

A Lightweight Method for the Modeling of Enterprise Architectures: Introduction and Empirical Validation

Henk Koning

Rik Bos

Sjaak Brinkkemper

Department of Information and Computing Sciences

Utrecht University

Technical Report UU-CS-2006-003

www.cs.uu.nl

ISSN: 0924-3275

A Lightweight Method for the Modeling of Enterprise Architectures: Introduction and Empirical Validation

Henk Koning, Rik Bos, Sjaak Brinkkemper,

Department of Information and Computing Sciences
University of Utrecht,
P.O.Box 80.089
3508 TB Utrecht, the Netherlands
{[h.koning](mailto:h.koning@cs.uu.nl), [rik](mailto:rik@cs.uu.nl), [s.brinkkemper](mailto:s.brinkkemper@cs.uu.nl)}@cs.uu.nl.

Abstract. This paper defines the high level key concepts necessary to describe enterprise architectures in a manner that is easy to understand and to communicate. We introduce the Enterprise Architecture Modeling method (EAM), which consists of four types of diagrams: the Supply Chain Diagram, showing the main relationships of the enterprise in its business environment; the Enterprise Function Diagrams for the interoperation of enterprise functions; the Scenario Overlay for modeling the most important business processes on top of the Enterprise Function Diagram; and the System Infrastructure Diagram, depicting the technical infrastructure of IT systems and networks. To date, we conducted about 40 case studies in almost all sectors of society. Examples of the diagrams taken from a case study of a recreational accommodation company illustrate the usage of EAM in practice. For further empirical validation we performed an enquiry among modelers and among readers of the enterprise architecture reports.

1 Enterprise Architecture Modeling in Practice

Within organizations, especially the larger ones, the use of Enterprise Architecture becomes ever more important. Enterprise architecture descriptions capture the essentials of the business and link them to the essentials of the IT-support. They need to be clearly understandable by business managers as well as by IT-specialists. The language in text and diagrams should be intuitively clear, without a need to learn many technical terms or icons. Despite its importance, we are still far away from a universal architecture language, let alone a universal approach for constructing enterprise architectures. Most existing approaches are burdened under lots of details, while we believe that the main focus of Enterprise Architecture should be on models that capture the essentials and are yet easy to communicate and from which it is therefore also relatively easy to take strategic decisions upon, e.g. outsourcing part of the business. With this paper we hope to bring these desired qualities within the reach of a broad public. Modeling enterprise architecture is a relatively new field and it is hard to find any empirical research to describe it. For the work described in this paper we base ourselves on our extensive experience in practice and academic interest with modeling in IT and architecting in systems development. See [4] for some recent

definitions of EA.

In this paper we propose a method specifically meant for capturing the essentials of an enterprise architecture and at the same time facilitating easy communication. This method was inspired by various high level business modeling formalisms introduced by vendors of ERP software; ARIS [12][13] for SAP, and Dynamic Enterprise Modeling [2] for Baan implementations.

Why are organizations interested in Enterprise Architecture? The reason is that nowadays many complex issues reside on the agenda of the board of the organization. Issues like:

Outsourcing. Should we outsource certain business functions? Which? How?

Organization. Should we change the structure because of new products and services? How can we accommodate the consequences of the merger?

IT support. Should we start implementing an ERP system in that unit? When can we replace system X with a standard product?

Service continuity. Is our IT infrastructure trustworthy for 7x24? Do we get back to our customers in time?

And many, many more ... See [3] for some more enterprise wide concerns, that are addressed by enterprise architectures.

All these decisions involve strategic assets: budget, the organization, jobs, IT systems, product and services. The board has the need to decide and communicate in an efficient and transparent manner. Enterprise architectures are one of the means to assist these types of decisions. Therefore for the enterprise architect is confronted with the question of what documentation should be provided and what communication should be performed to support transparent decision making? The efficiency of decision making is also heavily dependent on the simplicity of the formalism. So our problem statement for this paper is: how can we describe Enterprise Architecture in a way that is easy to create (author) and easy to understand (reader)?

Based on earlier experiences with enterprise modeling for ERP implementations we have developed an Enterprise Architecture Modeling method (EAM). EAM was introduced to master students of Business Informatics in an elective course. Furthermore, some master research projects were executed using EAM as modeling formalism. In the course the students were given the task to try this method out within a company and under our supervision. As it turned out, the results obtained by the students were quite amazing given the restricted scope and time. We were pleasantly surprised by the clear understanding of the essential operating of the companies that showed from the student papers. Also the contact persons from the organizations were very satisfied with the resulting papers, which started to serve the internal communication. We will present parts of one of these case studies.

1.1 Related work

Several methods for developing enterprise architectures are available nowadays, e.g. ArchiMate [1] and ARIS [12][13]. Each method has its own strengths and usually has

its focus on one or more specific points, e.g. integration between different models, alignment between business and IT, business processes, communication etc.

The Archimate project has produced an elaborate language for describing enterprise architectures. The conceptual model consists of 26 entities. The visual representation consists of 40 symbols with each a specific meaning. It takes a while to get to learn each of the symbols and, in our experience, after some time of not working with the language one has to relearn the specific meaning of each of the symbols. The authors of Archimate have proposed 16 diagram types (viewpoints) as a basic set to work with the language, but many more could be constructed. The Archimate diagram type 'Business Function Viewpoint' comes very close to our Enterprise Function Diagram. They state 'Business functions are used to represent what is most stable about a company in terms of the primary activities it performs, regardless of organizational changes or technological developments' (p. 177).

ARIS is an often cited method for describing business processes. It is even more complex than Archimate. It has (only) 15 basic views, but many, many diagram types to populate the views. On the summary page ([11], p. 78) we count 23 'main' diagram types. The business process metamodel contains 300 entities and relations ([11], p. 48). For students the time needed to master the basics of ARIS is not proportionate to the analysis time needed to study the enterprise architecture of a company. The emphasis in ARIS on business processes leads to a too detailed view of the organization.

The well known framework of Zachman [14][19] gives aid in partitioning the architectural information, but gives no guidance regarding the modeling of the information. The same goes for an architecture process description like TOGAF [15]. It gives guidance regarding the activities of the architect, but does not have a modeling method. Kruchten (1995)[8] gives a framework and modeling method, but is geared toward software architecture and not enterprise architecture.

Dynamic Enterprise Modeling [2] was introduced by Baan in 1996 as a means for implementing the Baan ERP product. The modeling focused on a Petri-net based technique for business process modeling to which the Baan application units were to be linked. DEM consists also of a supply chain diagram tool for the logistic network of the company and of a business function modeling diagram. The latter two diagrams formed an inspiration for our Enterprise Architecture Modeling method. The business process modeling turns out to be too detailed for modeling enterprise architectures.

The École Polytechnique Fédérale de Lausanne has developed an object-oriented enterprise architecture method, called "SEAM"[18]. As part of SEAM, they have developed the CAD tool "SeamCAD" [10]. SeamCAD enables the modeling of hierarchical systems (spanning from business down to IT) at different levels of detail (e.g. from large business transaction to detailed interactions). It has a philosophical underpinning which is not so easy to understand. The notation is UML-like and some training is needed to read the diagrams.

1.2 Outline of the paper

In chapter 2 we present an overview of the Enterprise Architecture Modeling method and introduce a case study. In chapter 3 we present each of the models of EAM and illustrate them with example diagrams from the case study. In chapter 4 we outline our evaluation efforts by a large series of case studies and two questionnaires. We finish the paper with conclusions and future work

2 Enterprise Architecture Modeling

We now want to introduce our solution, the Enterprise Architecture Modeling method (EAM). In our view an enterprise architecture should describe the essential functioning of an enterprise and the essential functioning of the IT-support for the enterprise. These essentials lay the groundwork which shapes the more detailed functioning of the organization and of the IT; therefore it is called an architecture. Our Enterprise Architecture Modeling (EAM) method endeavors to give a high level modeling of an enterprise and its IT-support. We have studied existing EA modeling methods and made a choice of key concepts needed for capturing high level essentials. See **Fig. 1** for the metamodel of our modeling method.

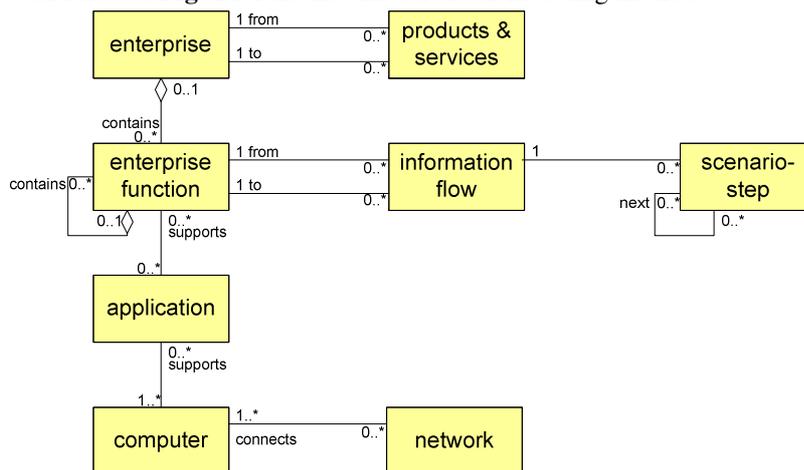


Fig. 1. The metamodel of the Enterprise Architecture Modeling method (EAM)

For modeling the enterprise the key concept is the enterprise function. To show the interoperation of the enterprise functions we also portray the flow of information. Scenarios indicate a sequence of information flows. To model the context of an enterprise we use enterprises connected by products & services. For modeling the IT-support our key concepts are computer, application and network (component). We have found that these key concepts are sufficient for creating a model that shows how in essence an enterprise functions. With these the current IT-support can be evaluated at a high level of abstraction and future IT-support can be planned. The Enterprise Architecture Modeling method contains these models:

- A *Supply Chain Diagram* (SCD) shows how the enterprise works together with business partners to produce the goods or services for the customers.
- An *Enterprise Function Diagrams* (EFD) gives a top level breakdown of the main functions of an enterprise.
- A *Scenario Overlay* (SO) shows how in a particular situation the functions in an EFD interoperate.
- A *System Infrastructure Diagram* (SID) shows the main network topology, the main computers that function in the network and the main information systems that run on these computers to support the enterprise functions.

See **Fig. 2.** for an overview of these models.

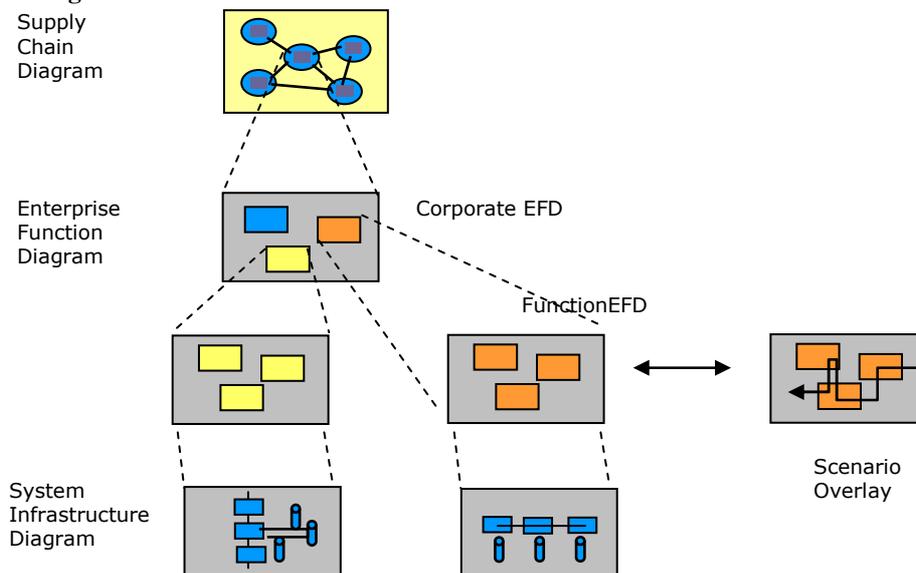


Fig. 2. Overview of EAM models

We will discuss these models each in turn in the next chapter. With each we give an example diagram taken from a case study performed at the company Center Parcs (CP). See Koning et al. [6] for many practical guidelines concerning the design of diagrams.

2.1 Case Context

Center Parcs Europe is the ‘European leader in short stays’. Its concept is based on short stays in a totally natural environment, open year round and far from the stress of the city. CPs headquarters is in Rotterdam, the Netherlands and over 10,000 full-time and part-time employees are currently working at CP all over Europe. CP is part of The Pierre & Vacances Groupe. Currently, CP offers about 10,000 bungalows and cottages in 20 parks. CP has parks in different categories, aiming at designated customer profiles. Each park has a Tropical Aquatic Paradise and every CP village is located within two hours driving time from urban areas. Four departments in every park take care of the customers. These departments are housekeeping, catering, retail and leisure (tennis, horse riding, squash and saunas). The water activities are one of

CPs most famous characteristics. In 2003, CP welcomed over 3 million guests while the turnover was about 525 million euros; 81% of the guests stay a weekend or midweek while only 2% stay more than a week.

3 The EAM Models

In this chapter we present each of the models of EAM and illustrate them with example diagrams from the case study

3.1 Supply Chain Diagram (SCD)

A Supply Chain Diagram is a model of the business context of the enterprise showing the external parties and the exchange of products and services. A SCD shows the business relationships of the enterprise in a kind of dependency network. Flows preferably show the exchange of products and services between the enterprises in the supply chain, and not the flow of information, unless the product or service regards information objects, such as for example tax forms for the National Taxation Office or weather report for a meteorological institute.

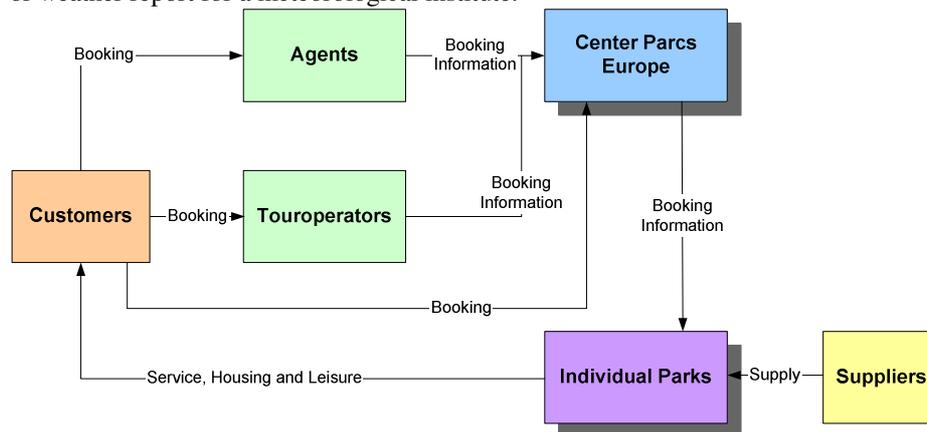


Fig. 3. SCD of Center Parcs

See **Fig. 3** for an example of a SCD. It shows how CP Europe and the Individual Parks cooperate with Agents, Touroperators and Suppliers to accommodate Customers. The boxes denote enterprise, the arrows the flow of products and services, or of information. In this example the two main units of CP are shown, the central Europe headquarters and all the individual parks, and the main external parties which we describe here one by one.

Customer The customers (visitors) of CP are able to make a booking in three different ways: (1) directly at CP via the website or the call center, (2) via a tour operator, who will subsequently handle all the customer contacts, or (3) via an agent, who will pass through the customer information to CP.

Tour Operator and Agent When a customer makes a booking via a tour operator or agent, the tour operator or agent is responsible for passing the booking information to CP. In case of a booking via a tour operator, the tour operator is concerned with the

financial handling. In the case of an agent, the agent only passes on the booking information to CP, and CP takes care of the financial handling.

Suppliers The suppliers primarily deal with the individual parks. They receive orders directly from the departments in the parks (e.g., souvenirs for the non-food retail departments), and ship their goods directly to the parks. The finance department within the park takes care of all the financial transactions.

The enterprises in an SCD are not decomposed further, but for large companies business units can be treated as separate enterprises. An enterprise architecture description contains one SCD.

3.2 Enterprise Function Diagrams (EFDs)

An Enterprise Function Diagram is a model from an enterprise business perspective. EFDs give a top level breakdown of the main functions of an enterprise.

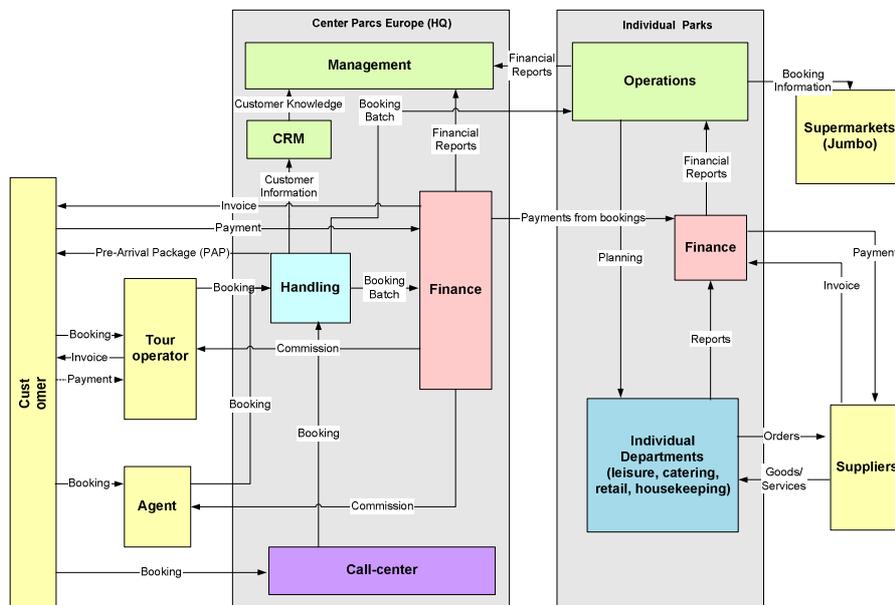


Fig. 4. EFD with the main enterprise functions of Center Parcs

See Fig. 4 for an example of an EFD. The boxes denote enterprise functions. To the left and to the right the external parties are portrayed. Arrows indicate the flow of information. To give you an idea of the meaning of this diagram we discuss each of the functions briefly.

Center Parcs Europe (HQ) – Enterprise Functions

Call center Handles bookings from customers, made by either telephone or via the website.

Handling Deals with all incoming bookings from call center, tour operators, and agents. It processes this data and passes it to the Finance function. Subsequently, it

sends a Pre-Arrival Package (PAP) to the customers, containing swimming pool entry passes, brochures, etc. Furthermore, it passes the information on to the Operations function of the respective parks.

Finance Receives information about the bookings that are made from the Handling function, sends invoices to the customers, and receives the payments from them (not if the customer booked via a tour operator). Finance will also calculate and pay the commission to the tour operators and agents. The payments from the bookings are transferred to the individual parks (which are all separate corporations). Finance also reports to the Management function.

Management Receives and reviews financial reports from the Finance functions of both the CP HQ and the individual parks. Planning and feedback on the financial reports are sent to the Finance functions.

CRM The Customer Relationship Management function is responsible for all CRM-related activities. It will receive customer information (i.e., information about the customers, the bookings they made, etc.) and aims to transform this information into knowledge about the customers (i.e., customer profiles and segments, etc.).

Individual parks – Enterprise Functions

Individual Departments Are not further elaborated, since they all perform the same functions from the perspective of enterprise architecture. They receive a planning from the Operations function, and report to the Finance function. They are all individually responsible for the ordering of supply.

Finance Receives payments from the CP HQ, handles invoices from and payments to the suppliers. Reports to the Operations function.

Operations Responsible for all the planning, including that of the different individual departments in that park. They receive reports from the Finance function, and have to report to the management of the CP HQ.

Enterprise functions in an EFD can be decomposed in the same diagram or in a separate EFD. See **Fig. 5** for an example of a decomposition. A tree structure of EFDs can be set up to analyze the architecture of an enterprise. Usually two levels are enough to get sufficient grip on the complexity of an organization. We call the top level EFD a *corporate* EFD (see **Fig. 4**) and a decomposition a *functional* EFD.

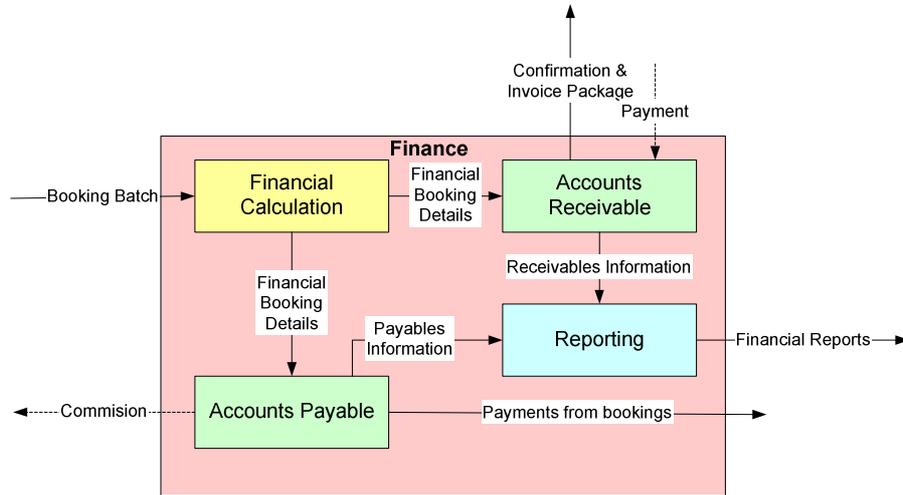


Fig. 5. Decomposition of the Finance function of Center Parcs Europe

The EFD of Finance shows the following subfunctions.

Accounts Receivable Responsible for all incoming payments. For this, it receives booking details from the Financial Calculation function. It sends out booking confirmations and invoices to the customers and collects payments from the customers. Accounts Receivable reports to the Reporting function.

Accounts Payable Responsible for all outgoing payments, mainly commissions to the tour operators and agents, and payments collected from the customers, which are diverted to the parks where the customers booked their stay. Details of the payments that are to be paid are determined by the Financial Calculation function. Accounts Payable reports to the Reporting function.

Financial Calculation Responsible for all computations regarding bookings and commissions. It sends this to the Accounts Receivable and Accounts Payable.

Reporting Receives financial information from Accounts Payable and Accounts Receivable functions, and prepares financial reports that are sent out to the Management function.

3.3 Scenario Overlay (SO)

A scenario is a continuous processing of a request trigger by various enterprise functions, which results in one or more feedback triggers. A Scenario Diagram provides insight in the interoperation of enterprise functions and in the completeness of the EFD. A scenario is drawn on an EFD with a proper explanation. Only essential flows are elaborated into a scenario (highest frequency, large impact). See **Fig. 6** for an example. Red lines are drawn that touch EFD functions in the sequence of the process that is triggered.

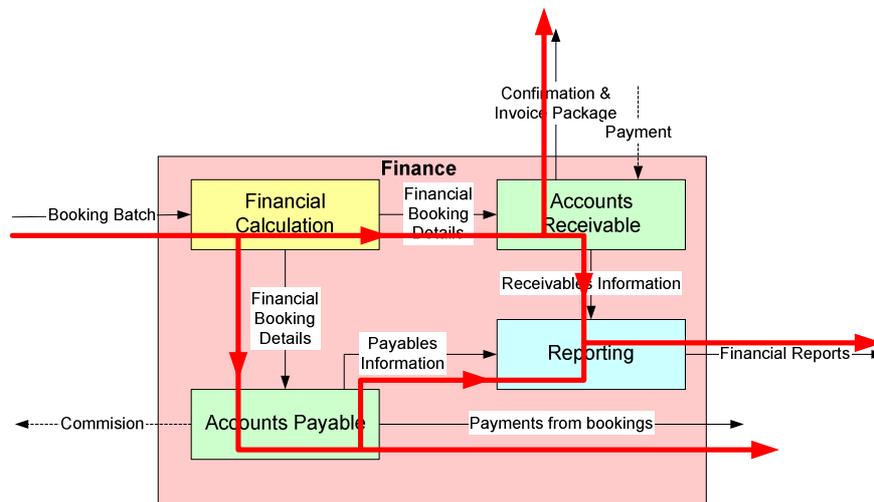


Fig. 6. Example Scenario Overlay, the financial processing of a Booking Batch

Here we show one scenario overlay for the Finance function. This scenario concerns a booking made via the CP call center, so there is no commission. The booking information enters in a batch. The Financial Calculation function determines the exact price of the booking. This information is sent to both the Accounts Receivable and the Accounts Payable functions. The Accounts Receivable sends a confirmation and invoice package to the customer. It sends all information regarding unpaid and paid amounts to the Reporting function. Accounts Payable will divert the payment from the customer for its booking to the respective park, and forwards information about the payments to Reporting. The Reporting function within the Finance function prepares financial reports, which are sent to the Management function.

Scenarios are an optional part of an enterprise architecture description. Typically an EA description will contain several SOs.

3.4 System Infrastructure Diagram (SID)

A SID shows the complete information technology infrastructure of an enterprise, or a well-defined part thereof. A SID shows the main network topology, the main computers that function in the network and the main information systems that run on these computers to support the enterprise functions. A SID contains two types of components. There are executional components (or systems): Applications, Workstations, Servers: data, web, file, applications, Databases, Fire-walls. And there are connectivity components (or Network): Internal Network (LAN, WAN), External Network (private: Leased lines, public: Internet).

For SIDs there are many popular notational variants. An EA description typically contains one SID, corresponding to the abstraction level of the corporate EFD. Complex parts in this SID can be exposed by creating a SID at a more detailed level.

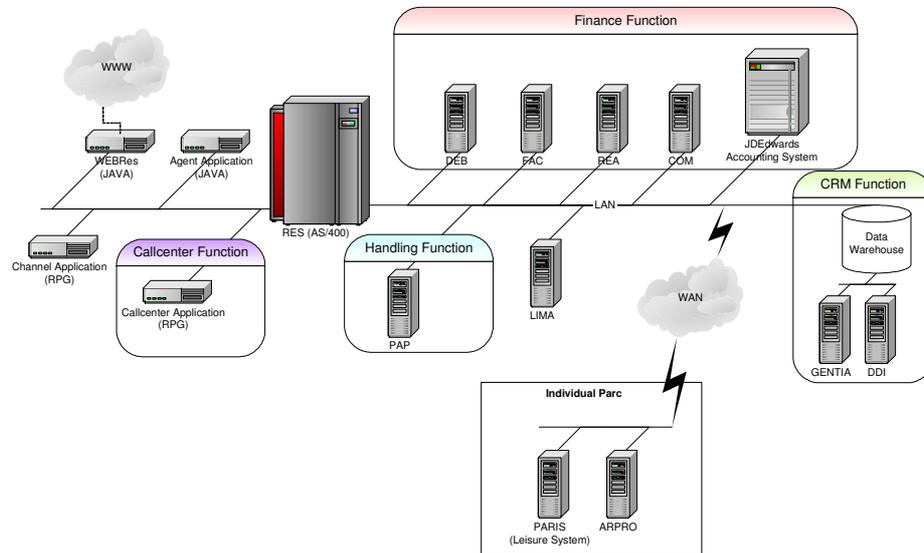


Fig. 7. Example System Infrastructure Diagram, a mixture of platforms at Center Parcs

See **Fig. 7** for an example. The computer icons depict hardware systems, the adjoining acronyms software systems that run on them. The hardware systems are grouped by the enterprise functions they support. The lines depict network connections. The central system depicted in the system infrastructure diagram is the Reservation (RES) system, running on an IBM AS/400 mainframe. On the left side of the RES system, the systems and applications that handle the incoming bookings are shown. Of the systems on the right side of the RES system we mention only briefly the main functions and we do this per business function they belong to:

Handling function The preparation and mailing of the pre-arrival packages.

Finance function Accounts receivable, invoicing, financial reporting, calculation and payment of the commission for the agents and tour operators, Ledger system.

Individual Park Bookings of extra activities in the parks, keep track of information about arriving customers.

CRM function Customer history, central data storage, statistical analysis and data mining system.

As can be seen in the infrastructure diagram, CP uses many different systems. In fact, currently over 200 applications are in use. In the future this will change. The parks use currently about 40 applications in their front office, back office and communications. The headquarters are using about 30 applications. These amounts will respectively change in 14 and 15 applications. Also, The RES system and JDEwards are now accessed by text-based interfaces. For both programs a Graphical User Interface will be implemented.

4 Empirical validation

4.1 Course Enterprise Information Architecture

Each fall a master course on Enterprise Information Architecture (see [17] for a general description) is given as an elective course in the international MSc program of Business Informatics at Utrecht University. An important assignment in this course is to do a case study in practice, meaning that they have to produce a number of Enterprise Architecture Models for a real company. Last year (fall 2004) we had 40 students in 19 groups working at 15 different organizations. This year we had 68 students in 23 groups working at 18 different organizations. Six organizations participated for the second time, making up a total of 27 different organizations.

In Table 1 we list the distribution of the organizations among different categories in the public and private sector according to a standard categorization of companies of the Netherlands Bureau of Statistics [15]. Except for the three categories AFH (Agriculture, Forestry and Hunting), Mining and Education, all categories are represented by at least one company. So this is a pretty good coverage.

Table 1. Organizations per category

Category	Organizations	#
Public Utilities (water, electricity, gas)	Gasunie, Rendo	2
Construction Industry	NCL	1
Transportation, Communication, Shipping	KLM, Schiphol, NorfolkLine	3
Finance	RABO Bank, AAV, HDN	3
Services	Accenture/UniT, AAC Cosmos, BiZZdesign, Conclusion, Exact, GTI, Morgan Chambers, VNO-NCW	8
Public Administration	Dutch Tax Admin., Social Security Admin., Municipalities (2)	4
Health Care	Academic Medical Center	1
Industry	Cisco, Thales, Sanoma	3
Tourism, Recreation and Sport	Center Parcs, NOC*NSF	2

The students are placed in groups of 2 or 3 each and before they visit the companies for the interviews, they get lectures on our EAM technique together with some small exercises. Lectures and exercises take about eight hours.

Besides the interviews, the students only incidentally contact the companies, mainly through email. To keep the interviews as efficient as possible the students prepare themselves by reading publicly available information on the company. For efficiency reasons some companies prepare a special information package for the students.

Even though practical modeling is an important and substantial part of our course, we provide overviews of other important topics, like frameworks or approaches, e.g. Zachman's framework [14][19], ARIS [12][13], TOGAF [15], IEEE Std 1471 [5], '4+1' View Model [8], Enterprise Unified Process [1], SOA [7], MDA [11].

In the case studies we tried to investigate how well the models could be communicated. Even though this is very hard to measure, most people involved (both technical and non-technical) agreed that the models were easy to understand and also

gave a quick impression of the essential functions within a company. Moreover, it turned out that certain characteristics within the diagrams and (in)consistencies between the diagrams could easily be traced and clarified.

4.2 Questionnaire

We launched a questionnaire among the students who performed the case studies. See **Table 2.** for a summary of the results. We had six questions that were asked for each of the diagram types in EAM. The students answered the questions immediately after performing the case study, and with respect to their own work.

Table 2. totals and averages of questionnaire among students (n=23)

Question	Supply Chain Diagram	Enterprise Function Diagram	Scenario Overlay	System Infra-structure Diagram
Is the .. diagram easily readable? Rate on a scale from 1 (very bad readable) to 5 (very good readable) how good the .. diagram readable is.	4,3	3,2	3,7	3,8
Has the .. diagram the right level of abstraction? Make a choice: a. less detail preferred / b. just right the way it is / c. more detail preferred.	a 4% b 74% c 22%	a 9% b 91% c 0%	a 9% b 86% c 5%	a 14% b 54% c 32%
Is the correctness of the .. diagram easily established (conformity with the reality within the company)? Rate on a scale from 1 (very difficult) to 5 (very easy) how well the .. diagram can be checked.	3,5	2,7	3,5	3,4
Is there information lacking on the .. diagram? Chose y (yes, information is lacking) or n (no, no information is lacking).	y 35% n 65%	y 35% n 65%	y 23% n 77%	y 24% n 76%
Is there redundant information in the .. diagram? Chose y (yes, there is redundant information) or n (no, there is no redundant information)	y 4% n 96%	y 9% n 91%	y 5% n 95%	y 10% n 90%
How easy is it to produce this kind of diagram on the basis of available information? Rate on a scale from 1 (very difficult) to 5 (very easy) how well the .. diagram can be produced.	3,7	2,7	3,6	3,3

On the whole we are satisfied by the scores, but there are some points that need attention (with remarks made by students in the questionnaire):

- the readability of the EFD diagram, (too many arrows)
- abstraction level of SCD and SID (SCD more parties, business units, only core business, SID (no remarks))
- the correctness of the EFD, (a lot of experts needed; a lot of outsourced processes)
- information lacking on all diagram types (SCD money flows? Data flows? High abstraction level; EFD only most important functions modeled, no room for company structures, maybe missed something; SO loops, timeframes, departments, maybe missed something, omitted some flows, conditions; SID maybe missed something)
- makeability of EFD (more guidelines, not all companies think in functions, need more information, what to do with exceptions).

To complement this internal evaluation we also put out the questionnaire among the readers of the case study reports at the participating companies, see **Table 3**. The sixth question (how easy to produce) was replaced by a consumer question (how well to use).

Table 3. totals and averages of questionnaire among readers (n=11)

Question	Supply Chain Diagram	Enterprise Function Diagram	Scenario Overlay	System Infra-structure Diagram
Is the .. diagram easily readable? Rate on a scale from 1 (very bad readable) to 5 (very good readable) how good the .. diagram readable is.	4,5	3,7	3,2	3,4
Has the .. diagram the right level of abstraction? Make a choice: a. less detail preferred / b. just right the way it is / c. more detail preferred.	a 0% b 45% c 55%	a 0% b 91% c 9%	a 0% b 64% c 36%	a 0% b 67% c 33%
Is the correctness of the .. diagram easily established (conformity with the reality within the company)? Rate on a scale from 1 (very difficult) to 5 (very easy) how well the .. diagram can be checked.	3,9	3,7	3,3	3,9
Is there information lacking on the .. diagram? Chose y (yes, information is lacking) or n (no, no information is lacking).	y 82% n 18%	y 40% n 60%	y 64% n 36%	y 50% n 50%
Is there redundant information in the .. diagram? Chose y (yes, there is redundant information) or n (no, there is no redundant information)	y 9% n 91%	y 18% n 82%	y 0% n 100%	y 0% n 100%
How well usable are this kind of diagrams in your organisation? Rate on a scale from 1 (totally unusable) to 5 (very well usable) how usable the diagram is.	3,7	3,7	3,2	3,7

The scores are overall comparable with the student scores. The differences are:

- the level of abstraction and the lacking of information of the SCD
- higher scores for EFD
- lower scores for SO
- the readability of the SID

5 Conclusions and Future Work

We conclude that EAM is a modeling method for enterprise architecture that meets our set goals of 'easy to create' and 'easy to read'. EAM can be learned and trained in a short course of one day. The resulting diagrams can be understood without any specific training.

We want to pay attention to the attention points coming out of the questionnaires. To fill a gap that is felt between the Enterprise Function Diagram and the System Infrastructure Diagram we want to introduce a new type of diagram, comparable to the Scenario Overlay, the Application Overlay. It shows which applications give support to which enterprise functions. We want to produce tool support for EAM. Together with a partner from industry we want to develop the application in practice of EAM.

6 Acknowledgements

We thank Ronald Bos and Inge van de Weerd for conducting the case study at CP.

7 References

- [1] Ambler, S.W., Nalbone, J. & Vizdos, M. (2005). The Enterprise Unified Process – Extending the Rational Unified Process. Prentice Hall PTR.
- [2] Van Es, R.M., Post, H.A. (1996), Dynamic Enterprise Modeling, Kluwer.
- [3] EWITA. Enterprise Architecture (accessed December 3, 2005)
http://www.ewita.com/EA_Overview/EADefinition.htm
- [4] EWITA. Enterprise Architecture and Related Definitions (accessed December 3, 2005)
http://www.ewita.com/EA_Overview/Definitions/ArchitectureDefinitions.htm
- [5] IEEE, 2000. IEEE recommended practice for architecture description, IEEE Std 1471.
- [6] Koning, H., Dormann, C., van Vliet, H., (2002). Practical Guidelines for the Readability of IT-architecture Diagrams. In: Proceedings ACM SIGDOC 2002, pp 90-99.
- [7] Krafzig, D., Banke, K. & Slama, D. (2005). Enterprise SOA – Service-Oriented Architecture Best Practices. Prentice Hall PTR.
- [8] Kruchten, P. (1995), Architectural blueprints – The ‘4+1’ View Model of Software Architecture, IEEE Software, 12(6); 42 – 50.
- [9] Lankhorst, M. et al (2005). Enterprise Architecture at Work, Springer-Verlag Berlin.
- [10] Le, L.S., Wegmann, A. SeamCAD 1.x: User’s Guide
<http://infoscience.epfl.ch/getfile.py?mode=best&recid=55774>
- [11] Object Management Group (2003), MDA Guide version 1.0.1, Miller, J. & Mukerji, J (eds.), <http://www.omg.org/docs/omg/03-06-01.pdf>.
- [12] Scheer, A.W. (1998a). ARIS – Business Process Frameworks, Springer-Verlag Berlin.
- [13] Scheer, A.W. (1998b). ARIS – Business Process Modelling, Springer-Verlag Berlin
- [14] Sowa, J.F. & Zachman, J.A. (1992), Extending and formalizing the framework for information systems architecture. IBM Systems Journal, vol 31(3).
- [15] Statistics Netherlands (Centraal Bureau voor de Statistiek): Standaard Bedrijfsindeling 1993 (SBI '93):
<http://www.cbs.nl/nl-NL/menu/methoden/classificaties/overzicht/sbi/default.htm>
- [16] The Open Group (2002), TOGAF "Enterprise Edition" Version 8.1.
<http://www.opengroup.org/architecture/togaf8-doc/arch/>
- [17] University Utrecht, 2005. Course Enterprise Information Architecture,
<http://www.cs.uu.nl/education/vak.php?vak=INFOEIA&jaar=2005>.
- [18] Wegmann, A, 2003. ON THE SYSTEMIC ENTERPRISE ARCHITECTURE METHODOLOGY (SEAM), proceedings ICEIS 2003,
- [19] Zachman, J.A. (1987), A framework for information systems architecture. IBM Systems Journal 26(3); 276 – 292.