations was updated on a regular basis, too.

However, these update lead to a problem: In some sectors, such as the banking sector, IT has been used for more than five decades. [Sch12], others (such as engineering or production of goods) have used IT for a shorter period, e.g. after the invention of the PC. Even then, some IT components have been in use in these companies for more than thirty years nonstop.

New technology with improved capabilities has been added to the IT landscape which could execute the tasks of old components, so these old components could retire. But the transition is not yet complete. For this reason multiple technology generations fulfilling the same purpose are often being used simultaneously. The problem of legacy technology is well-known and documented in sectors where the use of IT started early (e.g. for the banking sector [Sch12]). These examples for prolonged technology use are therefore well known in scientific context. Furthermore, the extension and migration of IT infrastructure is often only planned on an operational and not on a strategic level. This has led to most aged IT ecosystems being in a complex, fragmented and often in an undocumented or poorly documented state.[Lin05]. As the infrastructure ages, the problem of multiple technology generations, poor documentation and planning affects corporations more severely. For this reason the architecture of information systems, the convergence of technologies and the ICT infrastructure has been discussed more intensively in recent years. Architecture in this context is 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.[ISO11]

A fragmented, undocumented enterprise is not ideal and therefore Enterprise Architecture Management (EAM) proposes a solution for realigning the Business- and the IT-Architecture, as illustrated in figure 1.1.
EAM methods include creating or improving the existing IT documentation and restructuring the architecture of IT ecosystems. In the theoretical section of this thesis, an introduction into EAM will be given. On a technical level, the techniques and challenges of integration of the various applications used in an enterprise context are explained by giving an introduction into Integration Frameworks.
The theory outlined is practically implemented during development of an ICT blueprint. This will be the practical result of the thesis. This result will be used in the context of the HAWAI (**HAW** Laboratory for **A**pplication **I**ntegration) project. The HAWAI project is a project of the faculty "Engineering and Computer Science" of the Hamburg University of Applied Sciences (HAW).

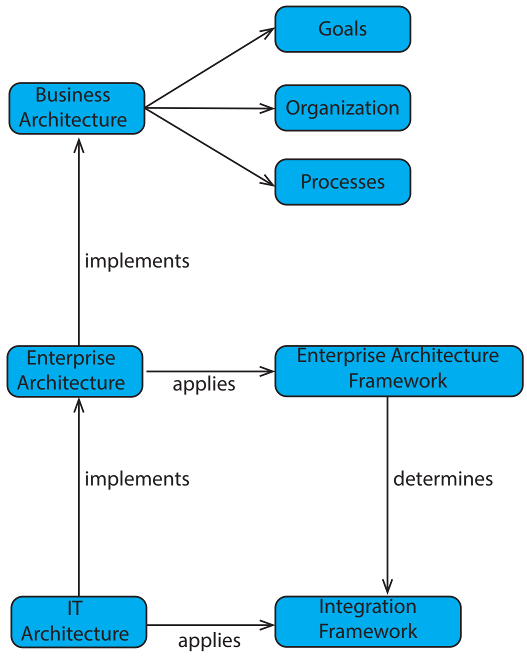


Figure 1.1: The Relation between Business Architecture, Enterprise Architecture and IT Architecture

The HAWAI project has been set up to provide realistic examples for students during teaching and practice sessions and to allow research in the areas of software architecture, application integration and EAM. One of the main project goals is the construction of several realistic application landscapes and their interfaces [HAW] for a system integrations lab within the department of computer science.[HRSS15]

Once completed, the lab will provide students with realistic application landscapes. A realistic structure in this context is an application landscape that contains all the design pitfalls that exist in realistic ICT infrastructures such as pragmatic IT integrations (see 3.2.1). However, to mitigate the risks associated with such a structure, one of the goals of HAWAI is to create an architecture that can be reset to a safe (initial) state. This limits the risks associated with experiments substantially: In worst case everything will be reset. This system property also is a major difference to a typical corporate environment. HAWAICycle - the corporation behind the ICT blueprint - as such is targeted to be a blueprint for such an IT environment and therefore the results found in this thesis will be used during the implementation of the HAWAICycle IT environment. The results are not only relevant for the HAWAI project as HAWAI partners with MARS (Multi-Agent Research & Simulation Group) to simulate corporate environments and their customers.
Of course, all the problems described before concerning aged infrastructure are currently non-existent in the project context, since the fictional mid-size build-to-order company HAWAICycle GmbH currently is neither aged nor has any infrastructure that could cause complications. However, to give students an opportunity to develop solutions to these problems, it is important to create an artificial environment which is affected by all those problems in an infrastructure that (unlike a real infrastructure) can be reset to an initial state at any given time to prevent lasting damage. For this reason the fictional enterprise is described and methods from EAM are used to create an infrastructure, which is a special case for the use of EAM methods. The common approach to EAM would be to identify existing systems by observation and therefore this approach to EAM is novel. However, EAM can be applied in this case as well - the reason for it is explained in this thesis. Because systems can't be identified by observation in an environment with no pre-existent infrastructure, an informal blueprint for built-to-order-companies is used as input for the a formal blueprint. The development of the formal ICT blueprint is divided into three stages.
In the first stage all systems of the ICT blueprint are identified and described. The description contains tasks, interfaces to related systems and data which is to be shared with each related system. This systems and their descriptions will be entered into an EAM Tool and will be documented in the appendix.
In the second stage the informal model will be cleaned and adapted to HAWAICycle.
In the last stage the model will be implemented using Iteraplan, an EA modeling tool. Additionally, a selection of business processes will be created to complement the ICT blueprint.
At last, a conclusion will be drawn and recommendations for further studies will be given.

2 HAWAICycle GmbH

HAWAICycle GmbH[1] is a virtual corporation constituting the basis of this thesis and of special value for the concrete example as shown below. The Business Architecture that is designed reflects the organization, its goals and the processes. As shown in figure 2.1 and explained in the theory chapter in section 3.1 the ICT blueprint reflects the Business Architecture. This description of the organization and its goals is one of the cornerstones of this architecture.

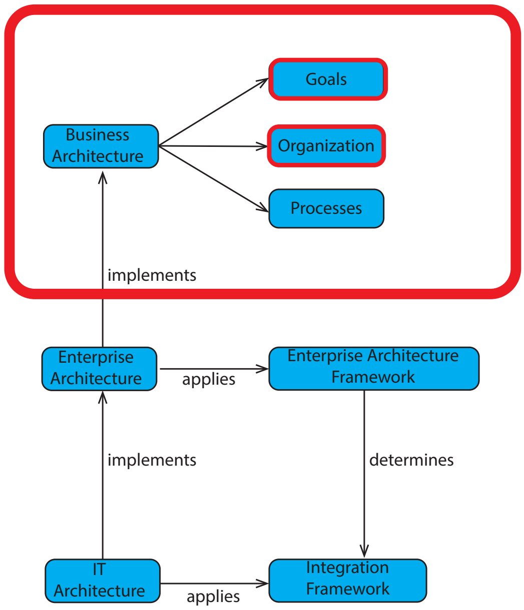


Figure 2.1: The Organization and its Goals: Part of the Business Architecture

2.1 Description

HAWAICycle GmbH produces customized bikes by applying mass customization[2]. The bikes are delivered to foreign and domestic customers. For customers within the same geographic area, various additional services are offered as well, such as on-site services. Customers from other geographic regions use service providers. The company as a whole has one site, located in an industrial park in Hamburg, Germany, where both production and management are located. Most employees work in Hamburg, but the Key Account Managers are visiting important customers in the field. Furthermore, the company works with sales representatives located within the targeted geographic areas.

[1] The GmbH is the selected corporate body for HAWAICycle - it is the type of incorporation most often selected for a capital company in Germany.[Wel] It is a privately held limited liability company and as such a corporate body governed by private law similar to a PLC (UK) or a LLC (USA). The regulation for a GmbH is available in English at [Deu86]

[2] see section 3.3.1 for an in-depth classification of the production process

2.2 Product

HAWAICycle sells custom bicycles. A bicycle is generally defined as any two-wheeled vehicle in one line designed to be propelled solely by the rider(s).[US 95][p. II-3]. Bicycles are classified first and foremost by wheel size (diameter), basic type, and number of speeds. [US 95][p. II-4]. Bicycles consist of components including:[US 95, p. II-4 - II-5]

- Frame
- Fork and handlebar
- Saddle
- Wheels, tires and brakes
- Chain, chain rings, pedals
- Lighting and wiring (depending on the bike type)
- Small parts (e.g. bell) and auxiliary materials such as lubricants

As for the HAWAICycle, most parts of the bikes are prefabricated and of high quality. As the frame has to be built individually for each customer he or she has the free choice of a color. Standard colors are ware-housed, but HAWAICycle is ready to order any other requested color. Most of the components for bikes are produced in South-East Asia. This assumption has been verified by researching the production locations of two large part suppliers, Shimano and SRAM. Shimano has production facilities in China [Shi], SRAM has a production facility in Taiwan [Eve15]. The World Bank also backs the assumption regarding the location with a case study.[KM15, p.95ff.] This case study states that since 1995, Bangladesh exports of Bicycles and Bicycle parts to the EU are growing gradually. All these findings support the initial assumption that most bike parts are produced in South-East Asia. In result: Due to the long and time-consuming distances, the logistics and warehousing is relevant to HAWAICycle.

2.2.1 Customers

HAWAICycle has two groups of customers:
The first target group are bicycle shops which order these bicycles for their customers, offering an alternative to pre-existent off-the-shelf bicycles.
The second group of customers are private customers that use the HAWAICycle web shop to order the bicycles directly from HAWAICycle.

In both cases the typical size of the order is small (less than 5 bikes per order) and all bikes produced differ as the frame of each bicycle is adjusted for each customer. For this reason all customers are required to pay their purchases in advance. Consequently no returns are accepted for products sent out to customers except for faulty products. In the case of individually manufactured products, the rules of the german long distance trade act concerning the revocation of the purchase agreement are not applicable.[Brä14]

2.2.2 Organizational structure - Hierarchy

The corporation has several divisions which are further divided into subdivisions. The divisions identified are:

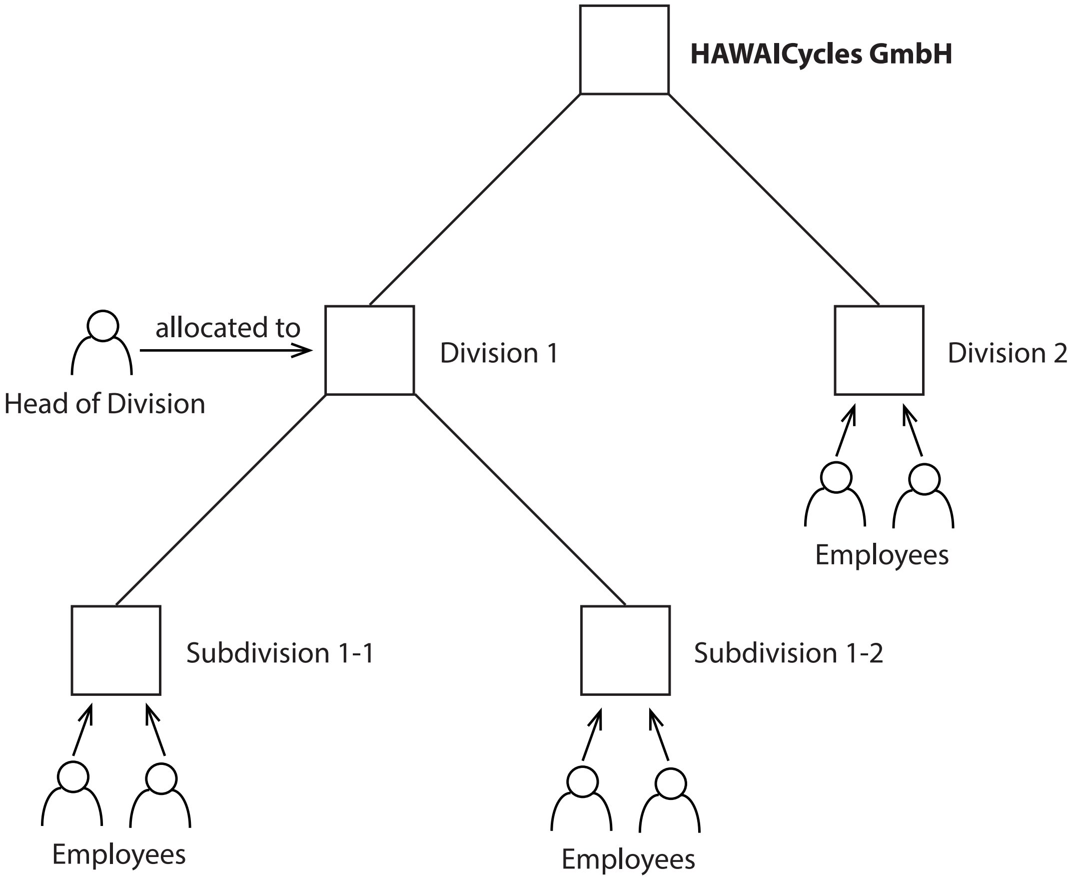


Figure 2.2: Organizational Structure of HAWAICycle GmbH

- **Production**
  This division is responsible for the production of the bikes

- **Marketing**
  This division is responsible for the Marketing activities, which is the activity, set of institutions, and processes for creating, communicating, delivering, and exchanging offerings that have value for customers, clients, partners, and society at large. [Ame] For HAWAICycle, this includes the preparation of magazines for the bicycle shops, the management of the website and mailing activities.

- **Distribution**
  is responsible for getting the produced bikes from the HAWAICycle production facility to the customer in good order. Therefore, packaging prior to shipping is an essential part of the distribution.

- **Human Resources**
  is responsible for all tasks related to the human capital of HAWAICycle. This includes the management of contracts, communication with social security authorities and the payment of salaries

- **Logistics**
  In-house logistic transports (from warehouse to production and vice versa) and the handover to the carriers.

- **Accounting**
  Responsible for issuing bills for customers and recording all financial transactions as well as the preparation of finance-related statements

- **Research & Development**
  Responsible for quality control and the improvement of the products

- **Purchasing**
  In charge of ordering the parts required for production in time and negotiation of purchasing terms

The description found above is only a stub. A more detailed definition can be found in the chapter that describes the informal model in section 4.3.

3 Theoretical Background

In the previous chapter the corporation HAWAICycle has been introduced. In summary HAWAICycle is a fictitious enterprise that is built for the HAWAI lab at HAW and is currently in the planning stage. There are neither established business processes nor any software selected to execute these processes. This thesis will model business processes and suggest an IT infrastructure that is aligned with these business processes.
The alignment of business processes and IT infrastructure is the topic of Enterprise Architecture Management (EAM). The practical implementation of HAWAICycle implements an Enterprise Architecture Framework (EAF)the Iteratec Best Practice EA, which is described in more detail in 3.1.1. Section 3.1 explains the theoretic fabric behind the framework.
In a typical IT environment - like the one planned for HAWAICycle - several IT systems are used to execute processes. These systems are connected by means of integration. Two approaches for the connection of these processes - EAI and SOA - have been selected (due to their widespread use) and are going to be discussed. Lastly, all these integrated systems serve one purpose - to aid HAWAICycle in producing custom bikes on a large scale. This type of production (see 3.3.1) is typically referred to as mass customization and manifests itself in the IT infrastructure. This is relevant both to the development of an ICT infrastructure and to the present discussion of the theoretical background.

3.1 Enterprise Architecture and Enterprise Architecture Management

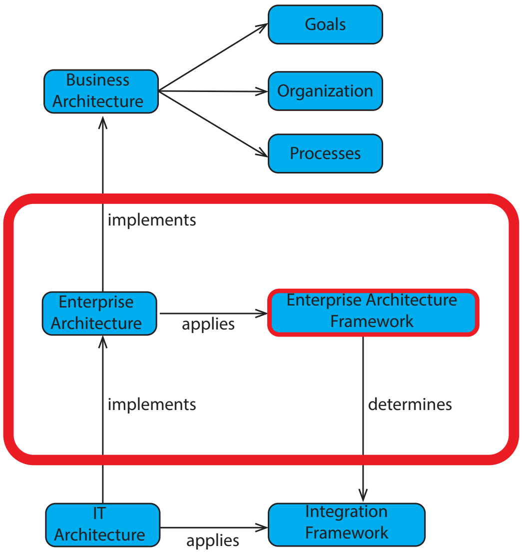


Figure 3.1: EA and EAM in the Context of this Thesis

The practical part of this thesis implements an Enterprise Architecture Framework, the Iteratec Best Practice EAM, which is described in more detail in section 3.1.1. This section explains the terminology and the general characteristics of frameworks.

The **Architecture** of a system is the fundamental organization of a system embodied in its components, their relationships to each other and to the environment. The Architecture is the result of the principles guiding the system design and evolution. [IEE00, p.3] In the case of an EA, the principles the IT infrastructure reflects are the business processes the IT system supports. A system is a collection of components organized to accomplish a specific function or set of functions.[IEE00, p.3]

The **Enterprise** in this context is any collection of organizations that has a common set of goals/principles and/ or single bottom line. In that sense, an enterprise can be a whole corporation, a division of a corporation, a government organization, a single department, or a network of geographically distant organizations linked together by common objectives.[Sch, p.22]

An **Enterprise Architecture (EA)** provides a clear and comprehensive picture of the structure of an entity.[HBT03] Enterprise Architecture therefore describes the interdependence between the IT Architecture on the one side and the Business Architecture (The entity, its goals and processes) on the other side. EA therefore includes much more than simply a description of the IT. It covers business operations, finance, people and buildings in addition to technology[The13]. It provides a strategic, conceptual and organizational framework for the management and design of the IT ecosystem [ABB+11, p.5] and the alignment of business processes and IT operations.

Enterprise Architecture Management (EAM) is a Management discipline. The Goal of EAM is the further alignment of the business and its relations to the IT by integrating and enhancing the existing Enterprise Architecture.[ABB+11, p.15]. As IT is an essential backbone

for the activities of an enterprise, the necessity to properly align the processes and the infrastructure to remain manageable is getting more important. EAM has methods for structuring the integration of ICT components and improving the Enterprise Architecture.
The proper alignment of processes and IT structure is a well-established problem. In 1987, Zachman ([Zac87]) described the need for a logical construction architecture for defining the integration of systems.
As a result, he constructed the first **Enterprise Architecture Framework (EAF)**, the Zachman Framework, to provide a practical guideline for the alignment. An EAF establishes a common way for creating, interpreting, analyzing and using Architecture descriptions within a particular domain of application or stakeholder community. [ISO11, p.9]
By way of example, the Zachman framework realizes this by providing six different views on:

- how an IT entity operates (a system)
- what resources it uses
- where it operates
- who uses the system
- when the system performs operations and
- why it operates

This framework, while discussed in literature, is regarded as either too complex for a complete adoption (see the case study on [Ahl12, p.212-213]) and/or as too generic to provide any tangible output (Bente in [BBL12, p.108]). Today, a great number of alternative frameworks exist for this reason. Currently used well-known EA frameworks include [Han09, p.61-62]:

- TOGAF (**T**he **O**pen **G**roup **A**rchitecture **F**ramework)
  This framework has been published by the Open Group, according to which TOGAF is one of the most widely used frameworks [The11b] [The]
- FEAF (US **Fe**deral **A**rchitecture **F**ramework)
  Developed by the US Government for US Government Organizations
- DoDAF (**D**epartment **of** **D**efense **A**rchitecture **F**ramework)
  Used for projects and enterprise in the US military context
- E2AF (**E**xtended **E**nterprise **A**rchitecture **F**ramework)
  The E2AF builds on FEAF and TOGAF and combines it with practical experiences from the application of EA frameworks

The EA framework used in this thesis - Iteratec EA - is described in more detail in subsection 3.1.1 below.
EAM and the framework used in this thesis are normally applied to an existing IT infrastructure to support the further development of the ICT infrastructure. In the case of HAWAICycle, no infrastructure exists which at first glance rules out EAM. However, although it is not common, the principles of EAM can be applied to new enterprises. The Enterprise Architecture simply needs to implement the goals and processes of the Business Architecture. This implementation is supported by using a suitable framework for guidance, in this thesis the Iteratec EA. Consequently the IT architecture should implement the Enterprise Architecture.
Since the EA implements the Business Architecture (see 3.1) by applying an EAM, in a first step the process of developing an ICT blueprint includes the modeling of processes - based on the informal model described in chapter 4.
The IT architecture can be developed after creating an EA, depending on the following IT structure (see the sections explaining EAI (3.2.1) and SOA 3.2.2 for further information).
This **IT Architecture** consists of the formal description of a system, a specific plan of this system and its components, the structure of the components and its interdependencies as well as the principles and design guidelines that have been used for the design, construction and implementation of that system.[Mas07, p.21]
When completed, this IT structure will consist of several independent subsystems that need to be integrated to form a functioning system that executes the business processes. The approaches most relevant for the integration of independent systems are SOA (see 3.2.2) and EAI (see 3.2.1).

3.1.1 Iteratec Best-Practice EA

The previous section explained general characteristics of EA and EAM, but this thesis uses a specific implementation, the **Iteratec Best Practice Enterprise Architecture Framework** (Iteratec EA) for the modeling of both the Business and the Technical Landscape.
The framework is described in both a manual Iteratec provides and a book by one of the co-designers of Iteraplan. This book criticizes other frameworks for their high abstract level and their scientific approach. According to the author, Inge Hanschke, this prohibits an ad-hoc use[Han09, p.65], which was - according to her - one of the reasons for the creation of the EA framework. Instead of a scientific approach, the Iteratec EA uses a Best-Practice approach and allows a practical use without knowledge of the theoretical background. In this EA, all IT systems are added and the systems are connected through interfaces. The interfaces transport business objects which need to be added, too. Additionally, the framework also allows the

management of the underlying technical structure (in the technical landscape model). This results in a mapping of the the current state of the Infrastructure and a complete application landscape. Next, the business processes are added and used to align the applications to these processes. By mapping the business landscape (processes) and the application landscape model(applications) the previously invisible connections become apparent.
Apart from the mapping - which requires the existence of the elements - all these steps can be completed independent from each other. Since the formal model created uses the framework it is explained further in the following sections.

Framework Meta model

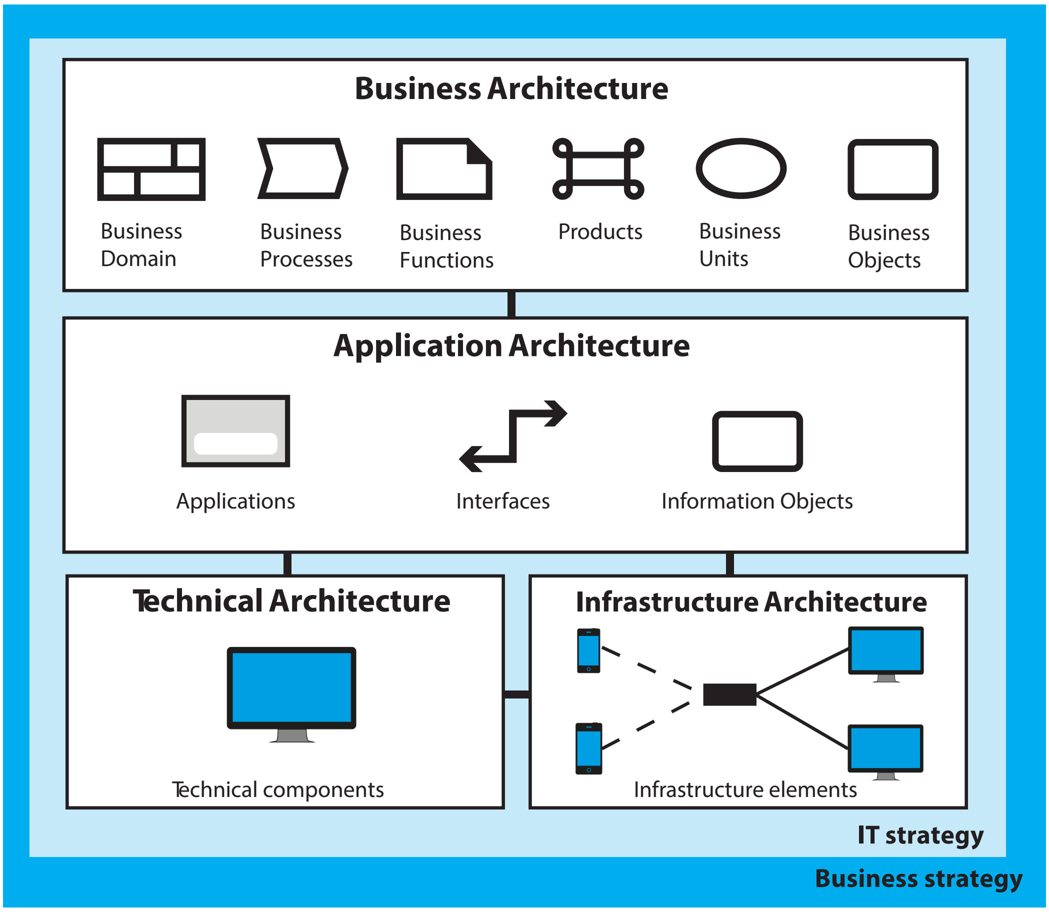


Figure 3.2: The Iteraplan EA Model
Based on [Han09, p.66]

The meta model of the framework uses elements called **Building Blocks** on all Landscape Models. Three Landscape Models exist - Business Landscape Model, Application Landscape Model and Technical Landscape Model. This section is a summary of the two sources available, [Han09, p.66-93] & [Itec].

Business Landscape Model The Business Landscape Model contains all relevant structures for the business and allows the modeling of Business Processes, Products, Business Functions and the organizationals structure and maps relationships between the individual Building Blocks. [Han09, p.71]. The following Building Blocks are part of the Business Landscape Model:

- **Business Domain**
  The Business Domain is an optional grouping element to allow the combination of related Building Blocks. The Grouping into individual Business Domains is possible by Business Processes, Products, Business Functions, Business Units and Business Objects. [Han09, p.91]

- **Business Processes**
  A Business Process is a recurring sequence of logically connected activities or sub-processes with a defined start and end point that contribute value to the enterprise [Han09, p.71], e.g. Quote-To-Order. In the current version, it is not possible to implement looped structures.

- **Business Function**
  A Business Function is a recurring task carried out by the enterprise. Business Functions can be organized into sub-activities It has connections to other Building Blocks. Business Functions can be used in multiple Business Processes. [Han09, p.71]

- **Product**
  The result of a Business Process, e.g. a bike

- **Business Unit**
  A structural unit of the enterprise (e.g. department, locations, logical groups) [Han09, p.71], e.g. Production or Sales

- **Business Object**
  A Business Object represents a concrete or abstract real-world entity that encapsulates some part of the Business Activity of an enterprise, e.g. customers or products. Business objects can be composed of other Business Objects (e.g. an order consists of a customer

address, a delivery address and the ordered articles). The objects are used by Business Processes or Functions. Business Objects can be linked to each other. [Han09, p.72]

Application Landscape Model The Application Landscape Model is documented as part of the IT Landscape and ties together the Business Landscape (with the business processes) and the Technical & Infrastructure Landscape. The Application Landscape helps to weed out the IT infrastructure by segregating important applications from unimportant ones and helps identifying connections between applications. Additionally, the Landscape allows the planning of future Application Landscape states [Han09, p.78]. The following Building Blocks are part of this model:

- **Information System Domain**
  The Information System Domain can be used for grouping information systems [Itec]

- **Information System / Application**
  An Information System (Iteraplan) / Application (Hanschke) is a cohesive entity which users perceive as a technical and functional unit (Definition from Siedersleben cited after [Han09, p.78]). In general it supports associated Business Functions distinct from other areas of functionality, and which can be supported entirely or to a large extent by IT.[Itec]

- **Interfaces**
  The interface connects two applications (information systems) and allows the unidirectional or bidirectional exchange of information objects. [Han09, p.79]

- **Information Objects**
  Information Objects are sent over Interfaces from one application to another application. [Han09, p.79]

Technical Landscape Model The Technical Landscape Model defines standards (standard applications, technical standards) specific to the enterprise for implementing applications, interfaces and infrastructure. It gives information about in-house standards.[Han09, p.85] One blueprint that can be retrieved from Iteraplan is the technical blueprint - a list of standardized components.[Itec]

Infrastructure Landscape Model The Infrastructure Landscape Model coarsely describes the infrastructure (e.g. the computers, cloud services,...) on which the applications run.

[Han09, p.85] In this landscape, applications are linked with the infrastructure. Iteraplan allows the integration of a CMDB (Configuration Management Database) to retrieve the actual Infrastructure from the Database.

Evaluation of the Iteratec EA

The Iteratec EA has a simple structure, but allows the combination of different methods important for Enterprise Architecture. Most importantly it allows the modeling of business processes and the mapping of business processes to the ICT infrastructure through links. The Enterprise Architecture also allows to link IT systems through interfaces, which makes it suitable for modeling both a service-oriented architecture and the Enterprise Architecture Integration.

3.2 Integration Frameworks

As explained in the introduction to the Theory chapter the IT environment planed for HAWAICycle integrates several IT systems to execute the business processes that are created in the business architecture.
These systems are connected by the means of integration. While it is entirely possible to connect the systems organically, the results of this approach that is called P2P is often less than ideal. For this reason, integration frameworks exist to allow the systematic integration of systems.
Two approaches for the connection of these processes - EAI and SOA - have been selected and are discussed.

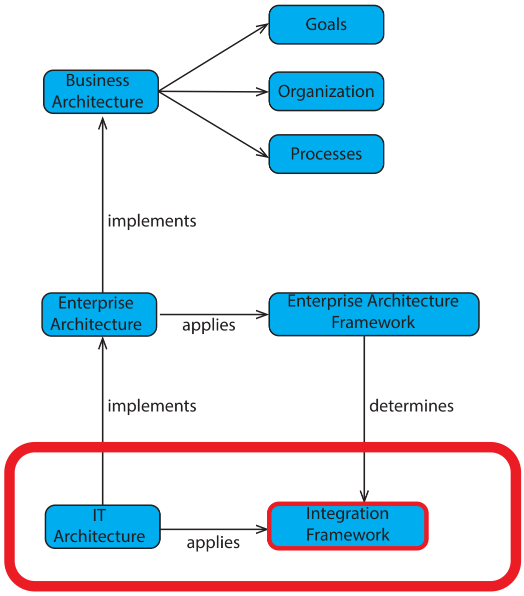


Figure 3.3: EA and EAM in the Context of this Thesis

3.2.1 Enterprise Application Integration (EAI)

EAI is the name of a structured integration approach to information systems within an organization. EAI includes building wrappers, adapters, and using standardized middleware (see section 3.2.1) to connect and integrate the systems. [LC04] This combination of processes, software, standards, and hardware resulting in the seamless integration of two or more enterprise systems allows the systems to operate as if it were a single system.
The systems - called applications in this con-

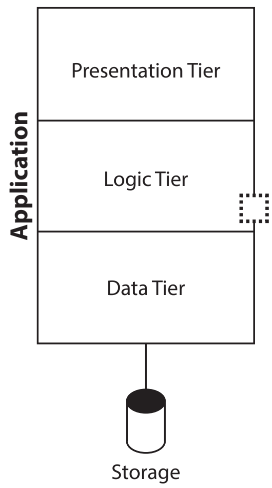


Figure 3.4: Monolithic Application Architecture - Three Tiers

text - itself have a monolithic structure which means they solve specific enterprise tasks and include a data input / data output, some logic to cope with the data to solve a specific task. This chapter focuses on integrating applications, which is known as Application-to-Application (A2A). With this approach, the individual applications are still used for data input/output. It is also possible to integrate the systems by supplying a common user interface - Person to System (P2S), integrating several enterprises (Business to Business (B2B)) and to combine all these integrations (Total Business Integration - TBI). These kinds of integrations are not planned and therefore not further discussed in this thesis. The integration of applications (A2A) can occur on different levels. These are known Data-Level, API-Level, a Application-Method-Level and UI-Level integration [Lin99] [Fen]. The different types use different tiers (see 3.4) to integrate other applications. The integration results in synchronized and therefore consistent data.

Data Consistency / Synchronization

Data consistency in this context means that the data available is coherent and complete. [Lac] In the case of connected IT systems, data can often be saved and updated in several systems. Data consistency is relevant for this thesis because several applications are used that need access to the same data. For instance, HAWAICycle includes a web shop where customers can order bikes. All customers can login into the web shop for ordering bikes from HAWAICycle and all details regarding customers and orders are stored in the web shop database. For different reasons (e.g. for mailings and analysis purposes) the customer data is also stored in the CRM (Customer Relationship Management) Tool. If the address of the customer is updated in either one of the systems, consequently it needs to be updated in all systems concurrently to ensure consistency.[SA12]

Integration Levels

Generally, an application can be connected with other applications outside of the application itself or within the application. When integrating outside of the application by accessing the data created by the application in a database or in a file the application itself is not modified. When integrating within the application, the application is modified on different tiers (see figure 3.4). Depending on the tier that is changed or accessed this approach is further classified as either an Application-Interface-Level integration, a Method-Level integration or a UI-Level integration. [Lin99, Ch. 3 - Application Interface-Level EAI]

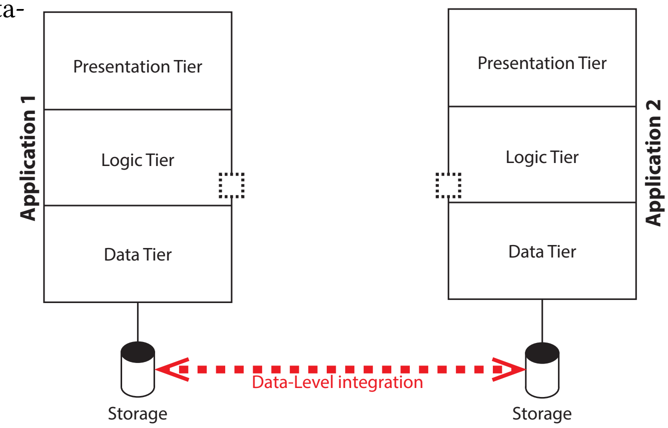


Figure 3.5: Data-Level Integration
An Adaption from [Lin99, Figure 2.1]

Data-Level Integration With Data-Level integration the applications are integrated on their storage level. Information is converted on-the-fly between the different databases and thus represented correctly both by source and target application. This task is further simplified by the fact that accessing databases is simple and most current applications use external databases for data storage, since decoupling data from application logic is a common practice. However, application and data are often tightly coupled and thus, dealing with the data without dealing with the application logic is impossible. When adding data into the database the integrity checks that are done on an application level are not applied. The integrity constraints on application level need to be dealt with to prevent the applications from crashing.
Application-Method level EAI (see section 3.2.1) is an approach that might be employed alongside or instead of Data-Level integration. However, the Data-Level approach is simple and fast to apply since the business logic does not need to be altered. Also, numerous middleware products that allow migration on a data-level exist. [Lin99, Chapter 2 - Data-Level EAI] recommends against using a Big-Bang approach when doing a Data-Level integration due to the complexity and difficulty of this approach and recommends a start with two or three-databases.

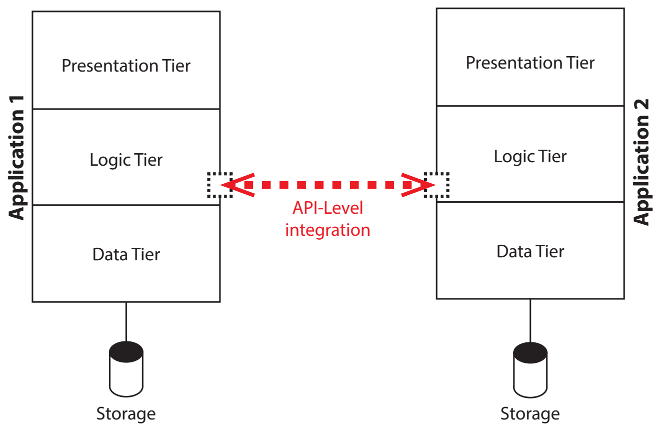


Figure 3.6: Application-Interface-Level Integration

API-Level Application Interfaces (APIs) are well-defined interfaces that the developers of an application publicly expose to allow access to various levels or services of the application. Compared to data-level-integration, where the application and storage logic needs to be understood, APIs provide access to objects on a higher abstraction level. APIs typically provide access

to data stored in the application database, to business objects and/or to business services. For instance, a CRM application might provide read access to the customer as a data object, the customer as a business object (which could be an aggregation of various data objects connected to the customer) by invoking an API call for the data or the business object. Additionally, the application may provide a business service with write access. For instance such a business service could create a new customer and thus provide an opportunity to add new data. [Lin99, Ch. 3 - Application-Interface-Level Integration]
If an API-Level integration is possible, it ideally provides more information than a data-level-integration. However, the functionality provided by the API is limited to the functions that the original developers of the Software have provided. [Lin99, Ch. 3.4]

Application-Method-Level The intention of Method-Level integration is sharing business processes. Unlike the API-Level integration discussed in the previous section, this approach changes the Application which needs to be integrated by adding or replacing program code in order to allow the integration to take place. This is also indicated in figure 3.7. integration in this form results in tight coupling of different applications in the application landscape of the corporation. By adapting the application the software is no longer restricted to the functionality intended by the original application developers. However, Method-Level integration has several downsides. One of these is the cost for application changes - which is significant. [Lin99, Ch. 4 - Method-Level integration]. Also, while the tight-coupling of the applications may be a good choice for performance, maintainability suffers as a result from the decision.[Man09] Additionally, a poor implementation may cause applications to malfunction. [Lin99, Ch. 4 - Method-Level-Integration]

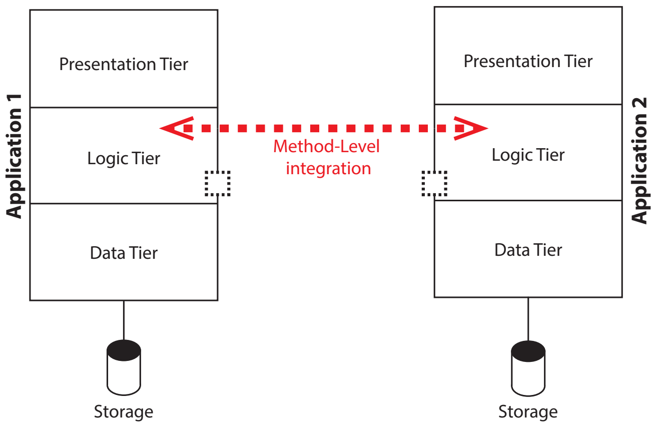


Figure 3.7: Application-Method-Level Integration

User-Interface-Level Integration

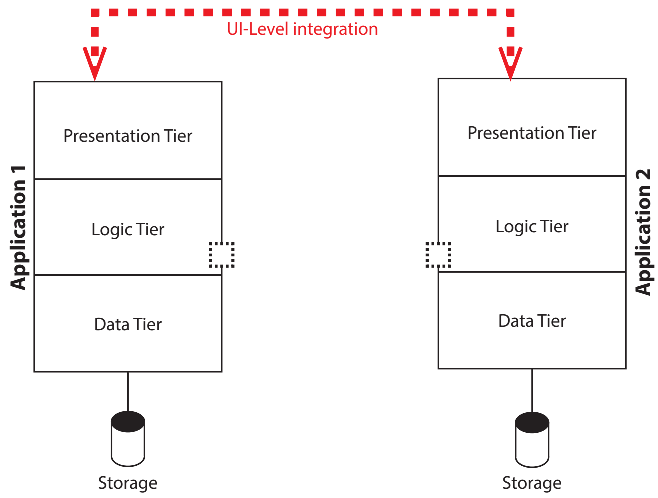


Figure 3.8: UI-Level Integration

Integrating on the user interface (UI) level doesn't change the program code and is thus similar to API level integration. However, it is technically different because when applying this approach, the UI is used by the integration mechanism - a user is emulated for this purpose. The method itself is crude and relatively inefficient, however it doesn't change the target application and - correctly implemented - gets the job done. Key disadvantages include slow data retrieval due the UI being an inappropriate data-serving interface, possible instability due to poor implementation, and the necessity to implement error processing for the integration. One last downside should also be mentioned: The flow of information depends on the number of application instances that are running at the same time. As a result [Lin99, Ch. 5 - User Interface-Level EAI] as many as 1000 Instances of the target application need to be running at the same time. The actual number of instances is irrelevant, but signifies that this method may result in a very high system load and scales poorly.

Integration Topologies

The previous sections focused on approaches on how to connect individual applications. Setting up these connections is essential. However, the larger context that results out of these connections has more consequences, since it influences concerns such as maintainability, fault-tolerance and performance. The design of this integration context should specify the locations, structure, and channels that is used to connect these elements together to form a coherent whole. This context is called an integration topology.[TRH+04]

P2P The Peer-to-Peer (P2P) integration is considered as a traditional integration approach that is useful if there are only a limited number of applications to integrate.[SA12] A connector - called pipe - has to be implemented for each communication between systems. The connector handles the transformation between the systems and is responsible for the delivery of the messages between the systems. As already mentioned, this integration method is useful for a small number of communications and systems, as the complexity and number of pipes increases with each new link added. This correlation can be seen in Figure 3.9, where twelve pipes exist for just four applications. Although the scenario shown is unlikely, it illustrates the problem at hand. For this reason and because P2P has poor maintainability it is not wise to use this integration topology.

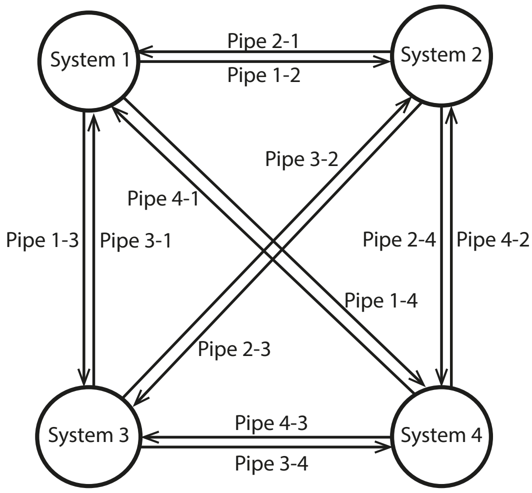


Figure 3.9: P2P Integration

Middleware Instead of connecting systems directly (P2P), middleware can be used to mediate between systems. [Fen] Essentially, middleware sits in the middle of the systems (thus the name) and the systems are connected to the middleware component. The idea behind using the middleware is to decouple the systems from each other and also to allow some re-usability of the connectors.
Assuming systems 1-4 in figure 3.11 all synchronize data objects regarding customers to each other because each system is both used to read and write informations regarding customers. If data is exchanged using P2P, twelve different transformations routines are necessary (1-2,1-3,1-4,2-1,2-3,2-4,3-1,3-2,3-4,4-1,4-2,4-3), whereas the middleware approach requires only eight transformations routines.
Different implementations for middleware exist - Hub&Spoke (see figure 3.10) and Bus/Pipeline/Publish&Subscribe. The main difference between Hub&Spoke and Bus is that the Hub is a central instance while the Bus is a distributed system. The Bus provides a higher fault tolerance but requires more effort for the initial implementation. The Hub on the other hand is a potential bottleneck and a risk - a failure prohibits all attached applications from exchanging data.

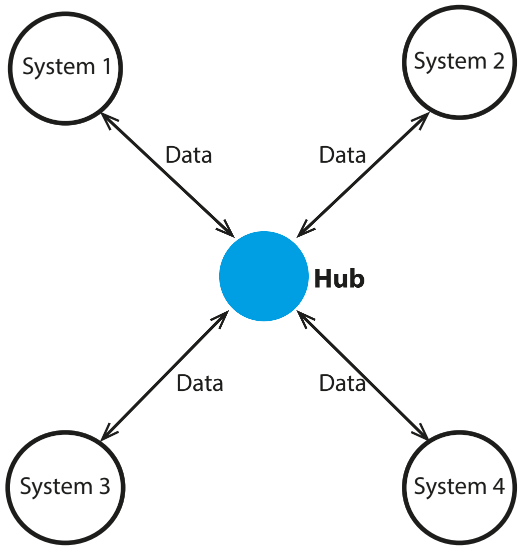


Figure 3.10: Middleware: Integration with Hub&Spoke

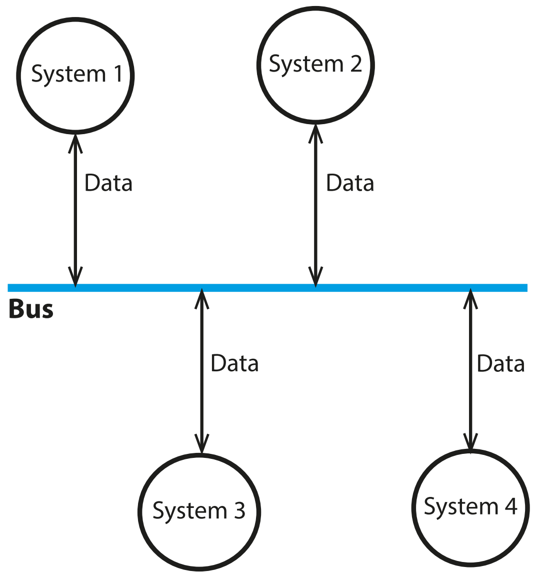


Figure 3.11: Middleware: Integration with a Bus

Comparison of P2P and Middleware

From an architectural point of view, using middleware instead of connecting systems directly is clearly better. middleware allows re-usability of software and makes the replacement of a component easier. In the long run, this should also result in lower costs for the enterprise. The only valid reason for preferring P2P over middleware would be a one-off-integration (e.g. a Prototype) with a limited number of applications.

3.2.2 Service-Oriented-Architecture (SOA)

In the mid-1990's, the **service-oriented architecture (SOA)** integration framework emerged as an advanced alternative to EAI for interweaving business processes.[Raj13] SOA focuses on the strategic development of IT according to the business objectives [The09] and thus is a versatile topic. This chapter focuses on informing about which improvements a SOA can provide when using the approach in managing the Enterprise Architecture.

Definition of SOA

SOA can be defined as a software design and implementation methodology (architecture) of loosely coupled, coarse-grained, reusable artifacts (services), which can be integrated with each other through a wide variety of platform-independent service interfaces.[Ste05]
As explained earlier, EAI integrates monolithic applications with a specific set of functions as Building Blocks. In SOA, services serve a similar purpose as applications serve in EAI. However, services are not monolithic - they are composable units. Also, services - unlike monolithic applications - are intended to be reused/integrated with each other, therefore services by definition contain an API.[Mas07, p.18] Legacy applications can be integrated into a SOA by creating a service wrapper (see figure 3.13) that contains the required API and maps the API to the legacy application functions. The preferred way of realization of both the API and the application is in the form of web applications that provide standardized web interfaces (see section 3.2.2). Web interfaces in general result in the possibility that external providers (other companies) can provide web services on demand, which is increasingly common today. This means that a well-implemented SOA can be and probably will be an arbitrary mix between external and internal services. Also, when comparing EAI and SOA, services are only a part of SOA. While definitions of SOA exist that focus only on technical aspects, SOA should be regarded as an Architectural Style for application landscapes that aims at an optimal IT realization of IT processes. [AE08]

SOA, EA and Service-Oriented EA

SOA aligns strategic business goals and operative IT objectives with the introduction of the Service Layer. This Service Layer – as displayed in figure 3.12 – covers the GAP between Processes and IT.[The11a]
The service-oriented enterprise architecture (SOEA), as introduced by [AE08] is a way of aligning the business architecture (based on goals, organizational constraints and processes) and the IT Architecture. In this concept, an Orchestration Engine is used to model processes with services. The software based wiring (soft-wiring) of Services allows to change processes without the need of special programming skills and furthermore it permits to change processes fairly simple.

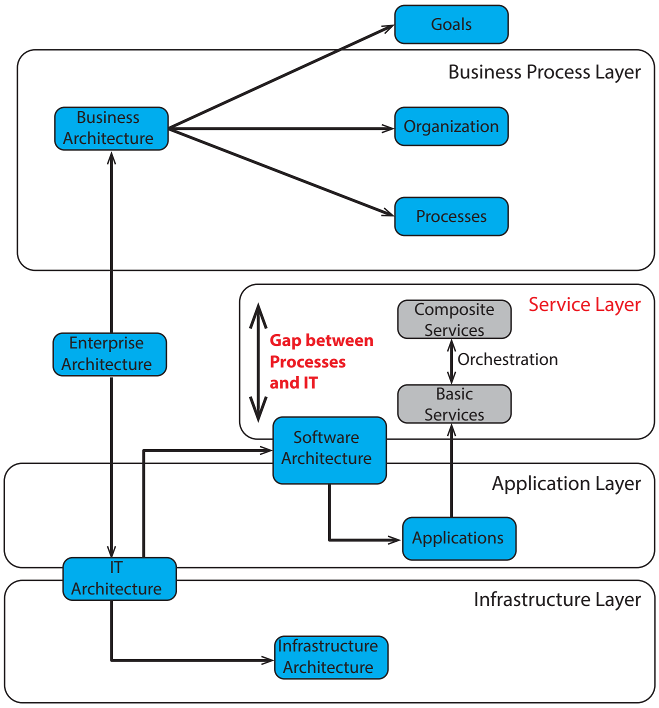


Figure 3.12: SOA and EA
Source: [AE08]

This approach improves the alignment of business processes and the IT structure. However, this approach is no solution for the structure of an organization, it only aligns the aligns the business processes and the IT on a technical level. The organizational framework still needs to be created before employing this approach.

Definition of a Service

Services in the context of service-oriented architecture... [Mas07, p.16]

... provide capabilities or functions such as customer look up or credit card validation

... are instantly usable.

... can be reused. Re-usability is a key element in this definition because it enables the creation of new business processes and operational processes. Services expose their capabilities via well-defined standard service interfaces.[Lew10] These standard service interfaces are usually based on web standard technologies. [SW04]

... have a well-defined behavior

... have defined inputs and outputs.

... are managed to fulfill non-functional requirements.

... are created and used to fulfill an organizational goal.

... can be modeled

... are composed to create new Services.

Legacy applications commonly miss some or all of the above capabilities. Most importantly, they are usually not composable. However, if they are sufficiently modularized, they can be wrapped into services. [OSHS07]
Both wrapped legacy applications and "natural" services have a service interface definition that is published in and available through a service registry. [Lew10]
The *composition* of services is possible by hard-wiring services (i.e. a service uses other services directly) or soft-wiring services (i.e. a software is used for the composition of services). In the context of the application landscape, software-based composition of services (soft-wiring) in an Orchestration Engine allows the modeling of business processes with services and the subsequent execution improves the alignment of business processes and IT Structure. To allow better alignment, services should use an event-based communication style (publish-subscriber pattern).[AE08]

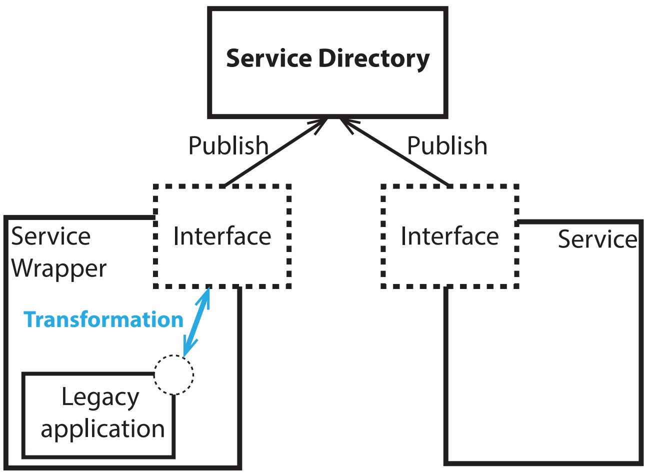


Figure 3.13: SOA - Encapsulation of legacy applications and native services are published in a service directory

Service Interfaces - REST and WS*

Service interfaces in SOA ideally use standardized interfaces to allow easy composition and (re-)usability. The most common standards used for providing these standardized interfaces are *WS* Web-Services* and *REST-Interfaces*. REST stands for **Re**presentational **S**tate **T**ransfer. Both interfaces types are still being developed and their use includes some pitfalls.
WS* Web Services define a method of transportation (HTTP), encoding (XML), a message format (SOAP) and a description language for the web service (WSDL) as well as methods for the discovery (UDDI) and the management of the services. However, there are currently more than 100 competing implementations of Web Services. While WS* guarantees how the language is composed – i.e. the syntax – the semantic is still wanting. This results in the need for a transformation engine.[Lew10, p.4-5]. Often, Web Services are perceived as very complex and inefficient due to the overhead imposed by XML/SOAP.
REST does not contain a definition of these elements. Instead, REST - typically used in conjunction with HTTP - has the following four principles [PZL08]:

- **Resource identification through an URI**
  Each resource is identifiable through a unique URI - usually in the form of an HTTP link
- **Uniform Interface**
  The four CRUD-Operations[1] mapped to the HTTP commands PUT, GET, POST and DELETE.
- **Self-Descriptive messages**
  The messages are decoupled from the representation.
- **Stateful interactions through hyperlinks**
  Every message is self-contained.

REST is simple to implement since the infrastructure required (HTTP Clients and Servers) is very common and the REST services are therefore easy and inexpensive to implement. This creates a much lower entry barrier for using REST-services compared to WS*. However, the implementation of the Uniform Interfaces poses some problems because firewalls may block HTTP out requests partially – typically only POST and GET are allowed to pass through firewalls – which might require workarounds. Additionally, the payload data in GET requests is limited in size and exceeding the size limits may result in the rejection of the request.
Choosing the right service interface for HAWAICycle is not a trivial question. An in-depth

[1] CRUD = Create, Read, Update and Delete

view of this topic is provided by [PZL08]. It should be noted that most commercial services offer both type of service interfaces.

Evaluation For HAWAICycle, the primary use of REST for the first implementation is of advantage for the following reasons:

- The use of a web browser is sufficient to test assumptions and requires no coding.
- From a technical point, REST is easier to implement than WS*. This makes it more suitable for implementation by students in lower classes - HAWAICycle is also intended as a training environment for students.

Communication between Services with the Enterprise Service Bus

As explained in section 3.2.2, WSDL only guarantees the syntax, not the semantic of the messages and REST messages guarantee only that resources can be accessed through an URI, but does not give any guarantees about the message content. To allow the coupling of services, a transformation between the different models is still required. If integrated directly, the problems that result out of a direct integration are similar to the problems in EAI with a P2P Integration Topology. In result, there are a lot direct connections and essentially a P2P communication style results out of this. (see section 3.2.1 for further information).

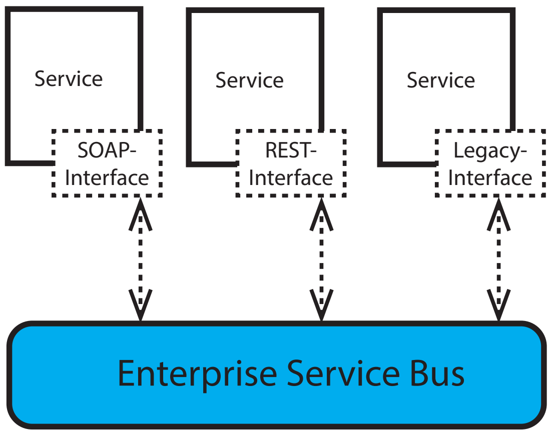


Figure 3.14: Enterprise Service Bus

A solution to solve this problem is the Enterprise Service Bus (ESB). The ESB is a Message-Oriented middleware (MOM) that provides routing, invocation and mediation services to facilitate the interactions of services.[Men] An ESB has to provide audit and logging facilities for monitoring infrastructure and integration scenarios and possibly also for controlling the business process execution. [Men] As a result, when using an ESB, Services are only registered and available through the ESB that handles both communication and transformation.

Monitoring

Business-Level-Monitoring is one of the capabilities that can be achieved through the use of an ESB that controls the Business Process Execution. This allows the management to follow the execution of processes and makes the identification of challenges (e.g. shortages, increasing processing times) easier. In the case of HAWAICycle monitoring might give both students and instructors valuable information about problems. Furthermore, monitoring is useful for the real-time visualization of processes.

3.2.3 Comparison of SOA and EAI

Both EAI and SOA provide mechanisms that result in an integration that is superior to an unplanned (organic) integration. Both approaches improve the re-usability of written code and make the upgrade and replacement of components easier. The biggest advantage of SOA over EAI is the alignment of the business and the IT architecture. In case that SOA has been properly implemented, it allows the monitoring of all business processes. As this is of special value, SOA has to be given priority as the most suitable implementation style for HAWAICycle. Last but not least, all these systematically integrated systems serve a purpose - HAWAICycle produces custom bikes on a large scale. This type of production (see 3.3.1) is typically referred to as mass customization and manifests in the IT infrastructure. This makes it relevant when developing an ICT infrastructure and therefore a relevant last section for discussion in the theory chapter.

3.3 Business Domain

The previous theory sections focused on technical aspects of developing an ICT infrastructure. However, this infrastructure serves the purpose to enable the business operations. As has been explained in the Introduction (see chapter 1), HAWAICycle produces bikes according to customer specifications. This type of production has some special characteristics that also manifests itself in the ICT infrastructure. Therefore, this last section of the theory chapter will give some information about mass customization.

3.3.1 Mass Customization

Mass customization is a way of production in which products are manufactured according to specifications set by customers. The strategy has become more important in the last years as producers seek to find new markets in more or less saturated markets by serving micro-markets, as in the case of HAWAICycle. Serving micro-markets is of competitive advantage. This shift from a sellers' to a buyers' market [KH12] has led to a point where the mass market is proclaimed dead (Kotler: "The mass market is dead" [Kot89]). To balance this provocative statement: The reports of the death of the mass market have been greatly exaggerated, however, since many products are still produced on a mass scale. In reality, mass market producers and mass-custom producers serve distinct market segments.[Kot96]
This is even clearer from an economic point of view, since customization results in a loss of economies of scale. [PMS04, p.438]. However, in saturated markets it provides a possibility to sell goods in micro-markets which mass-market manufacturers cannot realize since the market is simply too small for the mass-production of goods.
Today, two different approaches for mass customization exist. The classical variant for mass production is modularization - the custom combination of mass produced parts (an individual bike with a standard fork, a standard frame,...) [FS11, p.35-36].
With Direct Digitized Manufacturing (DDM) and the Smart Factory, a second approach exists today. This kind of mass customization relies on a high-level of IT integration and it includes and at the same time requires tools such as automated milling machines or 3D printing to produce custom parts automatically. These parts are finally combined and result in a truly individual item which is nonetheless produced on a large scale. This approach, which is also publicly discussed as *Industry 4.0* is further described in [Kul15].

Types of Mass Customization

Usually four different types of mass customization can be found in Literature. The differences consist of the level of customization and by the amount of items produced.
Build-To-Stock (BTS) is the common name for mass production of goods. Items which have

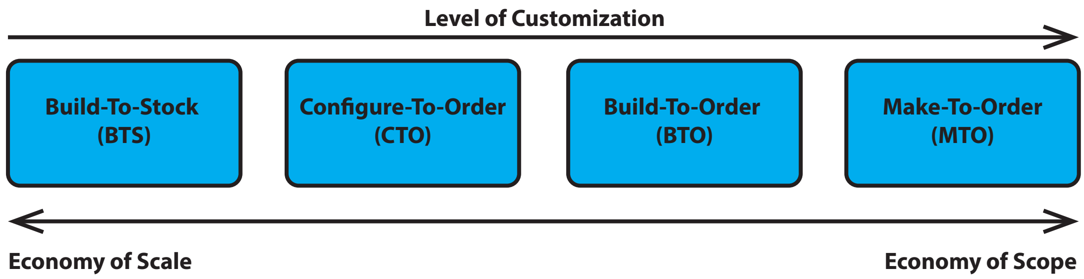


Figure 3.15: Levels of Customization

been produced on a large scale are warehoused (stocked) afterwards. This kind of production is characterized by large quantities being produced. So generally customization is possible if the customer buys on a large scale at once.
Configure-To-Order (CTO) allows customers to define the component make-up of a product when ordering. [ZX11, p.1410] an item, selecting from a (limited) variety of stock items. The product is then produced on a mass-scale in an assembly-line process.
With **Build-To-Order (BTO)**, orders are only built after they have been ordered. BTO and Make-To-Order (MTO) are often used interchangeably. However, the difference between Make-To-Order and Built-To-Order is that BTO uses only parts that are already in stock. [GN05, p. 424].
In a pure **Make-To-Order (MTO)** Scenario on the other hand, all parts are ordered or produced only after a customer sends his order. This approach allows greater flexibility, but the other side of the coin is the production - it takes more time. The mixture of both scenarios, the one with orders on demand blended with the one on stock exists, but without a specific name. The scenario could in this case be classified by comparing the ratio of on-stock-parts and on-demand-parts to give an overall ratio-based classification for an individual company.
HAWAICycle uses a mixed BTO and MTO approach. While common prefabricated parts, such as the chain or the fork are warehoused, the frame is made completely individual for each customer. Since the most important parts are on stock and only the frame is made individually, a ratio-based approach results in a classification that the production is predominantly built-to-order. This result brings to focus that warehousing and considering lead times when ordering parts are relevant factors.

4 Description of the Informal Process Model

The most important source used for the development of the ICT infrastructure of HAWAICycle is an informal structured mass customization reference process model. This framework is included in the *SoftwareAtlas* product that has been provided for this thesis by GPS Software AG (GPS). This chapter contains informations about the model – which can be found at the end of this chapter as a full page reproduction – and will show its general advantages and weaknesses.

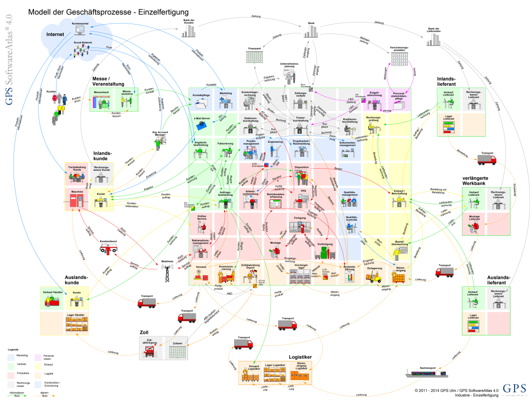


Figure 4.1: Informal Process Model by GPS Software AG

The model aims to cover all production types from uniquely designed, one-off productions up to small-batch productions - in other words essentially the full range of productions possible with mass customization (see chapter 3.3.1). For this reason GPS has covered an average range of different production types, but as it is too general, the model cannot be simply and readily used for HAWAICycle. Instead, some interpretation and adoption of the model is necessary (see chapter 5) to use it as a base for the business processes featured in chapter 7.

Concerning the ICT blueprint to be developed for HAWAICycle, the model contains only a few indications in regard to the information landscape. The elements included are discussed in section 4.4, which covers IT Integration aspects.

Generally, the model includes processes and business units assumed by GPS to exist in such a business. A business unit is a segment of the business that has its own plans, metrics, income and costs. Each business unit has exclusive access to some assets and uses the assets to create a value proposition for the customer in the form of goods and services. [ITI]. The business units are structured into divisions, which are color-coded in the plan. A division in the context

of this model is an aggregation of business units.
Additionally, the model contains external parties. The model includes flow elements for information (continuous arrow) and for goods (dashed arrow), color-coded with the department the arrow departs from (see section 4.2)

4.1 Model Intentions

GPS states that the purpose of the model is to showcase typical processes for mass customization to provide a broad overview of the processes in the corporation and to display the typical operations mapped into ERP tools. [GPSb] Furthermore, the model intends to provide exemplary solutions which are supposed to be tailored and extended to fit the target corporation.[GPSa]

4.2 Model Elements

The model contains both abstract and non-abstract elements. Non-abstract elements include elements that can be visually identified, e.g. trucks, ships and buildings.
Four repeated and abstract elements have been identified in the model.

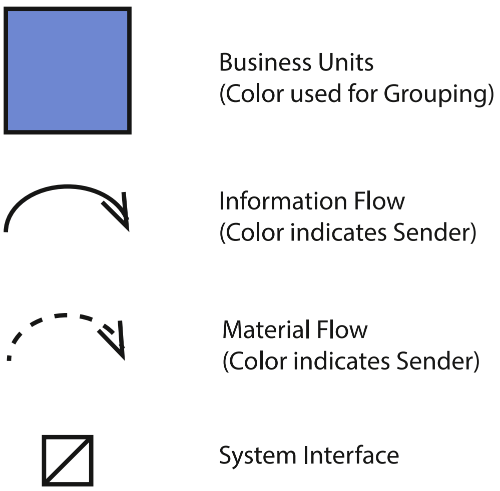


Figure 4.2: GPS Model Elements

- **Business Units (BU)**
  A colored square. The color depends on the BU
- **Flow Element: Information Flow**
  A rounded, continuous arrow. The arrow indicates the direction.
- **Flow Element: Material Flow**
  A rounded, dashed arrow. The arrow indicates the direction.
- **Interfaces**
  A colored square with an angular line

with Information Flows directing to and from the interface indicate processing of information in an IT system.

4.3 Divisions

The model consists of the following divisions: Marketing (Blue), Sales (Light Green), Production (Lilac), Accounting (Grey), Human Resources (Lavender), Purchasing (Yellow), Logistics (Light Red) and R & D(Pigeon Blue). Most of the business units are modeled within the enterprise, however the model also includes divisions of customer with the same color-coding. Within the model of the enterprise, the production is located centrally and is modeled most thoroughly. The other divisions appear more or less non-relevant.

4.3.1 Marketing

The marketing division consists of a generic business unit Marketing and of Contact Management, which could either be a business unit, a function or a software system (the model is both ambiguous). In the model, Marketing is responsible for providing product information to customers. It is safe to assume that in a normal enterprise, the marketing department takes up more resources than shown here.

4.3.2 Distribution

Sales is also documented quite scanty in the informal model. The model includes business units for factoring invoices, creating quotes and a third unit for managing the orders. The main focus of sales is on creating and editing quotations, managing orders and customer invoices. Additionally, some artifacts are attributed to the sales department, such as a mail server and an EDI-Interface [1] used for transporting customer orders to Order Management.

4.3.3 Accounting

Accounting consists of business units for Financial Transaction Processing, Financial Accounting, Accounts Payable, Accounts Receivable and Cost Object Controlling. Two essential tasks can be identified in the model. The first objective is the management of incoming and outgoing monetary flows. The second objective is the processing of these cash flows for the operative and strategic cooperation planning. For the first objective, interfaces to financial institutions (for

[1] EDI = Electronic Data Interchange

monitoring cash flows and instruct out-payments) and human resources (for salary payment instructions and sums) exist.

4.3.4 Human Resources

Human resources has two elements modeled. The first element, Payroll processing is a function executed by the staff of the Human Resources division. Payroll processing uses employment contracts to determine employee salaries. The second function, Human Resources Master Data Management keeps Employee details current.

4.3.5 Research and Development

R&D consists essentially of two different business units - projects and quality control. The project section has the business units Project Management, Engineering and Project Implementation and is responsible for creating new products. Quality Control (business units Quality Management and Quality Control) on the other hand is responsible for ensuring quality standards of production.

4.3.6 Production

The informal model allocates pre-production, production and assembly as well as production and requirements planning in the production division. Some of the components are more likely to be systems used by business units than business units by themselves and are therefore modeled as systems in the formal model later on. These include the Production planning system and production data acquisition, which are more likely to be implemented as software. For some departments such as the service hotline, the allocation into production itself is doubtful, while others such as quality control (allocated to R & D)may well be part of the production.

4.3.7 Purchasing

Purchasing is responsible for buying all goods required for the operation of the corporation. Requirements reported by Production result in orders being made. After orders are submitted, they are being monitored for (over-)due arrivals and reminders are sent out to suppliers where applicable.

4.3.8 Logistics

Logistics handles the physical transport of goods within the sphere of the corporation. It controls inbound goods by sending samples to Quality Control in the R&D division. After clearance, stocks goods in the warehouse, picks goods from the warehouse, prepares shipments to customers and hands over these shipments to the shipper. Function-wise, inventory management tracks changes in stock using a system interface. Finished products are being stocked - the model assumes that all finished products are sent to stock. Outbound Goods are being removed from stock. The model differs between products sold to domestic and foreign customers. If the product is sent abroad, customs clearance is initiated and after clearance the products are prepared for shipment. Accompanying export documents are attached to the shipment. For domestic shipments, no documentation seems to be necessary. In both cases, the shipment is being handed over to the shipper afterwards.

4.3.9 External elements

The model contains several external elements that are not part of the enterprise itself. These elements include other enterprises such as foreign and domestic suppliers, subcontractors for production, the corporations, foreign and domestic customers, financial institutions for both the suppliers and the enterprise, insurance corporations and shippers. Besides other enterprises the model includes state authorities, specifically taxation and customs. These external elements are both connected through material and information flows with the enterprise itself. Some of these information flows are direct, some are through interfaces, which is important with regard to IT Integration Aspects.

4.4 IT-Integration Aspects

As explained in Section 4.2, the model contains flow elements for Information Flows. Furthermore, the model contains some System Interfaces - mostly for connections to external partners / state authorities. The reason for this scant documentation is clear since the model has a clear focus on business operations, not IT operation and infrastructure (see Section 4.1). Therefore the model contains the required information only partially and with a focus on functionality of typical processes mapped into some ERP software.
Nonetheless, five internal systems (see 4.5 and three external systems (see 4.6) are included in the model. The most interesting findings are those concerning external parties. The model explicitly assumes an interface to ATLAS (see section 4.6.1) for customs handling.

4.5 IT - Identified Internal Systems

These systems have been identified in the informal process model as internal systems. Internal systems are under the control of the enterprise itself.

4.5.1 CAD System

Computer-aided design (CAD) can be defined as the use of computer systems to assist in the creation, modification, analysis or optimization of a design. [LMN08]

4.5.2 CRM System

A System for managing customer contacts is explicitly mentioned - which indicates a Customer Relationship Management System (CRM).

4.5.3 DMS

A Document Management System (DMS) is a database-driven Management System used for archiving and managing electronic documents. [Sof]

4.5.4 Mail Server

A mail server is explicitly mentioned in the model. The mail server in the model is used only used to send invitations for visits to customers.

4.5.5 MRP System

The Material Requirements Planning (MRP) is a computer-based production planning and inventory control system. [Gal15, p.1]

4.6 IT - Identified external systems

These systems have been identified in the Informal Process Model as external systems. External systems are not under the control of the enterprise.

4.6.1 ATLAS

ATLAS (Automatisiertes Tarif- und Lokales Zollabwicklungssystem) is a system provided by the German customs administration. The system allows the submission of electronic declaration to the German Customs Administration. [Deu15]. Declarations that can be processed include Import and Export declarations as well as declarations for inward processing and the customs warehouse procedure.

4.6.2 Bank Interface - Wire Transfer System

The IT landscape must contain some sort of communication mechanism to the bank to receive bank statements digital and to send wire transfer orders.

4.6.3 Customer portal

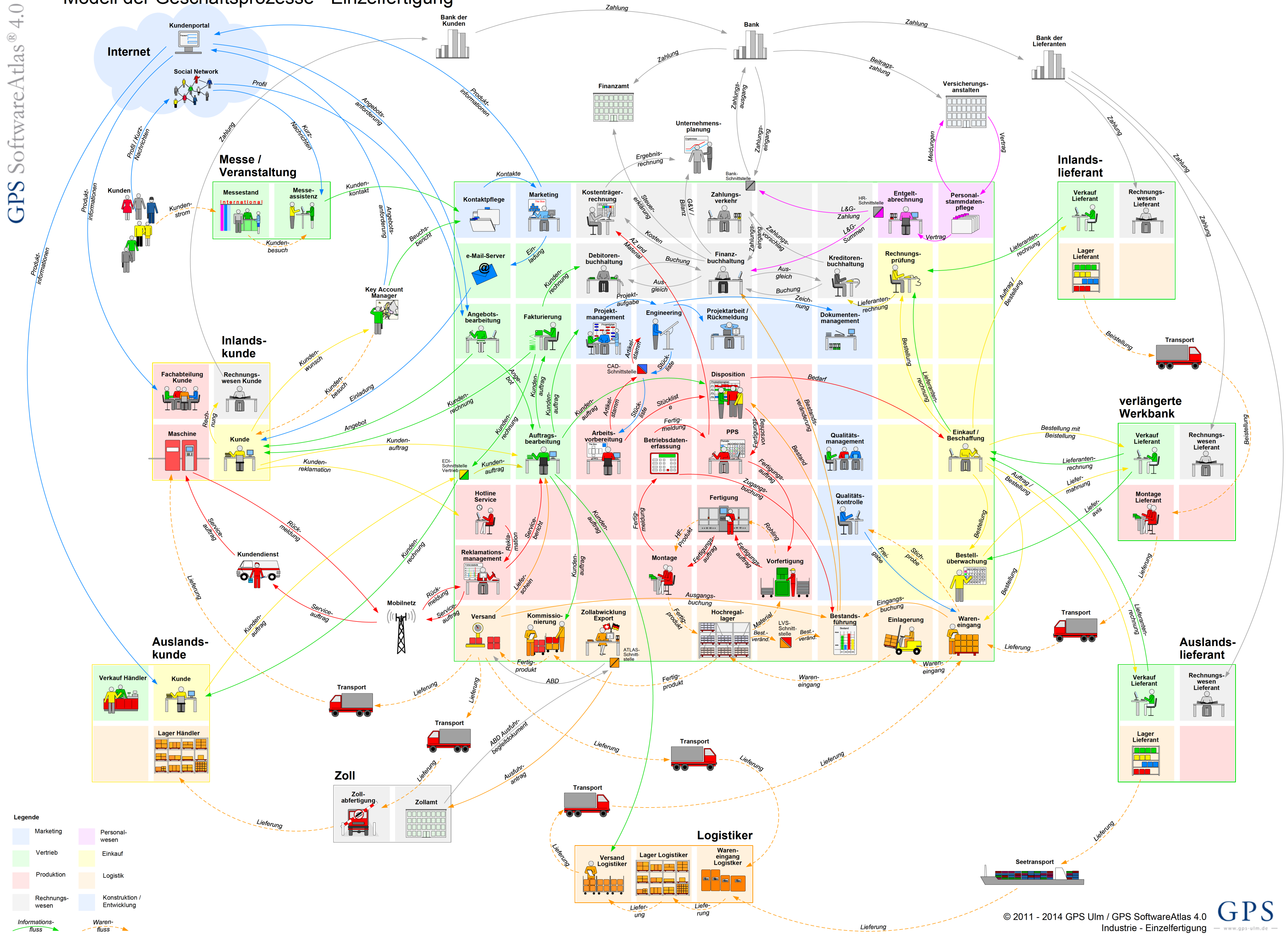
The model mentions a customer portal which customers can use for accessing product information. The portal is modeled outside of the enterprise, which indicates that it is either a portal used by several enterprises or it is hosted outside of the enterprises premises (for instance, with a cloud provider).

4.7 Process Model Evaluation

Starting with positive aspects of the presented model, this model has a focus on mass customization production. It also includes some of the external stakeholders, e.g. customers, suppliers and state authorities. As an overview model, it allows insights into mass-customization production in general. For companies in this field of work, it is a basic model with basic processes, which can be modified, extended and specialized in a more specific diagram depending on the concrete business objectives. Concerning the negative aspects of the model, the processes described are rather informally structured. There is no clear distinction between processes and resources. While the model aims to be a general model for mass-customization production, some assumptions are made depending the structure of the corporation described that are contradictory to processes in the setting of HAWAICycle. For instance, the model assumes that deliveries to foreign customers are made using intermediary traders and goods produced for foreign customers are built-to-stock, while deliveries for domestic customers are delivered directly and without intermediary stocking processes. However, in the case of HAWAICycle, all customers order their bikes themselves since the payments and orders come from the same customer. Of course HAWAICycle is aware of the business customers, but they are only relevant for Marketing and not for the modeling of the Order.
Continuing with the negative aspects - as stated before the model is not highly structured. While there is some notational elements (see Section 4.2), the primary purpose of the model

seems to be a generalized guideline upon which a corporate-specific model can be built. Furthermore, no clear distinction between business units, roles and business objects is made, so the model is ambiguous. Secondly, shared technical resources such as mail servers are attributed to a specific business unit, since there is no possibility in the model to denote shared resources. To create a new, more concise model, objects in the primary model need to be mapped to either business units, roles or business objects. This new information is necessary for creating a non-ambiguous model. Furthermore, since the question of the thesis is a concrete ICT blueprint, neither concrete software products nor software specifications are given in the model. Therefore, to create an ICT blueprint, a specification of requirements is necessary. This specification can be used to select suitable software for HAWAICycle GmbH and its business purpose - custom bicycles. To summarize the evaluation: With regard to HAWAICycle, the model contains interesting aspects but is not readily usable. Some adaption of the model is necessary.

Modell der Geschäftsprozesse - Einzelfertigung



GPS

5 Informal Process Model Conversion and Adaption

The conversion of the informal process model and its transformation is an incremental process. The reason for this approach is that the model first needs some careful analysis in order to identify false assumptions - which will be done in this chapter. The translation process is not described since it consists only of looking up words in dictionaries or help libraries (mainly the SAP Help Library). After this analysis, basic Building Blocks are inserted in Iteraplan and the processes are designed using both Iteraplan for the EA model and Bizagi BPM Modeler for creating BPMN models - Iteraplan does not contain the required capabilities for exhaustive process models. This results in the following steps:

1. Analysis and Description of the model (in chapter 4)
2. Correction of false assumptions in the base model (in this chapter)
3. Translation of Items (with both dictionaries and the SAP Help library)
4. Insertion of Basic Building Blocks (in Iteraplan, described in chapter 6)
5. Process design for selected processes (combination of Building Blocks - in chapter 7)

5.1 False Assumptions in the Base Model

5.1.1 Delivery of Components

In the model, the assumption is being made that domestic deliveries from suppliers are being sent to Contract manufacturers for further assembling. Shipments from foreign suppliers are shipped to the company directly. This assumption is not fully correct.

5.1.2 Finished Goods Handling

The base model assumes that finished Built-To-Order goods are being stored in the warehouse. This is a useful scenario in a small-batch production, but in our specific production environment (custom bicycles) finished products are delivered directly to the customer instead of being warehoused.

5.1.3 Customers

The model assumes that a customer consists of several business units. The model further assumes that domestic customers have machines and foreign customers have a dedicated warehouse. While the differentiation between foreign and domestic customers makes sense the assumptions are not necessarily useful in the given case. First, private customers are not represented in the model but are a major share of all customers. Secondly, in the case of HAWAICycle, foreign customers are similar to domestic customers (apart from taxation differences).

5.1.4 Customer Service

The model assumes that domestic customers give service orders which result in on-site operations for the corporation, which is not the case for HAWAICycle.
The only employees visiting the customers are part of the sales force. The sales forces visits its corporate customers and presents them with new and attractive products developed by HAWAICycle and the Sales Force isn't part of Customer Service but is part of the Marketing department.
The Customer Service of HAWAICycle is restricted to providing Information about the corporations products and configuration options and handling dead-on-arrival-case or giving financial compensation for delayed or damaged orders. Repairs are done by local bike stores or – if the customer has sufficient knowledge – by the customer itself.
If the bike is dead on arrival (e.g. the bike was broken during transport), a return to the plant or a controlled disposal of the bike is a more realistic scenario. If the bike is fully functional on arrival, there are sufficient locations for customers to get a repair. Even if the repair is covered by warranty, a financial compensation for parts and labor is probably the more economic alternative. In any case, HAWAICycle does not have a on-site repair team.

5.2 Aspects not modelled in this Thesis

5.2.1 Trade Fairs

The formal model will not include any reference to trade fairs at this time. While trade fairs certainly are relevant, they can be added later on. The focus at this time is on key elements.

6 Converted Formal Model

As a result of the analysis of the informal process model and the removal of false assumptions, a formal Enterprise Architecture Model has been created for HAWAICycle with Iteraplan. This section describes the model created and the software that has been used for the creation of the EAM model.

6.1 Iteraplan

Iteraplan is a EAM tool developed by the Iteratec, a German software development and consulting company. The tool implements the "Iteratec Best-Practice EA" described in the theory chapter in section 3.1.1. This section briefly describes the system installed, explains the implementation process, highlights problems found for the implementation of business processes and includes an evaluation of the tool.

6.1.1 Tool Information

Iteraplan is available in three different versions.
A **Community Edition** (available via Sourceforge [Sou]) is published with sources under an AGPL license. The community edition was last released in version 3.3 in October 2004.
Starting with October 2014 and Version 3.4, the new additions to the Iteraplan source code are no longer available in an Open Source (AGPL) license. Development continues under a non Open-Source license and Iteratec provides a **Lite Edition** instead of the Community Edition as well as an enterprise edition.[Itea] The Lite Edition and Community Edition are available free of charge, the **Enterprise Edition** is usually available in combination with consulting services.[Mat, p.57]
The Lite Edition unsurprisingly offers less features than the enterprise edition. The most limiting restrictions of the Lite edition is the predetermined database (only an integrated HSQL database) which limits the backup possibilities and the disabled import and export functions. A detailed overview can be found in [Itea]. Particularly because of the disabled import and export functions an Enterprise Edition was used for this thesis. Iteraplan provided both an

unlimited enterprise license for version 3.4.1 as well as a time-restricted enterprise license for version 5 upon request free of charge. Due the restrictions (and to allow later use within the HAWAI project) only the license for version 3.4.1 has been actively used for modeling

6.1.2 Installation

Iteraplan has been installed on a virtual machine provided by MARS (Multi Agent Research and Simulation) Group[1] of the Hamburg University of Applied Sciences. The machine runs on a cluster and has Linux (Ubuntu) installed on its 5 GB partition. Prior to installation a user account with root rights was created on the machine and the files have been copied via SSH and compiled (with Java) using custom settings
The Installation Guide for Iteraplan 3.4.1[Iteb] was used for installation. The following components were installed onto base system:

- Database
  MariaDB Version 10
- Java Version 7
- Web server
  Apache Tomcat Version 7 with custom SSL certificates for access of the web interface. Additionally, the standard ports have been changed to disable non-SSL access
- Iteraplan 3.4.1
  Iteraplan was copied after its compilation to the Tomcat Web Apps Folder

After the installation and start of Tomcat, Iteraplan is available for use with a standard account already created during installation.

6.1.3 Stability of the Environment

Iteraplan has had some stability issues in the environment chosen. The instability can probably be attributed to the small partition. Regrettably, this also caused some data loss and required the repeated input of elements. After unnecessary files were removed (resulting in more free space on the partition) from the hard drive the system stability improved.

[1] http://mars-group.mars.haw-hamburg.de/

6.2 Description of the Formal Model

As a result of the modeling a formal model has been created that is based on a cleaned version of the informal model (see chapter 4).
The original terms of the informal model are included in the description of the formal model. This formal model is included in the appendix (see 9.1) and includes the 186 Building Blocks that have been constructed.
The Building Blocks created bear the names from the elements on the informal model but have been translated for this thesis. Where a name was given in the informal model, this name is included in the column 'Description' at the beginning of the description of the individual Building Block. Since this model is supposed to be used later on, references to external sources such as the SAP Library - that has been used to identify the correct terms - that may provides further insight is also included in the description.
As a result, this model contains all the divisions of the Informal Model (see section 4.3) - these sections have been grouped into Business Domains (for increasing the coherence of the model) and structured into business units (see paragraph Business Landscape Model in section 3.1.1).
Not all 186 Building Blocks are described in detail in this section since the description already covers 30 pages in the appendix. Instead, this section provide an insight into the landscapes that have been implemented.

6.2.1 Business Landscape Model

The Business Landscape Model contains the bike that is created by HAWAICycle as Product as well as ten business domains and 26 business units as well as 25 business processes have been created. These Business Processes use 53 Business Elements. The business domains (green) and business units (violet) are shown as a diagram on the next page.
Representative business processes (especially the Quote-To-Order process) included in the model are modeled in greater detail in the Business Processes section due to the fact that the Iteratec EA allows only a sequential progression of the processes and in reality these processes contain iterative parts.
The Quote-To-Order process shown in Figure 6.1 illustrates this problem.

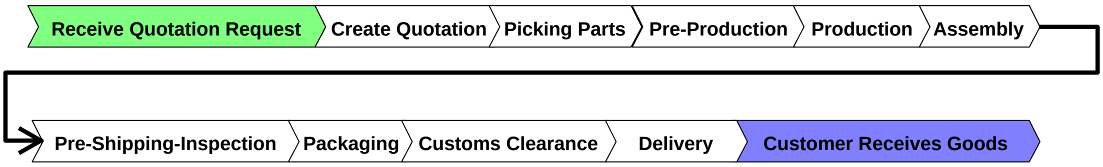
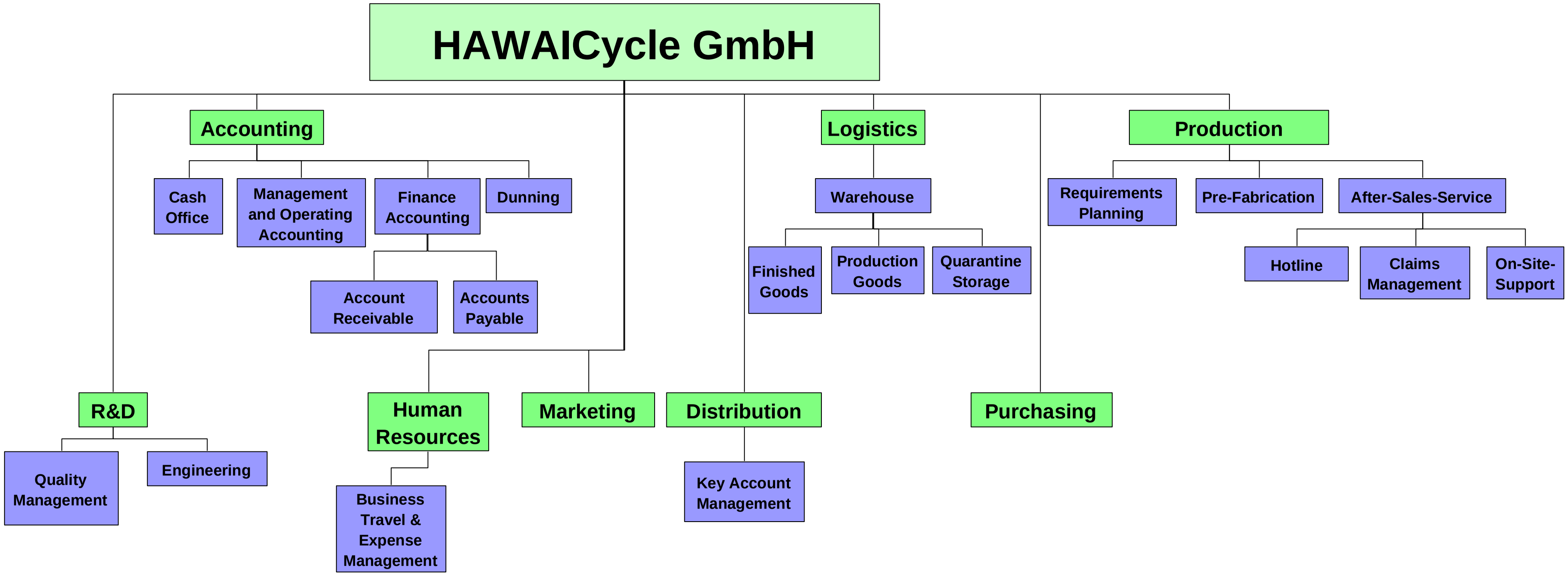


Figure 6.1: Quote-To-Order Process in Iteraplan

Unlike the process shown in Figure 6.1, a quote can and often will be improved after the initial creation because of changes which results in a loop.
The processes modeled contain no defined exit points that allow the termination of the processes. In reality, processes can terminate after the creation of the quote (e.g. because the purchase is not completed by the customer or because the product cannot be produced as requested).
Although this coarse-granular structure is intended by the EA, the focus is on workflows and highlighting risks. [Han09, p.92] states that the process management commonly involves documenting business processes and their interactions with organizational structures, applications and business objects as EPCs (Event Driven Process Chains) using tools such as ARIS[2]. In the case of HAWAICycle, BPMN is used.

[2] ARIS is a concept to enterprise modeling developed by August-Wilhelm Scheer in the 1990's that implements the EPC

Business Domains (green) and Business Units (lilac)



6.2.2 Application Landscape Model and Technical Landscape Model

The **Application Landscape Model** is divided into six *Information System Domains* and 31 *Information Systems*. The systems are connected with 23 Interfaces. They transport the business objects that are created in the Business Landscape Level. The diagram on the next page that has been generated with Iteraplan shows this transportation of business objects. The Interfaces are usually – since no name was shown in the informal model – named after the systems that are joined with the Interface.
The division into different domains was necessary in order to split up the systems into internal systems - that are under the control of HAWAICycle - and external systems under the control of both state authorities (tax authority, customs) or customers.
The specific representation of external systems proved to be necessary as all information flows are directional (from one system to another system) and the modeling therefore required at least a hull element for the external systems.

For instance, the *HRMS ⇆ SSRS Interface* transports Social Security reports (that include the salary and Social Security taxes) from the internal Human Resources Management System to the external Social Security System under the control of the Social Security authorities.

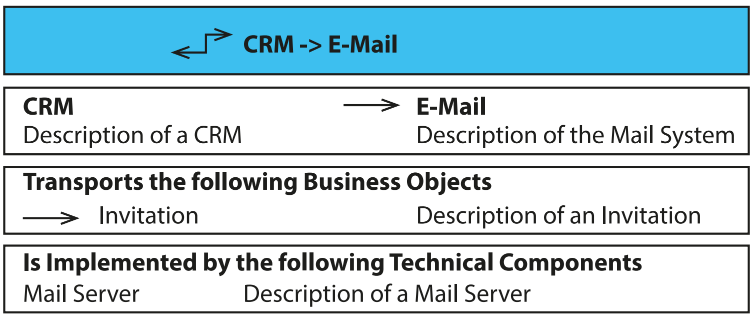


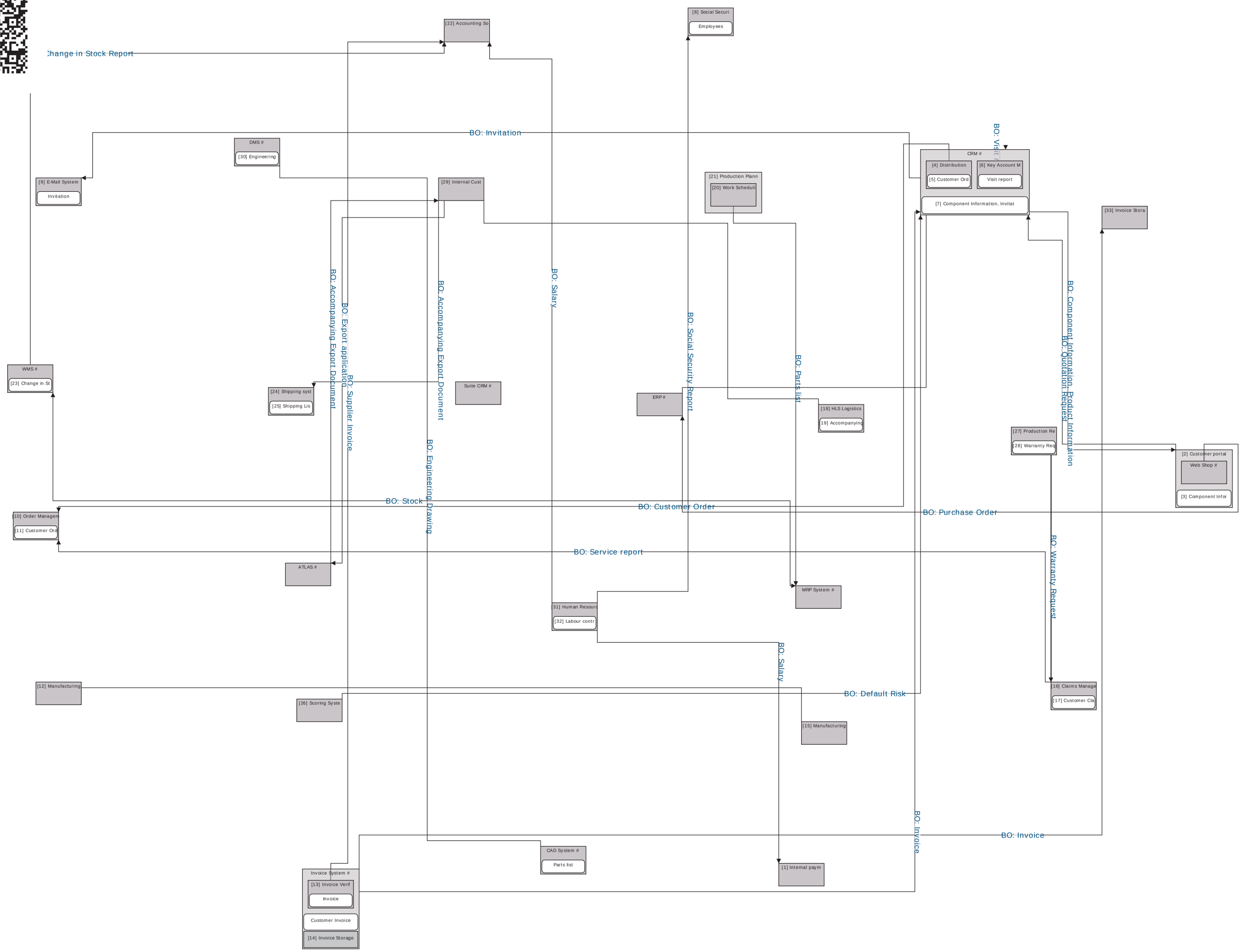
Figure 6.3: Interface in Iteraplan: The Interface between the CRM System and the Mail System

The communication between internal systems also transports business objects. So from a modeling point of view the only difference between internal and external interfaces is that HAWAICycle has only one port under its control. This results in the need to adapt the systems to meet the external needs rather than changing the interface.
The **Technical Landscape Model** is rather small as there exist no systems at the time being. Only two technical components have been modeled for this reason. The only systems that have are included in the informal model are the 'Mail Server' and the 'Web Server'.
The 'Mail Server' is responsible for the technical transportation of E-Mails while the 'Web Server' is responsible for the Hosting of the Customer Portal.

iteraplan Information Flow Diagram

Based on saved query:
InformationFlowAsPNG

Legend	Attribute / Building Block
Colour	n.a.
Line type	n.a.
Line caption	Business Objects



7 Business Processes

As explained earlier (see the formal model description in chapter 6), Iteraplan is not suitable for fine-granular process modeling. Therefore, representative processes have been modeled for this thesis with the Business Process Model Notation (BPMN). For readers unfamiliar with BPMN, the elements and the notation itself are roughly described and a brief overview of the essential visual symbols is given, largely based on the authoritative current standard presented by the Object Management Group (OMG).[1]

7.1 Business Process Model and Notation (BPMN)

The processes in this section are modeled using the Business Process Model and Notation (BPMN).
BPMN is a graphical notation for drawing business processes proposed by the OMG. BPMN has been published first as Version 1.1 in January 2008. This introduction is based on the latest version published, BPMN 2.0.1, which has been standardized as ISO/IEC 19510:2013[Obj13] in July 2013. BPMN aims to bridge the communication gap that frequently occurs between business process design and implementation. The language is similar to other informal notations such as UML activity diagrams and extended event-driven process chains.[LO09, p.294]. A model expressed in terms of BPMN is called a Business Process Diagram (BPD).[LO09, p.295]. Since Version 2, a Business Process Diagram consists of five basic categories of elements: [Obj11]

1. Flow Objects
2. Data
3. Connecting Objects
4. Swimlanes and

[1] Founded in 1989, The Object Management Group is an international, open membership, not-for-profit technology standards consortium. See http://www.omg.org for further information.

5. Artifacts.

In the publication of the original specification [Obj08], data was considered an artifact and not a main part of the process flow. While the data flows are still separated from the Sequence flows, data is upgraded to a first class element in BPMN 2.[SWB+12, p.25] Therefore, literature referring to older BPMN standards often omit data as an element [LO09, p.295].

7.1.1 Flow Objects

Flow Objects are the main graphical elements to define the behavior of a Business Process.[Obj11, p.25] All the graphical representations for all elements discussed are included on an extra page later in this section. The three flow objects are

1. **Events**
   An Event is something that “happens” during the course of a Process [...]. [Obj11, p.31] Three event types exist, Start Event, Intermediate Event and End Event.
   a) **Start Event**
      As the name implies, the Start Event indicates that where a particular process will start. [Obj11, p.31]
   b) **Intermediate Event**
      An Intermediate Event occurs between a Start Event and an End Event. This type of Event will affect the flow of the process. [Obj11, p.31]
   c) **End Event**
      End Events terminate the process.[Obj11, p.31]
2. **Activity**
   A generic term for work that the company performs [Obj11, p.32]
3. **Gateway**
   They are used to control divergence and convergence of the flow (Split and Merge) [Biza] and will determine branching, forking, merging and joining of paths. [Obj11, p.34] Gateways are represented by a diamond-shaped tag.
   a) An **Exclusive Gateway**
      is used for Decisions and Merging. [Obj11, p.34] Logically, it represents an XOR-Decision.[FR14, p.30] Both Exclusive and Event-Based Gateways perform exclusive decisions. [Obj11, p.34]. An Exclusive Gateway is represented by either a diamond-shaped tag or by a diamond shaped tags with an X inside the diamond.

b) **Event-Based Gateway**
Event-Based and Parallel Event-Based Gateways can start a new instance of the Process. [Obj11, p.34]

c) An **Inclusive Gateway**
can be used to create alternative but also parallel paths within a Process flow. Unlike the Exclusive Gateway. all condition expressions are evaluated. The *true* evaluation of one condition Expression does not exclude the evaluation of other condition Expressions.[Obj11, p. 294] All paths are considered independent, so all paths may be taken. By design, at least one path should be taken. Logically, it represents an OR-Switch.[FR14, p.36]

d) A **Parallel Gateway**
is used to synchronize parallel flows and to create parallel flows.[Obj11, p.293]. A parallel Gateway creates parallel paths without checking any conditions. For incoming flows, the Parallel Gateway waits for all incoming flows. BPMN uses a token-based synchronization mechanism to ensure that all flows have been received in the in the incoming gateway.[Obj11, p.293] The Parallel Gateway MUST use a marker that is in the shape of a plus sign and is placed within the Gateway diamond. to distinguish it from other Gateways. [Obj11, p.293]. Logically, this gateway represents an AND-Switch.

e) **Complex Gateway**
The Complex Gateway can be used to synchronize complex synchronization behavior, such as N-out-of-M-decisions. The Complex Gateway MUST use a marker that is in the shape of an asterisk and is placed within the Gateway diamond to distinguish it from other gateways. [Obj11, p.295]

7.1.2 Data Objects

Data is represented with four elements[Obj11, p.26] The four Data elements are:

1. **Data Object**
A Data Object provides information about what Activities require to be performed and/or what they produce (see page 205), Data Objects can represent a singular object or a collection of objects. Data Input and Data Output provide the same information for Processes. [Obj11, p.36]

2. **Data Inputs**
   Data Input is the data that is required to execute an activity or a process. [Obj11, p.211]

3. **Data Outputs**
   Data Output is the data that is produced by an Activity or a Process. [Obj11, p.211]

4. **Data Stores**
   A Data Store provides a mechanism for Activities to retrieve or update stored information that will persist beyond the scope of the process. [Obj11, p.208]

7.1.3 Connecting Objects

There are four ways of connecting the flow elements to each other or to other information. *Data* elements[Obj11, p.26] [SAW04] These four Connecting Objects are:

1. **Sequence Flows**
   A Sequence Flow is used to show the order that activities will be performed in a Process [Obj11, p.29]

2. **Message Flows**
   A Message Flow is used to show the flow of messages between two participants [Obj11, p.29]

3. **Associations**
   An Association is used to link information and artifacts with BPMN graphical elements [Obj11, p.29]

4. **Data Associations**
   Data Associations are used to move data between Data Objects, Properties, and inputs and outputs of Activities, Processes, and Global Tasks [Bro]

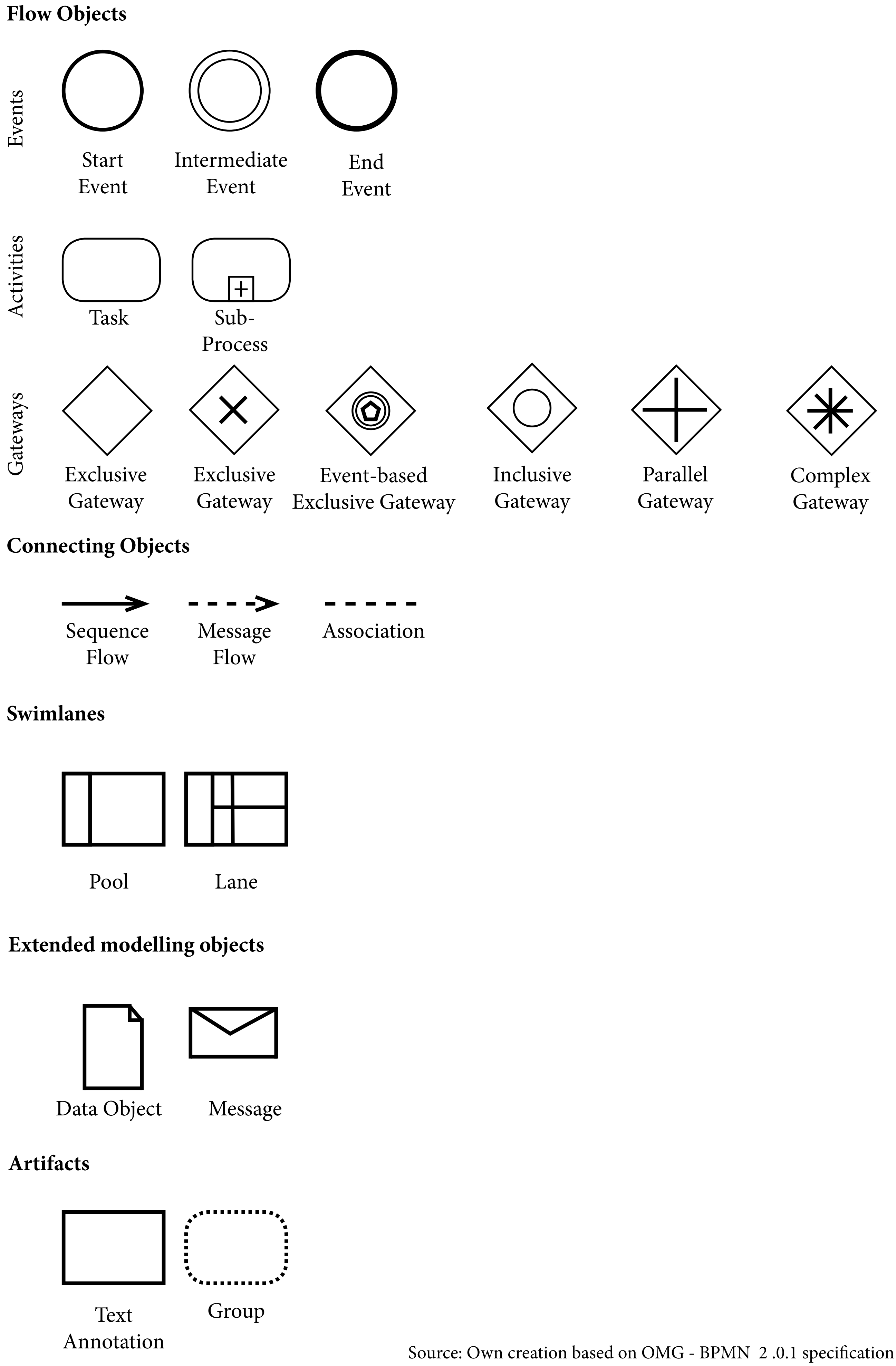
7.1.4 Artifacts

Artifacts are used to provide additional information about the Process. There are two standardized Artifacts, but modelers or modeling tools are free to add as many Artifacts as necessary.[Obj11, p.26]. This functionality is implemented in a number of modeling tools, e.g. in Bizagi Process Modeler [Bizb] or in Oracle JDeveloper [Obj]. Standardized Artifacts include:

1. **Text Annotation**
   Text Annotations are a mechanism for a modeler to provide additional text information for the reader of a BPMN Diagram [Obj11, p.40]

2. **Group**
   A Group is a grouping of graphical elements that are within the same Category [...]. This type of grouping does not affect the Sequence Flows within the Group. The Category name appears on the diagram as the group label. [...] Groups are one way in which Categories of objects can be visually displayed on the diagram. [Obj11, p.41]

Essential BPMN Elements



Source: Own creation based on OMG - BPMN 2 .0.1 specification

7.2 BPMN Processes

The modeling of processes using BPMN is possible with a number of tools. This thesis uses Bizagi Modeler, a software published and provided free-of-charge by Bizagi. Compared to other BPMN modeling tools, Bizagi Modeler includes a model checker that allows to check the syntax of the models and allows to simulate the workflow [Bizc] - this simulation has been performed for the models in the following sections. The processes modeled in this section are representative and are modeled as a starting point to give input for the actual implementation. The processes do not represent the range of all possible processes but have been selected due to their importance.

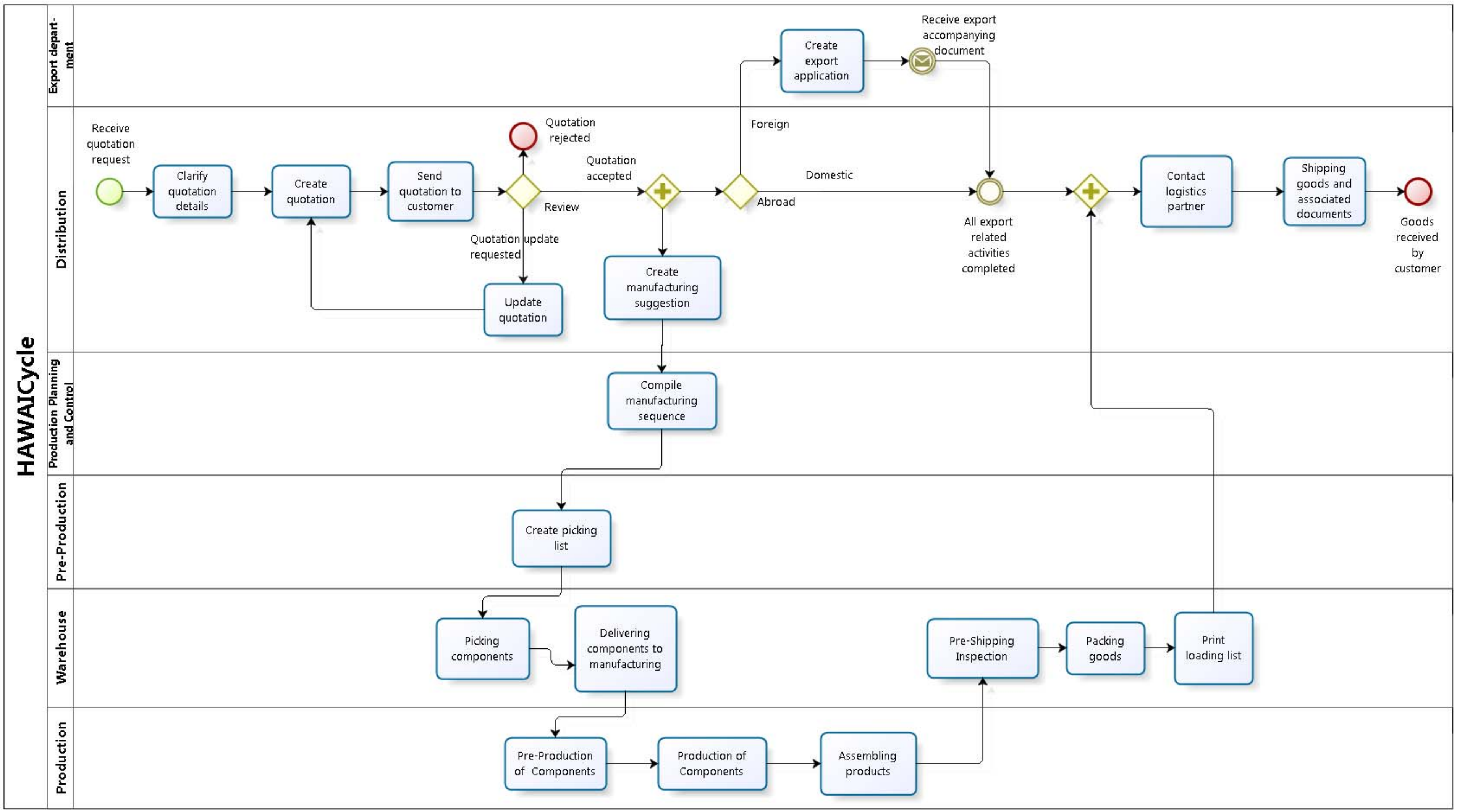
7.2.1 Built-To-Order

The Built-To-Order process is modeled after both the informal model and information published in [SAP].
This process is initiated by a customer of HAWAICycle sending a quotation request. This process has already been mentioned in the formal model and covers the complete process from the moment a customer sends a request to the moment the order is delivered and these orders create value. Therefore, it is the central process for HAWAICycle.
After the request has been received, a binding quotation is being sent to the customer. The customer can afterwards request a change of the quotation. When the customer accepts the quotation and pays the invoice , the bike is produced and shipped to the customer. The first version of the model includes a decision point based on the location of the customer - only shipments to foreign customers require export documents.
This is also the central difference between this process and the process that has been created in Iteraplan itself, since this process contains loops and decision points and is a more realistic representation of the actual process. It is also the first process that will be implemented in the actual HAWAICycle infrastructure. The initial process to be implemented differs from the model process - it is simpler and includes a new aspect (payment) and is therefore also included in this thesis.

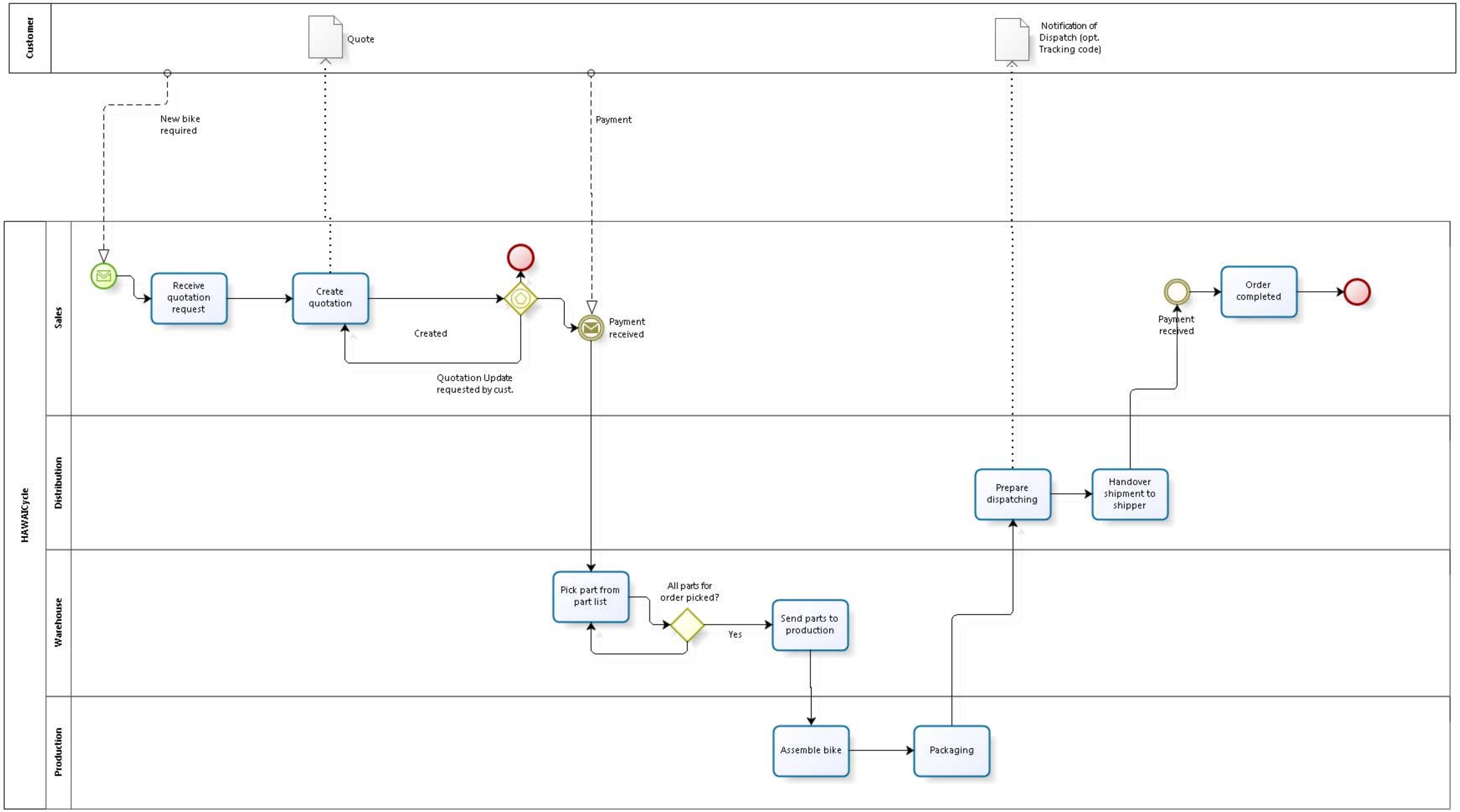
7.2.2 Goods Required

In this process, the *Requirements Planning* department (compare to the formal model in section 6.2.1) determines that goods need to ordered. This process is central since the production at HAWAICycle relies heavily on the import of goods, as it has been explained in chapter 2. To

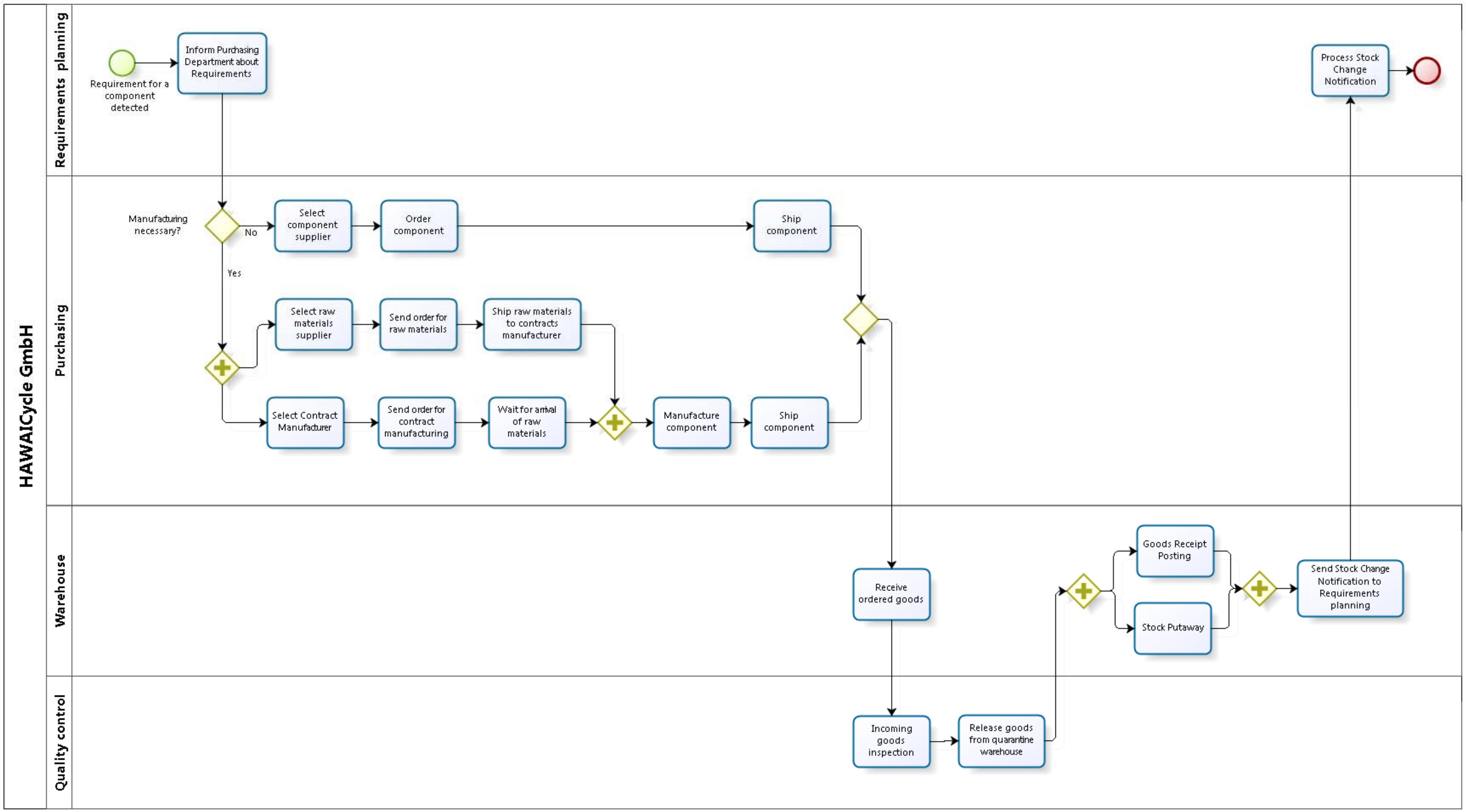
allow prompt fulfillment of orders in the Built-To-Order Process, all relevant parts must be available. Due to long shipping times from South-East-Asia – as explained in chapter 2, bike parts mostly originate there – substantial lead times need to be considered for ordering bike parts.
Example: The transit time (without production and order factored in) from Shenzhen, China to Hamburg, Germany is between 49 and 57 days in November 2015 according to Searates.com[Sea]. These long lead times require careful planning in advance to ensure the business success of HAWAICycle, which is supported by this process.



Quote-To-Order Process



Quote-To-Order (Initial Process For HAWAI



Goods Required Process

8 Findings, Conclusions and Recommendations

This thesis has successfully implemented an ICT blueprint for HAWAICycle GmbH, a fictional mid-size Built-To-Order corporation that currently got no infrastructure.
HAWAICycle has been described in greater detail during this thesis. This analysis results in finding that HAWAICycle uses a mass customization approach to produce bikes. Mass customization production has been inspected further which has resulted in a ratio-based classification approach for HAWAICycle, since HAWAICycle has used a mix between two different types of mass customization - Built-To-Order and Made-To-Order. As a result of the classification, the production of bikes in HAWAICycle is classified to be mostly Built-To-Order mass customization. For this type of mass customization, the inbound logistic processes are highly relevant.
For the creation of the ICT blueprint, an informal model provided for the thesis was used as a template for the implementation of the ICT infrastructure. This model was a good source of inspiration although the analysis of the informal model resulted in the finding that it claimed to be a very general model. However, this claim was contradicted as the thesis shows that the informal model makes several assumptions without clearly stating them in the model. This makes the model less general that it claims to be. Furthermore, the model is ambiguous and not precise enough for a direct implementation and requires a lot of interpretation.
For all this reasons it was necessary to interpret the model. For the interpretation, it would have been useful if the model had been documented properly. Regrettably, the model lacks documentation. This made the interpretation more difficult.
The interpretation showed that the model was also incorrect in some aspects which resulted in the need for a clean up prior to the conversion to a more formal model adapted to the HAWAICycle corporation.
After the clean up, all the systems found in the informal model were converted into an EA model. Since the EA model used does not allow the implementation of realistic business processes, relevant business processes were modeled as well using a BPMN tool to complement the ICT blueprint.

Since the informal model is not tailored to a bike factory, the formal model that was created might not accurately reflect a realistic bike factory - so one result of this thesis is that using an actual bike factory would probably yield more realistic results and thus a better blueprint.
The tool used for the implementation of the Enterprise Architecture - Iteraplan - has proven valuable. The instrument has a good web based user interface and allows a quick start of the EA implementation. The software is likely to be very useful in actual enterprise context.
The usefulness of Iteraplan was impaired by some stability problems and some of the data inserted in the model was lost due to crashes. This is likely to be correlated to the small hard drive but is still a chilling factor that needs to be fixed.
Concerning the HAWAI project, it is regrettable that Iteraplan is no longer open source. If it still were open source, this would allow to change some aspects in the modeling tool, which would create in a more powerful tool with regard to a scientific context. Also, it is always possible that Iteratec - the company behind Iteraplan - ceases operation which – if Iteraplan were used – would impair the HAWAI project due to a dependency to an aging tool. Therefore the use of Iteraplan needs to be evaluated further prior to a final decision.
It has also been shown that Iteraplan is not an adequate tool for advanced process modeling. Although the tool was never intended for this task by the developers this function would have increased the value of the software significantly. Regrettably, it is also not possible to import the basic process models exported from Iteraplan into any other software than Iteraplan itself since the file does not adhere to the standards proclaimed.
It is possible that a different EA model – such as TOGAF – results in a more efficient solution since the EA model used ultimately influenced the findings of this thesis. However, using Iteraplan resulted in swift initial results and thus it was a good choice for this thesis.
This thesis has also proved that the choice of an Enterprise Architecture Framework is not enough for the implementation of an ICT blueprint. Additionally, the integration of systems needs to be planned and a selection of an integration framework has become necessary.
Two different integration frameworks that are currently used in the enterprise context have been reviewed. The comparison between SOA and EAI showed that SOA is the better integration framework for HAWAICycle. It allows better maintainability and is also better suited for monitoring business processes.
SOA allows different implementation styles for the service interfaces. Two different service interfaces – WS* and REST – have been reviewed. The comparison of REST and WS* revealed that REST requires less formal knowledge and less tools to dive into the implementation and change it. In effect, REST is the better service interface for HAWAICycle because it requires less technical skills and is therefore easier to implement and change. This is relevant since one of

the target groups of the HAWAI project are undergraduate students. When the implementation is completed, they are supposed to change the environment. Especially for them, REST is the better choice. However, it is likely that other users won't object to technology that doesn't require a very steep learning curve. This makes SOA with REST interfaces the best choice for the HAWAI project.
For both SOA and EAI a middleware should be used, for SOA the Enterprise Service Bus has been reviewed and found suitable.
This results in an ICT blueprint using a service-oriented architecture with the Enterprise Service Bus for the communication of the individual services.

To sum up, an informal model has been provided, adapted and refined to create a formal model that has been implemented in Iteraplan. This model will be implemented and adapted further to be used in the context of the HAWAI project. The design of the model is reflecting the future use of the model.

8.1 Recommendations for Further Studies

The conclusion and findings section already indicated that the model might not reproduce a realistic scenario. For this reason, it would be interesting to check this model with a real corporation and re-adapt it afterwards.
However, to find a large corporation that allows a deep inspection and public documentation of its IT infrastructure is probably difficult. A feasible strategy might therefore be to offer a full-fledged analysis of the business operations to small local bike production store. Normally such small firms cannot afford expensive consulting services and thus this opportunity holds significant value to them. Assuming that the results can be transformed from a small enterprise to HAWAICycle this can be a mutually beneficial arrangement.
Since large corporations might object to a deep inspection and the transformation from a small company might be difficult and could result in deviations due to the projection it might be easier to create a corporation that implements the processes and change the company based on the findings. This strategy is already successfully used by a wine store created and owned by McKinsey for training purposes (see [Sch13]).
Apart from a somewhat vague scenario, in this thesis no applications have actually been selected. The selection of actual applications most likely results in a change of the relatively theoretical model. The selection of real applications also bears the need to implement interfaces and thus would allow to research the interface definition further.
Some aspects have not been modeled on purpose due to focus. Trade Fairs is one aspect that

could be modeled based upon this thesis, since it was left out intentionally. Alternatively and additionally, using a different reference model such as eTOM (for the telecommunications sector) would permit to create a different blueprint and therefore would allow to choose from a variety of different blueprints in the lab.
Also, it is possible to re-implement the ICT blueprint with a different EA tool, such as ABACUS or the *Rational Software Architect* since the implementation with a different tool – sebis (Software Engineering for Business Information Systems) Institute at TU München compiled a list of different tools [seb] – might yield new findings.
One aspect especially interesting for realistic business processes and for IT processes is the modeling of capacity restrictions. In reality, corporations have capacity restrictions that cannot be removed immediately. These capacity restrictions apply both to the business processes (e.g. the corporation can only handle a limited number of orders in a given time frame) and to the IT (e.g. a computer has a maximum output) These capacity restrictions can be modeled both statically and dynamically using a variety of approaches, mapped to business processes and visualized. This would allow the management to follow the execution of processes and would make the identification of challenges (e.g. shortages or increasing processing times in the plant) easier. The model presented in this thesis had been successfully developed under the given premises. Nonetheless there are further processes possible for future developments which might give other valuable insights and show further possibilities.

9 Appendix

9.1 Formal Model

9.1.1 Business Landscape Model in Iteraplan

The following pages contain the Building Blocks created in Iteraplan that belong to the Business Landscape Model.

Products

Product	Hierarchical Name	Description
Bike		null
Components	Bike : Components	The bike is assembled using components

Business Domains

Business Domain	Hierarchical Name	Description
HAWAI Bikes GmbH		Our company. We produce Bikes
Accounting	HAWAI Bikes GmbH : Accounting	(Rechnungswesen)
Distribution	HAWAI Bikes GmbH : Distribution	(Vertrieb)
Human Resources	HAWAI Bikes GmbH : Human Resources	(Personalwesen)
Logistics	HAWAI Bikes GmbH : Logistics	(Logistik)
Marketing	HAWAI Bikes GmbH : Marketing	null
Production	HAWAI Bikes GmbH : Production	(Produktion) Production creates goods
Purchasing	HAWAI Bikes GmbH : Purchasing	(Einkauf)
R&D	HAWAI Bikes GmbH : R&D	(Konstruktion / Entwicklung)

Business Units

Business Unit	Hierarchical Name	Description
Accounting		The Accounting Department
After-Sales-Service		The customer supports assist customers by solving problems that occur after sales
Human Resources		The Human Resources Department manages the Human Capital of the Company
Distribution		(Vertrieb) Distributes our articles
Engineering		(Engineering) Develops and improves products
Warehouse		The warehouse stores products and goods
Production		The production produces the goods, e.g. the bikes
Marketing		
Purchasing		(Einkauf) Purchasing orders goods after a demand for goods arises
Quality management		
Requirements Planning		(Disposition) Requirements planning involves ensuring that goods are available when recipients (stores or customers) and consumers require them. The quantities required have to be procured in good time. The following activities are required: - see http://help.sap.com/saphelp_46c/helpdata/en/12/085211470311d1894a0000e8323352/content.htm
Finance Accounting	Accounting : Finance Accounting	Finance Accounting manages and records all financial transactions and prepares financial statements
Accounts Payable	Accounting : Finance Accounting : Accounts Payable	Accounts payable
Accounts Receivable	Accounting : Finance Accounting : Accounts Receivable	Accounts receivable
Cash Office	Accounting : Cash Office	Management of Cash Funds
Dunning	Accounting : Dunning	Reminds customers about outstanding dues and initiates required legal actions

Management and Operating accounting	Accounting : Management and Operating accounting	(Controlling) Management and Operating accounting is about collecting information to support the decision process and allows better steering of the company
Claims Management	After-Sales-Service : Claims Management	(Reklamationsmanagement)
Hotline	After-Sales-Service : Hotline	(Hotline Service) The hotline supports the customer remotely
On-Site-Support	After-Sales-Service : On-Site-Support	(Kundendienst) On-Site-Support solves problems a customer has on-site
Key Account Management	Distribution : Key Account Management	Key Account Management handles requests that are deemed strategically important for the company - see http://www.wirtschaftslexikon24.com/d/key-account-management/key-account-management.htm
Business Travel & Expense Management	Human Resources : Business Travel & Expense Management	(Reisekostenabrechnung) Organizes business travels and associated costs
Pre-Fabrication	Production : Pre-Fabrication	Pre-Production orders the warehouse to remove parts which are required for the production of the bike from stock - see http://help.sap.com/saphelp_erp60_sp/helpdata/en/c6/f8402b4afa11d182b90000e829fbfe/content.htm
Finished Goods Warehouse	Warehouse : Finished Goods Warehouse	Contains the Finished Goods
Incoming goods - Quarantine storage	Warehouse : Incoming goods - Quarantine storage	Incoming goods are temporarily stored in quarantine storage until controls are performed
Production Goods Warehouse	Warehouse : Production Goods Warehouse	Contains The Production Goods

Business Processes

Business Process	Hierarchical Name	Description
Creating engineering drawing		During this process, the engineering department creates a new engineering drawing
Quote-to-order		The complete process after the first quotation request to the shipment of goods http://help.sap.com/saphelp_46c/helpdata/en/e0/10a837a703934de10000009b38f8cf/content.htm
Finished Goods Report		Construction sends reports about finished goods to Manufacturing Execution
Quality Control		(Qualitätskontrolle)
Invoicing		(Fakturierung) During invoice, an invoice is generated and sent to a customer. Afterwards, the invoice is being sent to the customer and to "Account payable"
Physical inventory		(Inventur) During physical inventory, objects stored in the warehouse are counted and compared to the records.
Invoice verification		(Rechnungsprüfung) The invoice verification checks incoming invoices for legitibilty
Goods Receipt		(Wareneingang) Incoming goods are being handled
Waste handling		Physical objects which inherit no value to the company are disposed by waste handling
Credit limit check		The Credit limit of a customer is being checked to prevent negative financial impacts
Goods Receipt Posting	Goods Receipt : Goods Receipt Posting	(Eingangsbuchung) The arrived goods are being posted to the WMS http://help.sap.com/saphelp_erp60_sp/helpdata/en/96/f7b05f888111d3b4c2006094b9b9dd/content.htm

Incoming Goods Inspection	Goods Receipt : Incoming Goods Inspection	During the inspection of incoming goods a quantity and quality check is performed. http://help.sap.com/saphelp_46c/helpdata/en/51/953caea1fa11d189ba0000e829fbbd/content.htm
Warehousing	Goods Receipt : Warehousing	After inspection, incoming goods are being warehoused
Create quotation	Quote-to-order : Create quotation	(Kundenauftrag) After a customer contacts the company with a request for a quotation, a quotation is being created
Customer receives goods	Quote-to-order : Customer receives goods	Customer receives goods
Customs clearance	Quote-to-order : Customs clearance	(Zollabwicklung) If the shipment is going to a foreign customer, shipment is being cleared by customers prior to leaving the country
Delivery of goods	Quote-to-order : Delivery of goods	After the pre-shipping inspection, the order is being delivered to the customer
Packaging	Quote-to-order : Packaging	(Verpackung) Finished goods are package before shipment
Picking	Quote-to-order : Picking	(Kommissionierung) During picking, the articles are being collected from warehouse and prepared for shipment
Pre-shipping inspection	Quote-to-order : Pre-shipping inspection	(Versand) Prior to shipping, the correctness of the order is being assured
Production process	Quote-to-order : Production process	(Fertigung)
Assembly	Quote-to-order : Production process : Assembly	The fitting together of manufactured parts into a complete [...] structure
Pre-Production	Quote-to-order : Production process : Pre-Production	(Vorfertigung) During pre-production, components are collected from the warehouse
Production	Quote-to-order : Production process : Production	(Fertigung) During production, raw materials are converted into intermediate goods
Receive quotation request	Quote-to-order : Receive quotation request	We receive a quotation request by a customer

Business Functions

Business Function	Hierarchical Name	Description
HR Master Data Maintainence		(Personalstammdatenpflege)
Inventory management		See http://www.wirtschaftslexikon24.com/d/bestandsfuehrung/bestandsfuehrung.htm and http://help.sap.com/saphelp_erp60_sp/helpdata/en/68/2eb853dcfcb44ce10000000a174cb4/frameset.htm
Payroll Run		(Entgeltabrechnung) Executing Payroll Run is done once a month. http://help.sap.de/saphelp_46c/helpdata/en/fd/45a2cc9d6411d189b60000e829fbbd/content.htm

Business Objects

Business Object	Hierarchical Name	Description
Accompanying Export Document		(Ausfuhrbegleitdokument) This document is sent alongside goods and documents the export permission
Article Base List		(Artikelstamm) The list of all articles produced by our company
Authorities		State / Legal authorities
Banks		Banks are handling financial transactions
Business Partners		Business Partners share an association with the company
Contact		A contact is a legal or natural entity
Credit reporting and scoring bureau		(Wirtschaftsauskunft) A credit reporting and scoring bureau evalutes potential and/or current business partners based on publicly available or otherwise retrievable information.
Customer		The customer purchases goods from the company, e.g. bikes
Customer Claim		(Kundenreklamation) A request from a customer to solve a problem
Customer Order		An order from a customer to us
Customer quotation		(Angebot) A quotation requested by a customer
Customer Request		A request from a customer
Default Risk		The risk that a customer is unable to pay outstanding dues
Delivery reminder		(Liefermahnung) A delivery reminder is sent to the supplier if goods are overdue
Dispatch advise		(Lieferavis) A dispatch advise is sent from a supplier to us if goods are ready to be delivered and an approximate arrival date is available
Employees		Employees work for the company

Employment Contract		(Arbeitsvertrag) An employment contract or contract of employment is a kind of contract used in labour law to attribute rights and responsibilities between parties to a bargain.
Engineering Drawing		An engineering drawing, a type of technical drawing, is used to fully and clearly define requirements for engineered items.
Export application		(Ausfuhrantrag) The export application is being sent to customs.
Insurance Claim		(Personalstammdatenpflege --> Meldungen --> Versicherungsanstalten)
Invitation		(Einladung) An invitation sent from marketing to a domestic customer
Invoice		An invoice is a legal document and documents that an amount is due for goods/services rendered
Labour contract		([Arbeits-]vertrag) A contract with an employee about the terms of employment
Loading list		(Lieferschein) The loading list is a list of all products shipped in a delivery
Logistics		Logistics handles incoming and outgoing shipments of goods rendered and products sold
Parts list		(Stückliste) A list of parts for a specific bike
Payment Proposal		(Zahlungsvorschlag) A proposal to pay an outstanding due
Product Information		(Produktinformationen) Product Informations are Informations that those informations relevant to the specified target audience
Report		A report provides a stakeholder with information
Salary		(Lohn & Gehalt) The salary is the compensation a worker or employee receives as a compensation for his/her work & time
Shipping List		(Lieferschein) A list of parts sent with a shipment
Social Security Report		(Sozialversicherungsmeldungen) A variety of reports sent to Social Security as required by law.

Stock		(Bestand)
Vendor		(Lieferant) The Vendor sells goods to the company
Customs Authority	Authorities : Customs Authority	The customs department collecties duties for shipments from/to abroad
Taxation Authority	Authorities : Taxation Authority	The taxation authority/revenue service collects taxes and fees owed to the state.
Insurance Companies	Business Partners : Insurance Companies	Insurance Companies are used to mitigate risks that occur during actions by the company
Domestic Customer	Customer : Domestic Customer	(Inlandskunde) Customer located within the same country
Foreign Customer	Customer : Foreign Customer	(Auslandskunde)
Purchase Order	Customer Order : Purchase Order	(Bestellung) A purchase order from a customer
Electronic purchase order	Customer Order : Purchase Order : Electronic purchase order	A purchase order submitted using the EDI interface from distribution
Service Order	Customer Order : Service Order	(Serviceauftrag) The service order is being sent to the on-site-service from the complain management
Quotation Request	Customer Request : Quotation Request	(Angebotsanfrage) The customer send a request for a quotation to the company
Warranty Request	Customer Request : Warranty Request	(Reklamation) A warranty request is sent from a customer if a promise is broken or the goods are not as expected
Customer Invoice	Invoice : Customer Invoice	([Kunden-]Rechnung) A customer invoice is sent to a customer and is a request to pay an amount due
Supplier Invoice	Invoice : Supplier Invoice	(Lieferantenrechnung) A supplier invoice is an invoice sent to us from a supplier
Component Information	Product Information : Component Information	Component Informations are informations about the single components of the product
Change in Stock Report	Report : Change in Stock Report	(Bestandsveränderung) A change in stock is being reported to
Service report	Report : Service report	(Servicebericht) A service report is being generated after a service visit at a customer
Visit report	Report : Visit report	(Besuchsbericht) A visit report is being generated by the Key Account Manager after visiting a customer

Accounting	Vendor : Accounting	(Buchhaltung Lieferant) The Accounting Department from The Vendor
Sales	Vendor : Sales	The Sales Dept. from the Vendor sells goods to the company

9.1.2 Application Landscape Model in Iteraplan

The following pages contain the Building Blocks created in Iteraplan that belong to the Application Landscape Model.

Elements of the

Application Landscape Model

Information System Domain

Information System Domain	Hierarchical Name	Description
Customer Internal IT-Systems	External IT-Systems : Customer Internal IT-Systems	IT Systems under the supervision of customers
Customs Authority IT-Systems	External IT-Systems : Customs Authority IT-Systems	IT-Systems under the supervision of the customs authority
External IT-Systems		IT-Systems outside of our control
Internal Systems		IT Systems within our control
Social Security IT-Systems	External IT-Systems : Social Security IT-Systems	IT-Systems under the supervision of social security authorities - specifically Pension & Public Health Insurance
Tax Authority IT-Systems	External IT-Systems : Tax Authority IT-Systems	IT-Systems under the supervision of the Tax Authority

Information Systems

Information System	Hierarchical Name	Description	Status
Accounting Software		(Finanzbuchhaltungssoftware)	Current
ATLAS		(ATLAS) Automated Tariff- and local customs handling system	Current
CAD System		Computer-aided design (CAD) can be defined as the use of computer systems to assist in the creation, modification, analysis or optimization of a design.	Current
Claims Management System		(Kundenreklamation) A system for handling customer complaints	Current
CRM		(Kontaktpflege) A Customer Relationship Management System	Inactive
Customer portal		(Internet - Kundenportal) A portal for our customers for requesting quotations	Current
DMS		A document management system	Planned
E-Mail System		A system for sending mails	Current
ERP		null	Current
HLS Logistics System		HAW Logistics System	Current
Human Resource Management System		(HR)	Current
Internal Custom Handling System		The internal system handling communication with ATLAS	Current
Internal payment transaction system		The internal payment transaction system is used to send payment orders to the bank using the banks interface	Current
Invoice Storage System (Excel)			Current
Invoice System			Current
Manufacturing Execution System (MES)		(Betriebsdatenerfassung) Receives Information from the production about finished products Sends information about finished products to the manufacturing management system	Current

Manufacturing Management System		(PPS) A system for scheduling the production	Current
MRP System		(Dispositionssystem) Material Requirements Planning (MRP) is a computer-based production planning and inventory control system. MRP is concerned with both production scheduling and inventory control. It is a material control system that attempts to keep adequate inventory levels to assure that required materials are available when needed. MRP is applicable in situations of multiple items with complex bills of materials. Source: http://www.columbia.edu/~gmg2/4000/pdf/lect_06.pdf	Current
Order Management System		(Auftragsbearbeitung) Supports the quote-to-order process	Current
Production Planning System			Inactive
Production Report System (Customer)		Customer sends reports from machines / production to our Claims department	Current
Scoring System		The scoring system evaluates the customers credit worthyness and responds with a recommendation.	Current
Shipping system		The shipping system is being used to prepare shipping, e.g. to contact transport companies and to collect all informations required for shipping	Current
Social Security Report System		The Social Security Report System handles reports from the employee about employees and uses the reports to handle mandatory insurances (e.g. pension, health, unemployment,...)	Current
Suite CRM		Actual CRM Implementation https://suitecrm.com/	Current

WMS		(LVS) A warehouse management system	Current
Distribution EDI system	CRM : Distribution EDI system	(EDI-Schnittstelle Vertrieb) The EDI system has at least an interface for receiving/sending EDIFACT messages	Current
Key Account Management System	CRM : Key Account Management System	Used for managing key account customers	Current
Web Shop	Customer portal : Web Shop		Current
Invoice Verification System	Invoice System : Invoice Verification System	(Rechnungsprüfung)	Current
Work Scheduling System	Production Planning System : Work Scheduling System	The work scheduling system is used for preparing and planning the work	Current

9.1.3 Technical and Infrastructure Landscape Models in Iteraplan

The following pages contain the Building Blocks created in Iteraplan that belong to the Technical and Infrastructure Landscape Models.

Technical Components

Technical Component	Description
Mail Server	A server for sending and receiving mails
Web Server	A Server for hosting the Web Shop

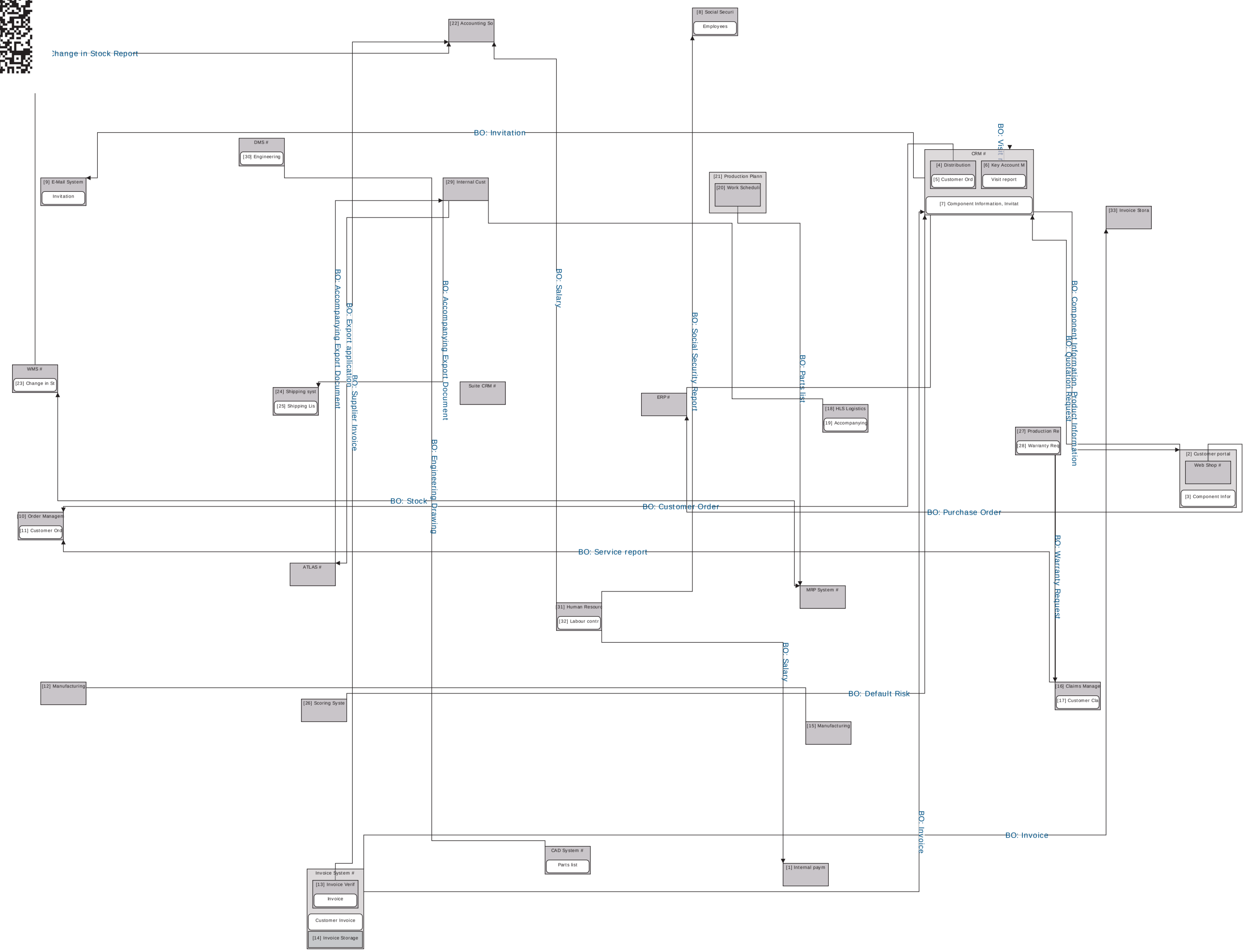
9.1.4 Generated Iteraplan Diagrams

The following pages contain a selection of diagrams generated by Iteraplan.

iteraplan Information Flow Diagram

Based on saved query:
InformationFlowAsPNG

Legend	Attribute / Building Block
Colour	n.a.
Line type	n.a.
Line caption	Business Objects



iteraplan Information Flow Diagram

Attachment

Number	Geschäftsobjekte
3	Component Information, Product Information, Customer quotation
5	Customer Order
7	Component Information, Invitation, Contact, Visit report, Product Information
11	Customer Order
17	Customer Claim
19	Accompanying Export Document, Loading list
23	Change in Stock Report
25	Shipping List
28	Warranty Request
30	Engineering Drawing
32	Labour contract, Employment Contract, Insurance Claim

Number	Informationssysteme
1	Internal payment transaction system #
2	Customer portal #
4	Distribution EDI system #
6	Key Account Management System #
8	Social Security Report System #
9	E-Mail System #
10	Order Management System #
12	Manufacturing Execution System (MES) #
13	Invoice Verification System #
14	Invoice Storage System (Excel)
15	Manufacturing Management System #
16	Claims Management System #
18	HLS Logistics System #
20	Work Scheduling System #
21	Production Planning System #
22	Accounting Software #
24	Shipping system #
26	Scoring System #
27	Production Report System (Customer) #
29	Internal Custom Handling System #
31	Human Resource Management System #
33	Invoice Storage System (Excel) #

iteraplan Master plan Diagram

Content: Information System

Information System	Status	Begin	End	2015 (Nov, Dec) / 2016 (Jan, Feb, Mar, Apr, May, Jun, Jul, Aug, Sep, Oct, Nov)
Accounting Software	Current	-	-	Productive from / to
ATLAS	Current	-	-	Productive from / to
CAD System	Current	-	-	Productive from / to
[1] Claims Management Sys	Current	-	-	Productive from / to
Customer portal	Current	-	-	Productive from / to
[2] Distribution EDI syst	Current	-	-	Productive from / to
E-Mail System	Current	-	-	Productive from / to
ERP	Current	-	-	Productive from / to
HLS Logistics System	Current	-	-	Productive from / to
[3] Human Resource Manag	Current	-	-	Productive from / to
[4] Internal Custom Handl	Current	-	-	Productive from / to
[5] Internal payment tran	Current	-	-	Productive from / to
[6] Invoice Storage Syste	Current	-	-	Productive from / to
Invoice System	Current	-	-	Productive from / to
[7] Invoice Verification	Current	-	-	Productive from / to
[8] Key Account Manageme	Current	-	-	Productive from / to
[9] Manufacturing Executi	Current	-	-	Productive from / to
[10] Manufacturing Manage	Current	-	-	Productive from / to
MRP System	Current	-	-	Productive from / to
[11] Order Management Sys	Current	-	-	Productive from / to
[12] Production Report Sys	Current	-	-	Productive from / to
Scoring System	Current	-	-	Productive from / to
Shipping system	Current	-	-	Productive from / to
[13] Social Security Repor	Current	-	-	Productive from / to
Suite CRM	Current	-	-	Productive from / to
Web Shop	Current	-	-	Productive from / to
WMS	Current	-	-	Productive from / to
Work Scheduling System	Current	-	-	Productive from / to

Number	Full Name
1	Claims Management System
2	Distribution EDI system
3	Human Resource Management System
4	Internal Custom Handling System
5	Internal payment transaction system
6	Invoice Storage System (Excel)
7	Invoice Verification System
8	Key Account Management System
9	Manufacturing Execution System (MES)
10	Manufacturing Management System
11	Order Management System
12	Production Report System (Customer)
13	Social Security Report System

iteraplan Landscape Diagram - Report Date 11/10/2015 12:59
Axes: Business Processes and Business Units
Content: Information Systems

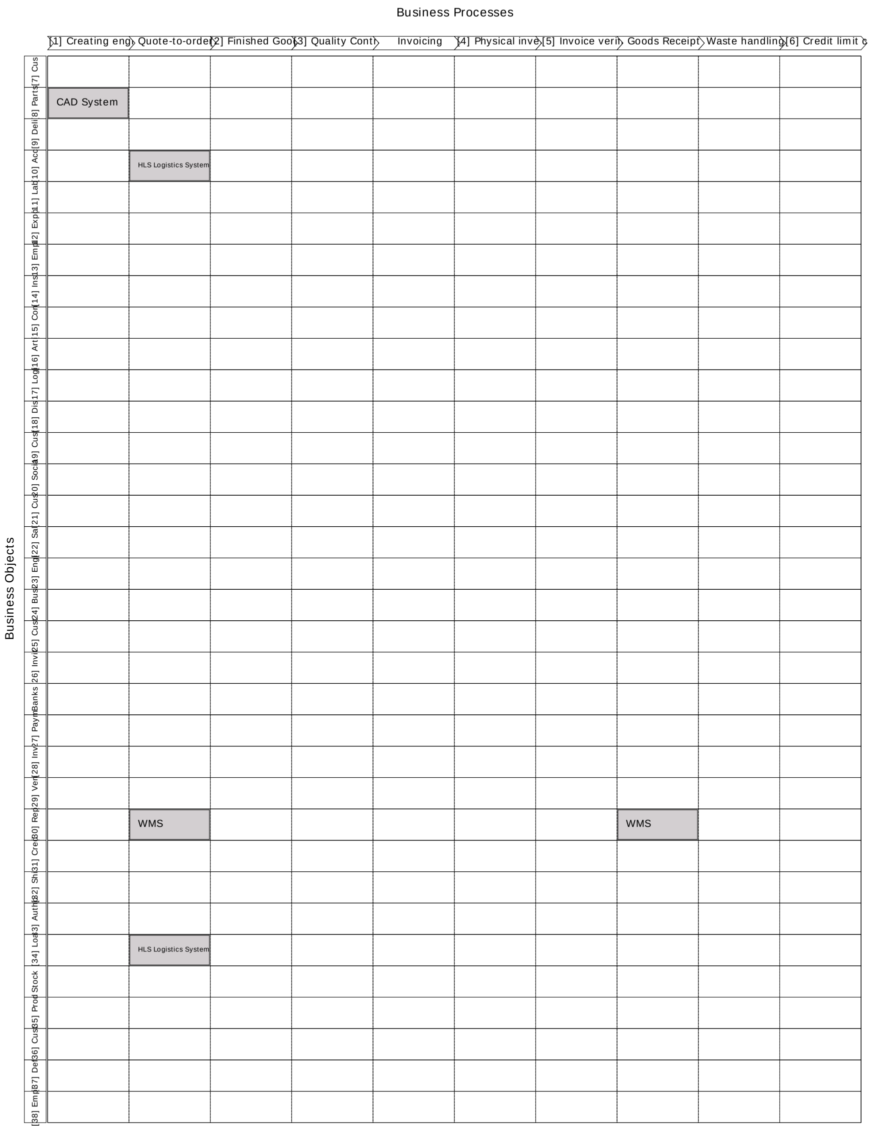
Business Processes

Business Units	[1] Creating engineering	Quote-to-order	Finished Goods Report	Quality Control	Invoicing	Physical inventory	Invoice verification	Goods Receipt	Waste handling	Credit limit check
Accounting										
[2] Quality ma										
Production		Work Scheduling System WMS								
Warehouse								WMS		
3] Human Reso										
Purchasing										
[4] Engineerin	CAD System									
[5] Distributi										
[6] After-Sale										
[7] Requiremen										
Marketing										

iteraplan Landscape Diagram - Report Date 11/10/2015 12:59
Axes: Business Processes and Business Units
Content: Information Systems
Attachment

Numbe	Full Name
1	Creating engineering drawing
2	Quality management
3	Human Resources
4	Engineering
5	Distribution
6	After-Sales-Service
7	Requirements Planning

iteraplan Landscape Diagram - Report Date 11/10/2015 12:57
Axes: Business Processes and Business Objects
Content: Information Systems



iteraplan Landscape Diagram - Report Date 11/10/2015 12:57
Axes: Business Processes and Business Objects
Content: Information Systems
Attachment

Numbe	Full Name
1	Creating engineering drawing
2	Finished Goods Report
3	Quality Control
4	Physical inventory
5	Invoice verification
6	Credit limit check
7	Customer Order
8	Parts list
9	Delivery reminder
10	Accompanying Export Document
11	Labour contract
12	Export application
13	Employees
14	Insurance Claim
15	Contact
16	Article Base List
17	Logistics
18	Dispatch advise
19	Customer quotation
20	Social Security Report
21	Customer Claim
22	Salary
23	Engineering Drawing
24	Business Partners
25	Customer
26	Invitation
27	Payment Proposal
28	Invoice
29	Vendor
30	Report
31	Credit reporting and scoring bureau
32	Shipping List
33	Authorities
34	Loading list
35	Product Information
36	Customer Request
37	Default Risk
38	Employment Contract

Bibliography

[ABB+11] Auer, Gerhard ; Basten, David ; Berneaus, Marcel ; Däberitz, Dirk ; Freitag, Andreas ; Haas, Holger ; Kröber, Gunter ; Schmidtmann, Verena ; Schweikert, Rainer ; Stettiner, Eric ; Thielscher, Jan ; Triebel, Thomas ; Weber, Mathias ; Weisbecker, Anette: Enterprise Architecture Management - neue Disziplin für die ganzheitliche Unternehmensentwicklung. In: *Bitkom* (2011), 1–46. http://www.bitkom.org/files/documents/EAM_Enterprise_Architecture_Management_-_BITKOM_Leitfaden.pdf

[AE08] Assmann, Martin ; Engels, Gregor: Transition to service-oriented enterprise architecture. Version: 2008. http://dx.doi.org/10.1007/978-3-540-88030-1{_}34. In: *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)* Bd. 5292 LNCS. 2008. – DOI 10.1007/978–3–540–88030–1_34. – ISBN 3540880291, 346–349

[Ahl12] Ahlemann, Frederik: *Strategic Enterprise Architecture Management: Challenges, Best Practices, and Future Developments.* 2012. – Online–Ressource (XVI, 298p. 39 illus, digital) S. http://dx.doi.org/10.1007/978-3-642-24223-6. http://dx.doi.org/10.1007/978-3-642-24223-6. – ISBN 978–3–642–24223–6

[Ame] American Marketing Association: *Definition of Marketing.* https://www.ama.org/AboutAMA/Pages/Definition-of-Marketing.aspx

[BBL12] Bente, Stefan ; Bombosch, Uwe ; Langade, Shailendra: *Collaborative Enterprise Architecture.* 2012. – 39–104 S. http://dx.doi.org/10.1016/B978-0-12-415934-1.00003-0. http://dx.doi.org/10.1016/B978-0-12-415934-1.00003-0. – ISBN 9780124159341

[Biza] Bizagi: *Bizagi - BPMN Quick Reference Guide.* https://www.bizagi.com/docs/BPMN_Quick_Reference_Guide_ENG.pdf. https://www.bizagi.com/docs/BPMN_Quick_Reference_Guide_ENG.pdf

[Bizb] Bizagi: *Bizagi Process Modeler User's Guide - Artifacts*. http://help.bizagi.com/processmodeler/en/index.html?artifacts.htm

[Bizc] Bizagi: *Simulation in Bizagi*. http://help.bizagi.com/processmodeler/en/index.html?simulation{_}in{_}bizagi.htm

[Brä14] Bräuer, Alexander: *Kein Widerrufsrecht bei individuell angefertigten Waren*. Website. https://www.ratgeberrecht.eu/wettbewerbsrecht-aktuell/kein-widerrufsrecht-bei-individuell-angefertigten-waren.html. Version: 03 2014

[Bro] Brookshier, Daniel: *BPMN 2.0 Tutorial (Presentation)*. Web Page. http://www.omg.org/news/meetings/workshops/SOA-HC/presentations-2011/14_MT-2_Brookshier.pdf

[Deu15] Deutsche Zollverwaltung: *Zoll online - Grundlegende Informationen*. http://www.zoll.de/DE/Fachthemen/Zoelle/ATLAS/ATLAS-Allgemein/Grundlegende-Informationen/grundlegende-informationen_node.html. Version: 2015

[Deu86] Deutscher Bundestag: *Limited Liability Companies Act*. http://www.gesetze-im-internet.de/englisch{_}gmbhg/index.html. Version: 23.7.2013 (BGBl. I S. 2586)

[Eve15] Everett, Dave: *Behind the scenes at SRAM - a company tour — CyclingTips*. http://cyclingtips.com.au/2015/01/behind-the-scenes-at-sram/. Version: 2015

[Fen] Fenner, James: *Enterprise Application Integration Techniques*. http://www.cs.ucl.ac.uk/staff/ucacwxe/lectures/3C05-02-03/aswe21-essay.pdf

[FR14] Freund, Jakob ; Rücker, Bernd: *Praxishandbuch BPMN 2.0*. München / Wien : Hanser Fachbuch, 2014. – ISBN 978–3–446–44255–9

[FS11] Fogliatto, Flavio S. ; Silveira, Giovani J. C. ; Fogliatto, Flavio S. (Hrsg.) ; Silveira, Giovani J. C. (Hrsg.): *Mass Customization*. London : Springer London, 2011 (Springer Series in Advanced Manufacturing). http://dx.doi.org/

10.1007/978-1-84996-489-0. http://dx.doi.org/10.1007/978-1-84996-489-0. – ISBN 978–1–84996–488–3

[Gal15] GALLEGO, Guillermo: *IEOR 40000: Production Management - Lecture 1 - Material Requirements Planning (MRP)*. http://www.cvn.columbia.edu/content/coursereview?course=IEORE4000&sem=M12. Version: 2015

[GN05] GUNASEKARAN, a. ; NGAI, E. W T.: Build-to-order supply chain management: A literature review and framework for development. In: *Journal of Operations Management* 23 (2005), Nr. 5, 423–451. http://dx.doi.org/10.1016/j.jom.2004.10.005. – DOI 10.1016/j.jom.2004.10.005. – ISBN 0272–6963

[GPSa] GPS SOFTWARE AG: *SoftwareAtlas — Das Tool*. Website. http://www.gps-ulm.de/web/de/software-atlas/das-tool.php

[GPSb] GPS SOFTWARE AG: *SoftwareAtlas — Referenzmodelle*. Website. http://www.gps-ulm.de/web/de/software-atlas/referenzmodelle.php

[Han09] HANSCHKE, Inge: *Strategic IT Management: A Toolkit for Enterprise Architecture Management*. Springer, 2009. – 353 S. http://dx.doi.org/10.1007/978-3-642-05034-3. http://dx.doi.org/10.1007/978-3-642-05034-3. – ISBN 3642050336

[HAW] HAWAI PROJECT: *HAWAI Project Website*. http://hawai-project.mars.haw-hamburg.de/wordpress/

[HBT03] HITE, Randolph C. ; BARKAKATI, NABA; BIRD, MARK; COLLIER, BARBARA; DAVIS, DEBORAH; DOHERTY, NEIL; GOLDSTEIN, Tamra; ; TEKELEY, Randolph: *GAO-03-584G Information Technology: A Framework for Assessing and Improving Enterprise Architecture Management (Version 1.1)*. http://www.gao.gov/new.items/d03584g.pdf. Version: 2003

[HRSS15] HÜNING, Christian ; RAMCKE, Eike-Christian ; SARSTEDT, Stefan ; STEFFENS, Ulrike: Business Process simulation with the cloud-based Massive Multi-Agent System MARS. In: *AKWI 2015*. Hamburg / Luzern : HAW Hamburg, 2015

[Hua12] HUANG, IVAN; ROC, GUO; XIE, HARRY; ZHENGXIAN, WU: The Convergence of Information and Communication Technologies Gains Momentum. Version: 2012. http://www3.weforum.org/docs/GITR/2012/

GITR{_}Chapter1.2{_}2012.pdf. In: Forum, World E. (Hrsg.): *Glob. IT Rep. 2012*. World Economic Forum, 2012. – ISBN 978–92–95044–33–3, Kapitel Chapter 1.

[IEE00] IEEE: *IEEE Std 1471-2000*

[Int03] International Business Machines Corp.: *IBM Archives: The IBM Personal Computer*. https://www-03.ibm.com/ibm/history/exhibits/pc25/pc25_intro.html. Version: Januar 2003

[ISO11] ISO/IEC/IEEE: *ISO/IEC/IEEE 42010:2011*. http://dx.doi.org/10.1109/IEEESTD.2011.6129467. Version: 2011

[Itea] Iteratec: *Editionen von iteraplan*. http://www.iteraplan.de/editionen/

[Iteb] Iteratec: *Installation Guide - iteraplan 3.4 Documentation - iteraplan Documentation*. http://www.iteraplan.de/doc/display/iteraplan34/Installation+Guide

[Itec] Iteratec: *iteratec Best-Practice-EAM - iteraplan 3.4 Documentation - iteraplan Documentation*. http://www.iteraplan.de/doc/display/iteraplan34/iteratec+Best-Practice-EAM

[ITI] ITIL: *ITIL Glossary*. Website. http://os.itil.org/custom/glossaren/

[KH12] Köber, J. ; Heinecke, G.: Hybrid production strategy between make-to-order and make-to-stock - A case study at a manufacturer of agricultural machinery with volatile and seasonal demand. In: *Procedia CIRP* 3 (2012), Nr. 1, 453–458. http://dx.doi.org/10.1016/j.procir.2012.07.078. – DOI 10.1016/j.procir.2012.07.078. – ISBN 2212–8271

[KM15] Kathuria, Sanjay ; Malouche, Mariem M.: *Toward New Sources of Competitiveness in Bangladesh: Key Insights of the Diagnostic Trade Integration Study*. The World Bank, 2015. http://dx.doi.org/10.1596/978-1-4648-0647-6. http://dx.doi.org/10.1596/978-1-4648-0647-6. – ISBN 978–1–4648–0647–6

[Kot89] Kotler, Philip: From mass marketing to mass customization. In: *Plan. Rev.* 17 (1989), Nr. 5, 10–47. http://dx.doi.org/10.1108/eb054267. – DOI 10.1108/eb054267. – ISBN 0094064X

[Kot96] Kotha, Suresh: From mass production to mass customization: The case of the national industrial bicycle company of Japan. In: *Eur. Manag. J.* 14 (1996), oct, Nr. 5, 442–450. http://dx.doi.org/10.1016/0263-2373(96)00037-0. – DOI 10.1016/0263–2373(96)00037–0. – ISBN 0263–2373

[Kul15] Kull, Hans: *Mass Customization*. Bd. 1. Berkeley, CA : Apress, 2015. http://dx.doi.org/10.1007/978-3-642-18390-4. http://dx.doi.org/10.1007/978-3-642-18390-4. – ISBN 978–1–4842–1008–6

[Lac] Lackes, Richard; Siepermann, Markus: *Gabler Wirtschaftslexikon - Stichwort: Datenintegrität*. (online) : Springer / Gabler Verlag (Herausgeber) http://wirtschaftslexikon.gabler.de/Archiv/57132/datenintegritaet-v9.html

[LC04] Land, Rikard ; Crnkovic, Ivica: Existing Approaches to Software Integration - and a Challenge for the Future. In: *Fourth Conference on Software Engineering Research and Practice in Sweden*, 2004

[Lew10] Lewis, Grace: *Getting Started with Service-Oriented Architecture (SOA) Terminology*. http://www.sei.cmu.edu/library/assets/whitepapers/SOA_Terminology.pdf. Version: 2010

[Lin99] Linthicum, David S.: *Enterprise Application Intergration*. Addison-Wesley, 1999. – 1–304 S. – ISBN 0–201–61583–5

[Lin05] Lin, Florence: *Enterprise Application Integration (EAI) Techniques*. Paper - Web Site. http://www0.cs.ucl.ac.uk/staff/ucacwxe/lectures/3C05-04-05/EAI-Essay.pdf. Version: 2005

[LMN08] Lalitrao, K. ; Mallikarjunasarcar, K. ; Narayan, M.M.M.: *Computer Aided Design and Manufacturing*. Paperback. Delhi : PHI Learning, 2008. – ISBN 978–8120333420

[LO09] Liu, Ling ; Özsu, M T.: *Encyclopedia of Database Systems*. 2009. – 1853–1857 S. http://dx.doi.org/10.1007/978-0-387-39940-9. http://dx.doi.org/10.1007/978-0-387-39940-9. – ISBN 9780387355443

[LW58] Leavitt, Harold J. ; ; Whisler, Thomas L.: Management in the 1980s - HBR. In: *Harvard Bus. Rev.* (1958). https://hbr.org/1958/11/management-in-the-1980s

[Man09] Mankovskii, Serguei ; LIU, LING (Hrsg.) ; ÖZSU, M. T. (Hrsg.): *Encyclopedia of Database Systems - Tight Coupling*. Boston, MA : Springer US, 2009. http://dx.doi.org/10.1007/978-0-387-39940-9. http://dx.doi.org/10.1007/978-0-387-39940-9. – ISBN 978–0–387–35544–3

[Mas07] Masak, Dieter: *SOA? - Serviceorientierung in Business und Software*. Berlin, Heidelberg : Springer Berlin Heidelberg, 2007 (Xpert.press). http://dx.doi.org/10.1007/978-3-540-71872-7. http://dx.doi.org/10.1007/978-3-540-71872-7. – ISBN 978–3–540–71871–0

[Mat] Matthes, Florian; Hauder, Matheus; Katinszky, Nikolaus: Iteraplan. https://wwwmatthes.in.tum.de/file/ru0jyblhtjsu/Sebis-Public-Website/-/Enterprise-Architecture-Management-Tool-Survey-2014-Update/EAMTS2014Update.pdf?details=truehttps://wwwmatthes.in.tum.de/file/ru0jyblhtjsu/Sebis-Public-Website/-/Enterprise-Architecture-Management-Tool-Survey-2014-Update/EAMTS2014Update.pdf. In: *Enterp. Archit. Manag. Tool Surv. 2014 Updat.* Munich, Germany : sebis TU München, 57ff.

[Men] Menge, Falco: Enterprise Service Bus. In: *FrOSCon (Free Open Source Softw. Conf. 2007*

[Obj] Object Management Group: *Customizing and Extending Oracle BPM Project Templates - 11g Release 6 (11.1.6)*. Documentation. http://docs.oracle.com/cd/E29597_01/fusionapps.1111/e16691/cust-bpm-proj-templates.htm

[Obj08] Object Management Group: *BPMNBusiness Process Model and Notation (BPMN) Version 1.1*. http://www.omg.org/spec/BPMN/1.1/PDF/. Version: 2008. http://www.omg.org/spec/BPMN/1.1/PDF/

[Obj11] Object Management Group: *Business Process Model and Notation (BPMN) Version 2.0*. http://www.omg.org/spec/BPMN/2.0/PDF. Version: 2011

[Obj13] Object Management Group: *INTERNATIONAL STANDARD ISO 19510:2013 / IEC Information technology - Object Management Group Business Process*. 2013

[Old13] OldComputers.net: *IBM 5150 Personal Computer*. http://oldcomputers.net/ibm5150.html. Version: 2013

[OSHS07] Offermann, Philipp ; Schröpfer, Christian ; Holschke, Oliver ; Schönherr, Marten: SOA: The IT-Architecture behind Service-Orientation. In: *MDD SOA und ITManagement Workshop*, 2007. – ISBN 9783970019066, S. 1–11

[PMS04] Piller, Frank T. ; Moeslein, Kathrin ; Stotko, Christof M.: Does mass customization pay? An economic approach to evaluate customer integration. In: *Production Planning and Control* 15 (2004), Nr. 4, 435-444. http://dx.doi.org/10.1080/0953728042000238773. – DOI 10.1080/0953728042000238773. – ISBN 09537287 (ISSN)

[PZL08] Pautasso, Cesare ; Zimmermann, O. ; Leymann, Frank: Restful web services vs. 'big'web services: making the right architectural decision. In: *Proceeding 17th Int. Conf. World Wide Web* (2008), 805–814. http://dx.doi.org/10.1145/1367497.1367606. – DOI 10.1145/1367497.1367606. ISBN 978–1–60558–085–2

[Raj13] Rajesh Pattabiraman: *Comparing and Contrasting SOA Variants for Enterprise Application Integration*. New York / London / Chennai, 2013

[SA12] Soomro, Tr ; Awan, Ah: Challenges and Future of Enterprise Application Integration. 42 (2012), Nr. 7, 42–45. http://dx.doi.org/10.5120/5707-7762. – DOI 10.5120/5707–7762

[SAP] SAP: *Inquiry and Quotation Processing - SAP Engineering & Construction - SAP Library*. http://help.sap.com/saphelp_46c/helpdata/en/e0/10a837a703934de10000009b38f8cf/content.htm

[SAW04] Stephen A. White, IBM C.: *Whitepaper: Introduction to BPMN*. http://www.omg.org/bpmn/Documents/Introduction_to_BPMN.pdf. Version: 2004

[Sch] Schekkerman, Jaap: *How to Survive in the Jungle of Enterprise Architecture Frameworks: Creating or Choosing an Enterprise Architecture Framework*. 2nd Editio. Bloomington, IN : Traffington. – ISBN 141201607X

[Sch12] Schwab, Frank: *Zukunftssichere IT Landschaften von Banken*. Web Page. http://www.imacor.eu/Zukunftssichere_IT_Landschaften_von_Banken.html. Version: Q3 2012

[Sch13] Schröder, Jürgen; Spillecke, Dennis: Experimenting without risk at the Digital Marketing Factory. Version: Perspectiv, 2013. http://www.mckinsey.

com/{~}/media/mckinsey/dotcom/client{_}service/retail/articles/perspectivesbook/07digitalmarketingfactory.ashx. In: *Perspect. Retail Consum. goods - Number 1, Spring 2013.* Perspectiv. McKinsey, 2013, 36–39

[Sea] Searates.com: *Transit Time, Distance calculator & Port to port distances.* https://www.searates.com/reference/portdistance/?B=737{&}E=1650{&}

[seb] sebis TU München: *List of EAM Tools.* https://wwwmatthes.in.tum.de/pages/exuqsrwkrch6/List-of-EAM-Tools

[Shi] Shimano: *Locations.* http://www.shimano.com/content/Corporate/english/index/portal-site/location.html

[Sof] SoftSelect: *Definition DMS - Dokumenten-Management-System Erklärung DMS - Dokumenten-Management-System.* http://www.softselect.de/business-software-glossar/dms-dokumenten-management-system

[Sou] Sourceforge.net: *iteraplan Sources at SourceForge.net.* http://sourceforge.net/projects/iteraplan/files/iteraplanCommunityEdition/

[Ste05] Steinbauer, Jörg: Convergent Next-Generation Service Management. In: *FITCE 2005.* Vienna, Austria : Forum for European ICT & Media Professionals / BearingPoint INFONOVA GmbH, 2005, 2

[SW04] Sprott, D ; Wilkes, L: *Understanding service-oriented architecture.* http://dx.doi.org/10.1007/1-4020-3705-8{_}1. Version: 2004

[SWB+12] Shapiro, Robert ; White, Stephen A. ; Bock, Conrad ; Palmer, Nathaniel ; Muehlen, Michael zur ; Brambilla, Marco ; Gagné, Denis: *BPMN 2.0 Handbook Second Edition: Digital Edition.* Lighthouse Point, FL, USA : Future Strategies Inc, 2012. – ISBN 9780984976416

[The] The Open Group: *TOGAF Users by Market Sector — The Open Group.* http://www.opengroup.org/togaf/users-by-market-sector

[The09] The Open Group: *Service Oriented Architecture.* http://www.opengroup.org/soa/source-book/soa/index.htm. Version: 2009

[The11a] The Open Group: *SOA Source Book - SOA Reference Architecture Technical Standard (Document C119)*. https://www.opengroup.org/soa/source-book/soa_refarch/index.htm. Version: 2011

[The11b] The Open Group: *The Open Group Architecture Framework (TOGAF) version 9.1*. 2011

[The13] The Open Group: *SOA and Enterprise Architecture*. http://www.opengroup.org/soa/source-book/soa/soa{_}ea.htm. Version: 2013

[TRH+04] Trowbridge, David ; Roxburgh, Ulrich ; Hohpe, Gregor ; Manolescu, Dragos ; Nadhan, E D.: *Integration Patterns*. Bd. 89. Microsoft Press, 2004 http://www.amazon.de/Integration-Patterns-Practices-Microsoft-Corporation/dp/073561850X. – ISBN 073561850X

[US 95] US International Trade Commission: *Bicycles from China - (Publication 2893)*. http://www.usitc.gov/publications/701_731/pub2893.pdf. Version: 1995

[Wel] Welt der BWL: *GmbH Definition - Gesellschaft mit beschränkter Haftung — Rechtsformen*. http://welt-der-bwl.de/GmbH

[Zac87] Zachman, Jonathan: A framework for information systems architecture. In: *IBM Syst. J.* 26 (1987), Nr. 3, 276–292. https://web.archive.org/web/20070120040422/http://www.research.ibm.com/journal/sj/263/ibmsj2603E.pdf

[ZX11] Zhang, Linda ; Xu, Qianli: Designing an integrated order fulfillment system for configure-to-order production. In: *IEEE International Conference on Industrial Engineering and Engineering Management* (2011), S. 1810–1814. http://dx.doi.org/10.1109/IEEM.2011.6118228. – DOI 10.1109/IEEM.2011.6118228. – ISBN 9781457707391

Glossary

A2A Application-to-Application. 17, 99

AGPL A type of Open Source License. 42, 99

API Application Programming Interface. 18, 99

ARIS "Architektur Integrierter Systeme" (ARIS) / Architecture of Integrated Information Systems). 45, 99

ATLAS Automatisiertes Tarif- und Lokales Zollabwicklungssystem. 36, 99

B2B Business to Business. 17, 99

BPD Business Process Diagram. 49, 99

BPMN Business Process Model Notation. 45, 49, 55, 99

BTO Built-To-Order. 30, 99

BTS Built-To-Stock. 30, 99

CAD Computer Aided Design. 36, 99

Contract manufacturers A independent company that assembles components. 40, 99

CRM Customer Relationship Management. 17, 36, 99

CT Communication Technology. 1, 99

CTO Custom-To-Order. 30, 99

DDM Direct Digitized Manufacturing. 29, 99

DMS Document Management System. 36, 99

DoDAF Department of Defense Architecture Framework. 10, 99

E2AF Extended Enterprise Architecture Framework. 10, 99

EA Enterprise Architecture. 9, 11, 99

EAF Enterprise Architecture Framework. 8, 10, 99

EAI Enterprise Architecture Integration. 8, 16, 99

EAM Enterprise Architecture Management. 2, 8, 9, 99

EDI Electronic Data Interchange. 33, 99

EDIFACT Electronic Data Interchange For Administration, Commerce and Transport. 99

EPC Event Driven Process Chain. 45, 99

ERP Enterprise Resource Planning. 32, 99

FEAF US Federal Architecture Framework. 10, 99

GmbH A GmbH is a privately held limited liability company and as such a corporate body governed by private law. The English version of the GmbHG (Limited Liability Companies Act) is available in the Web at [Deu86]. 4, 99

HAWAI Hamburg University of Applied Sciences Application Integration Lab. 2, 99

HR Human Resources. 99

HTTP Hypertext Transfer Protocol. 26, 99

ICT Information and Communication Technology. 3, 99

IT Information Technology. 1, 99

MES Manufacturing Execution System. 99

MMS Manufacturing Management System. 99

MRP Material Resource Planning. 36, 99

MTO Make-To-Order. 30, 99

OMG Object Management Group. 49, 99

P2P Peer-To-Peer. 21, 99

P2S Person to System. 17, 99

R&D Research and Development. 34, 99

REST Representational State Transfer. 26, 99

sebis **S**oftware **E**ngineering for **B**usiness **I**nformation **S**ystems - an Institute at TU München. 63, 99

SOA Service Oriented Architecture. 8, 23, 99

SOEA Service-Oriented Enterprise Architecture. 24, 99

TBI Total Business Integration. 17, 99

TOGAF The Open Group Architecture Framework. 10, 99

UI User Interface. 20, 99

WMS Warehouse Management System. 99

Versicherung über die Selbstständigkeit

Hiermit versichere ich, dass ich die vorliegende Arbeit ohne fremde Hilfe selbständig verfasst und nur die angegebenen Hilfsmittel benutzt habe.

Hamburg, 18. November 2015

Christian Gläser