

Validation of a Process-driven Database Design Method for an Electronic Health Record-system

From process models to data models

Roy Fischer



University of Groningen
Faculty of Economics & Business

Groningen, January, 2014

Studentenbureau UMCG

Universitair Medisch Centrum Groningen

Validation of a Process-driven Database Design Method for an Electronic Health Record-system

From process models to data models

Groningen, februari 2014

Auteur
Studentnummer

Afstudeerscriptie in het kader van

Begeleider onderwijsinstelling

Roy Fischer
1877844

Electronic Health Record
Technology & Operations Management
Rijksuniversiteit Groningen

dr. H. Balsters
Rijksuniversiteit Groningen

Alle rechten voorbehouden. Niets uit deze uitgave mag worden verveelvoudigd, opgeslagen in een geautomatiseerd gegevensbestand, of openbaar gemaakt, in enige vorm of op enige wijze, hetzij elektronisch, mechanisch, door fotokopieën, opnamen, of enige andere manier, zonder voorafgaande toestemming van de uitgever.

Voor zover het maken van kopieën uit deze uitgave is toegestaan op grond van artikel 16B Auteurswet 1912 j° het Besluit van 20 juni 1974, St.b. 351, zoals gewijzigd in Besluit van 23 augustus 1985, St.b. 471 en artikel 17 Auteurswet 1912, dient men de daarvoor wettelijk verschuldigde vergoedingen te voldoen aan de Stichting Reprorecht. Voor het overnemen van gedeelte(n) uit deze uitgave in bloemlezingen, readers en andere compilatiewerken (artikel 16 Auteurswet 1912) dient men zich tot de uitgever te wenden.

Trefw Electronic Health Record, Database Modelling, ORM, BPMN, FBM, RUG

PREFACE

This Master thesis is based on the final project of my study Technology and Operations Management at the Rijksuniversiteit Groningen. It was an interesting but challenging project. With the support and help of the large teaching hospital in the Netherlands where the research was conducted, this thesis was brought to life. Next to this, I want to thank my fellow students in the project (Sjoerd-Gerrit Hoekstra and Erik Spits) for their support and active cooperation in the project.

Finally, I want to thank my supervisor dr. H. Balsters for his active cooperation, guidance and support in this project and the co-assessor prof. dr. J. de Vries for assessing my thesis.

TABLE OF CONTENTS

ABSTRACT	1
1 INTRODUCTION	3
1.1 SCIENTIFIC VALUE	4
2 THEORETICAL BACKGROUND	5
2.1 REQUIREMENTS ENGINEERING.....	5
2.2 BUSINESS PROCESS MODELLING	5
2.3 DATA MODELLING	6
2.3.1 ER-Modelling.....	6
2.3.2 Object-oriented Modelling	7
2.3.3 Fact-based Modelling.....	7
2.4 PROCESS DRIVEN DATABASE DESIGN	8
2.4.1 Process modelling	9
2.4.2 Database modelling.....	9
3 PROPOSED METHODOLOGY	11
3.1 OVERALL RESEARCH FRAMEWORK.....	11
3.2 OVERALL PROJECT METHODOLOGY	12
3.2.1 Building the process models	12
3.2.2 From process models to data models	12
3.2.3 Validating the data models	12
3.3 INTERNAL METHODOLOGY.....	13
3.3.1 Step 1: Get an understanding of the Universe of Discourse	13
3.3.2 Step 2: Design FBM models according to preliminary process models.....	13
3.3.3 Step 3: Designing final FBM data models	13
3.3.4 Step 4: Critically examining the steps of the FB-BPM method.....	13
4 RESULTS	15
4.1 STEP 1: GET AN UNDERSTANDING OF THE UNIVERSE OF DISCOURSE	15
4.2 STEP 2: DESIGN FBM MODELS ACCORDING TO PRELIMINARY PROCESS MODELS	15
4.3 STEP 3: DESIGNING FINAL FBM DATA MODELS	15
4.3.1 Forming data models from BPMN in practice.....	16
4.4 STEP 4: CRITICALLY EXAMINING THE STEPS OF THE FB-BPM METHOD.....	20
5 DISCUSSION	23

6	CONCLUSION	25
6.1	SCIENTIFIC CONTRIBUTION.....	25
7	REFERENCES	27
8	APPENDIX	29
8.1	THEME CASE DESCRIPTION	29
8.2	NAMING IN THE MODELS	31

ABSTRACT

In designing a system like an Electronic Health Record (EHR), there is a proliferation of methods and best practices. Bitter truth is that many efforts to design such methods fail. Practice all too often indicates that there is no systematic link between a *business process* and the associated *business data*. This calls for a systematic approach for designing databases in a way in which the users are actively involved and their (business) processes are leading. That said, the business data, modelled in a database, can be derived from the processes. This method will be coined as Fact-Based Business Process Modelling (FB-BPM). This method is tested and validated in the design of an EHR at a large teaching hospital in the Netherlands (LTHN). The results of this research show that the method has been validated for complex business processes. It has also been proven that the method is validated in the context of an EHR.

1 INTRODUCTION

Healthcare systems are on an edge; an edge on which every single medical provider needs to change their way of doing, because the current way is obsolete. Government policies demand that hospitals specialize in best practices, and leave the rest to other hospitals (Modderkolk 2013). This means that patients have to travel more frequently from one hospital to the other. How convenient would it be that their health record, the record where all their medical history is stored, would travel with them. That is what can be achieved with a so-called Electronic Health Record (EHR), which is a digital system that records all data concerning treatments of a patient. An internal EHR system of a hospital can be the starting point for collaborating EHR's. There are however some drawbacks.

The Dutch healthcare needs to specialize in treatments in which they excel; at least, that is what minister of health, welfare and sports, Edith Schippers, is aiming for. She states that hospitals should be selective in what care to offer. A hospital offering every treatment at the expense of everything is a preacher of inefficiency (Van Dorrestijn 2012). This pleads for a national so-called Health Care Information Infrastructure (HCII), where patient records can be exchanged digitally between medical centres through EHRs. In 2011, a first attempt to a national EHR was rejected by the Dutch First Chamber due to concerns about privacy violations and the safety of the ICT system (Modderkolk 2013). From that moment on, several private initiatives have been founded to create patient records. A collaboration of healthcare providers has started an EHR under the name Landelijk Schakelpunt (LSP). The LSP lets, e.g., general practitioners and pharmacists exchange patient records under the condition that the patient has granted permission to do so (Vereniging Zorgaanbieders voor Zorgcommunicatie 2012). This system, however, is not yet successful (Modderkolk 2013). Practice has revealed that the failures of such systems could be caused by the method through which they have been designed.

In designing a system like an EHR, there is a proliferation of methods and best practices. Bitter truth is that many efforts to design such methods fail. Very often, a company has an idea of the to-be designed database. But practice all too often indicates that there is no systematic

link between a *business process* and the associated *business data* (Balsters 2013b). And in even more cases, no method is used at all. This calls for a systematic approach to designing databases in a way in which the users are actively involved and their (business) processes are leading. The business data, modelled in a database, can be derived from the processes (Balsters 2013b). The method described by Balsters (2013) will be coined as Fact-Based Business Process Modelling (FB-BPM). In this research this method will be validated in the design of an EHR at a large teaching hospital in the Netherlands (LTHN).

At the LTHN a large migration project is in progress, migrating some 50 source systems into one EHR. This is a comprehensive project, as these systems depend largely on each other, and the size of the data is enormous. Since these source systems are starting to decay, there is a need for mapping the business processes at the LTHN into a new system called the New EHR.

In 2010, a special taskforce has been brought to life to make this happen, and a cooperation with another large Dutch hospital has been established. Together, they now prepare a functional design to offer to the ICT company, who will – based on this functional design – implement the system. For this reasons, the taskforce has already made a proposal for the data model for the target database system, the Logisch Bedrijfs Gegevens Model (LBGM) and Technisch Bedrijfs Gegevens Model (TBGM). The LTHN wants to validate the correctness of this model; a research task being conducted by three master students of the RUG. From a business perspective, the stakeholders and their demands have to be incorporated into critical success factors for the EHR-system. This will be done in three stages, and due to limitations in time and designated personnel, the researched target system will be confined to that part of the system dealing with patient admittance in the hospital (i.e. the triage process).

The first stage is the mapping of the processes in a so-called Business Process Modelling Notation (BPMN), performed by Hoekstra (2014). This is a first step in documenting all steps that are associated with the admittance of a patient, together with the questions belonging to the method. From here, the BPMN models are transformed into Data

Models (specified in Fact-Based Modelling), which are essentially the basis of the EHR system. This will be elaborated on in this paper (Fischer, 2014). The final step is to validate the data models by the end user by creating user interfaces based on the models and creating an iterative loop to the earlier stages. Next to this, the proposed method is validated by critically examining its steps. The LTHN LBGm/TBGM model is validated by comparing the data models from this research with the LBGm/TBGM of the LTHN. This is done by Spits (2014). The proof of concept of this method has been established by a project in 2013 (Stephana 2013; Hijlkema 2013; Post 2013). This is a similar research project performed at another Dutch hospital. Here it has been proved that the method works for a relatively simple situation, but not in a complex situation of a large teaching hospital. The *overall research* will therefore be driven by the following research question:

How to validate a proposed general method of process-driven database design in the context of designing a database supporting an EHR-system?

The *part* of the research described in this thesis will be driven by the following question:

How can data models be systematically derived from process models in the context of designing a database supporting an EHR-system?

This question will be answered by the following sub-questions:

- *How can the FB-BPM method be used for a systematic derivation of data models from process models in the context of designing a database supporting an EHR-system at a LTHN?*
- *Are the steps from the FB-BPM method accurate and how can they be improved?*

1.1 SCIENTIFIC VALUE

For this research, a LTHN will be used to validate the proposed method. This is further explained in the methodology section. When the questions are answered, the scientific contents of the *overall research* (Hoekstra 2014; Fischer 2014; Spits 2014) will contain several components. The first is that the general usability of the method in the context of an EHR design will be determined. More specifically, is the method applicable for other cases of EHR design within different hospitals? Secondly, the scientific value of *this thesis* is the following. The correctness of the FB-BPM method will be determined by critically examining the proposed steps and improving them where possible. In this way, an addition can be made to the current scientific literature.

The research will start with a theoretical background of the proposed method, given in chapter 2. This is divided in paragraphs according to the different theoretical domains of the model. After this, in chapter 3, the proposed methodology will be discussed. The results will be shown in chapter 4, followed by the discussion and conclusion in chapter 5 and 6.

2 THEORETICAL BACKGROUND

In this chapter, the background of literature needed to understand the proposed method will be explained. This will be done by first examining research being done on Requirements Engineering, since the functional design of an EHR-system is first and foremost about getting the requirements of the to-be designed system right, before implementing the system. In the second and third place, we will examine process modelling and data modelling, and in that specific order. The idea is that a particular method of requirements engineering, coined as FB-BPM, will be suited to construct the functional design of an EHR.

2.1 REQUIREMENTS ENGINEERING

As said before, the functional design of an EHR-system is first and foremost about getting the requirements of the system right, before implementing it. This is encapsulated in so-called requirements engineering (RE). Requirements engineering is introduced as a way of conceptual modelling which improves the quality of a software (database) production process. The aim of this approach is to capture software requirements providing some methods and techniques and to supply a roadmap to move from these requirements to a conceptual schema in a traceable way (Insfrán et al. 2002). The RE method as described by Insfrán et al. (2002) globally consists of the introduction of a Requirements Model (RM) in which all functional information for the conceptual schema is collected, and the Requirements Analysis Process (RAP) which provides guidance in building a conceptual schema based on the functional requirements. In the RM phase, a mission statement is defined and data use cases are developed. The latter are tables which contain interactions between the system and system users. The RAP phase consists of a step-by-step walk through the use-case descriptions and designing the software accordingly. In this way, when the user requirements change, these changes can be traced to the end product more efficiently. According to Selvakumar & Rajaram (2011), the requirements of a system consist of functional (FR) and non-functional re-

quirements (NFR). The FR deal with the functionality of the system whereas the NFR deal with constraints to the system. The first are characterized by *what* has to be in the system, described in simple language. The latter are characterized by *how* the system should become in e.g. expensiveness, response time, reliability, scalability etc. (Selvakumar & Rajaram 2011)

When we aggregate this knowledge to a higher level, we can conclude that first some modelling of the requirements has to take place and subsequently these requirements have to be transformed into a conceptual schema. We can state that for the design of a to-be system it is utmost important to take the stakeholder requirements as a point of departure. The following techniques could bring guidance to this process.

2.2 BUSINESS PROCESS MODELLING

Use-case descriptions and documentation of complex procedures are often difficult to understand and error prone. For this reason, a clear picture depicting a workflow or business process is often used to convey the intended meaning of the process (Chinosi & Trombetta 2012).

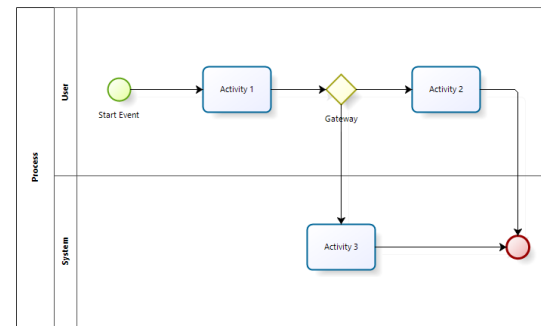


Figure 1 Example Business Process Model

A business process is a collection of subsequent tasks that a business uses in performing their work. A business process model describes tasks and the ordering of these tasks: what work is performed, when is it performed and who performs the task (Bridgeland & Zahavi 2009). As a standard for specifying these business processes, the Business Process Modelling Notation is often used, providing a graphical representation based on workflow diagramming. This means that BPMN models can easily be validated with a non-technical domain expert and end-users of a to-be system. It constitutes an internationally accepted (ISO-) standard for modelling business processes (Balsters 2013b). A short summary of BPMN and its assets from Bridgeland & Zahavi (2009) will follow next.

A BPMN model consists primarily of activities. An activity is a discrete chunk of work, something with a beginning and an end, that is performed one or more times. Every activity has attributes; a description of the activity to give more detail about how it is performed. Activities are connected by sequence flows, showing that one activity is performed before the other. This is shown as a solid line with an arrow between activities.

Figure 1 shows an example of a BPM. Activities occur in a so-called 'swim-lane', which graphically shows who performs which activities. Each process starts with a 'start event' and can end with an 'end event'. If there are alternatives to a certain sequence flow, a gateway is used. This is depicted as a diamond shape and multiple sequence flows exit a gateway. The actual sequence flow taken depends on the condition modelled in the gateway.

So far, the basic notation of BPMN has been explained. This is what is necessary for a good understanding of this research paper. Because of its international acceptance and ease of validation with stakeholders, BPMN forms the basis for the FB-BPMN method validated in this research. In this method, BPMN models are systematically transformed into data models which is explained next.

2.3 DATA MODELLING

When one aims to create a database for a particular business domain, one creates a model for it. The to be modelled business domain is then called the universe of discourse (UoD), which is a part of the 'real world'. The best

way to arrive at a clear description of the UoD is to use natural language, intuitive diagrams and examples. Favourably one fact at a time. (Halpin & Morgan 2008). Many methods have been developed for obtaining this goal, one more successful than the other. The previously mentioned authors have come up with a method which fulfils the 'natural language, intuitive diagrams and examples' demands in an excellent manner. Before this will be further elaborated, other methods will be briefly explained. We can globally distinguish three different data modelling approaches: Entity-Relationship modelling (ER), fact-oriented modelling and object oriented modelling (Halpin & Morgan 2008). These different modelling approaches will be discussed next.

2.3.1 ER-MODELLING

Entity-Relationship modelling is one of the most widely used approaches for data modelling. It was introduced in 1976 and ever since, it has evolved from a basic language with flaws, to a state where most deficiencies are cured and the language is precise enough for some modelling purposes (Patig 2006).

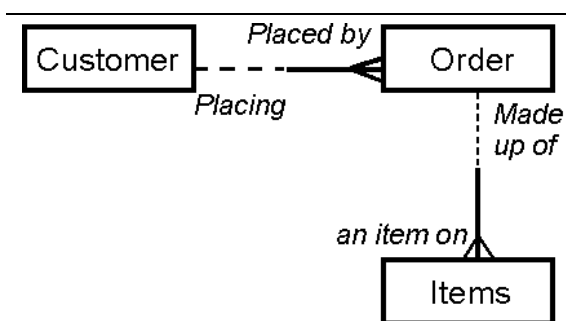


Figure 2 UML Class Diagram (Halpin & Morgan 2008)

As an example, the popular ER notation from CASE tools from the Oracle Corporation is described here (Halpin & Morgan 2008). The entity types are shown as named rectangles and attributes can be placed inside of these rectangles. Relationships are shown as named lines where a broken line is optional. The 'crow's foot' indicates that many instances of the entity type can be related to the with the same entity instance on the other end (Halpin &

Morgan 2008). For example: an order must *be placed by one and only one* customer, but a customer can place multiple orders.

The ER notation is nevertheless incomplete. For instance, a mandatory role cannot be modelled in this notation. Next to this, moving from the business processes to the model is not obvious, the modelling notation is too simple to have all possible constraints that can occur and it is very hard to validate the designed models, because of the lack of readability for end-users. (Halpin & Morgan 2008). Next to this, Halpin (2001) states that ER diagrams are far away from natural language, lack the expressibility and simplicity of a role-based notation for constraints and require complex design choices about attributes (Halpin 2001).

2.3.2 OBJECT-ORIENTED MODELLING

Another widely used modelling approach is Object-Oriented modelling, with its most influential approach Unified Modelling Language (UML). UML includes class diagrams to specify static data structures and encapsulates both data and behaviour within objects (Halpin & Morgan 2008).

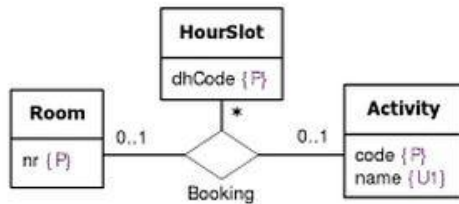


Figure 3 ER diagram from CASE tools (<http://cisnet.baruch.cuny.edu/holowczak/classes/9440/entityrelationship/>)

Because of its object-oriented focus, the classes don't require conceptual identification schemes. Entities are rather identified by internal object identifiers.

Figure 2 represents a UML class diagram. It can be seen that such a model is hard to validate with the domain expert. The reason for this is that UML association roles are not ordered. So formally, we cannot know if the sentence

should be read as "Room at HourSlot is Booked for Activity" or "Activity at HourSlot is Booked for Room". This can get even worse if the same class plays more than one role in this association. Next to this, identifiers of entities don't contain keys to determine which is a primary identifier. And, UML was never meant to model databases in the first place. (Halpin & Morgan 2008) As this modelling language is used and acknowledged worldwide, it cannot be overseen. As Doesburg and Balsters (2012) state: "The Unified Modelling Language (UML) is the lingua franca in current software engineering practice, and UML class diagrams are used for data modelling within software-engineering projects" (Doesburg & Balsters 2012).

2.3.3 FACT-BASED MODELLING

What can be made up from the above paragraphs, is that these methods both lack in model support validation with the domain expert and end-users, and are difficult to derive from process models. This said, there is a need for a modelling language that does not have these deficiencies. Fact-based modelling (FBM) overcomes these problems and is explained next.

The fact-based modelling language which is focussed on here is Object-Role Modelling (ORM). ORM views the world in terms of *objects* playing *roles* (Halpin & Morgan 2008). In contrast to the other modelling languages, in this language facts and rules may be verbalized in an even for non-technical domain experts understandable language. In contrast to Entity-Relationship (ER) modelling and Unified Modelling Language (UML), ORM treats all facts as relationships and its models are attribute-free (Balsters 2013b). As can be seen in figure 4, the model is unambiguous and requires only little knowledge of the domain expert to understand the model. Next to this, ORM models are more stable under a changing business domain and often capture more business rules in diagram form (Halpin & Morgan 2008). Fact-based modelling is the general name for alterations on the fact-based conceptual data modelling like ORM, Natural Information Analysis Method (NIAM) and Fully-Communication Oriented Information Modelling (FCO-IM) (Balsters 2013b).

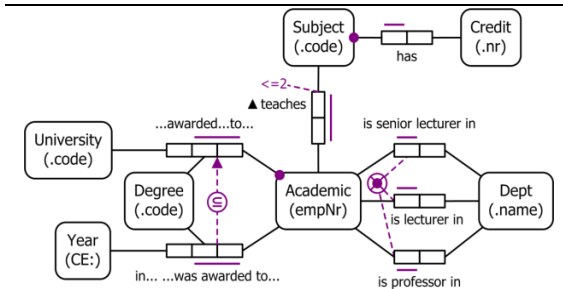


Figure 4 Example ORM model
(<http://www.orm.net/overview.html>)

Typically, the design of an ORM model is done according to the conceptual schema design procedure (CSDP). This contains the following seven steps (Halpin 2001):

- 1 Transform familiar information examples into elementary facts, and apply quick checks
- 2 Draw a draft diagram of the fact types and apply a population check
- 3 Check for entity types that should be combined, and note any arithmetic derivations
- 4 Add uniqueness constraints, and check arity of fact types
- 5 Add mandatory role constraints, and check for logical derivations
- 6 Add any value, set comparison, and subtyping constraints
- 7 Add other constraint and perform final checks.

Because fact-based modelling offers these advantages against the other languages, this modelling language will be used in this thesis and the exact method on how this language is used will be explained in chapter 2.4.2. However, as UML is still leading in database modelling, there is need for a translation between FBM (ORM) and UML. Database engineers that have specified their data schemas in FBM, are often faced with difficulties in communicating these schemas to software engineers using UML. The paper of

Doesburg and Balsters (2012) describes an FBM-based specification of a data-modelling kernel of the UML Superstructure. This kernel is fact-based, with the added advantage of enabling validation of this FBM-specification (Doesburg & Balsters 2012). This implies that if there is an end-user need for modelling in UML, then these can be generated using this research as can be seen in figure 5.

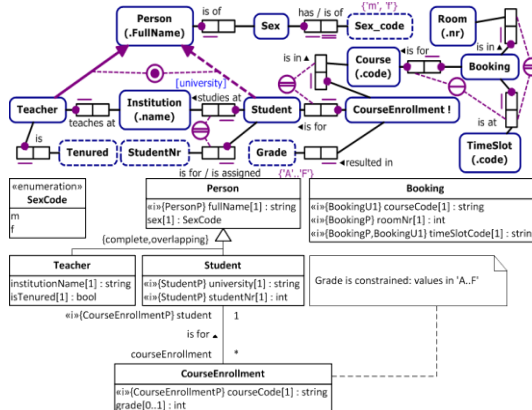


Figure 5 ORM to UML translation (Doesburg & Balsters 2012)

In the case it is necessary or convenient to have UML diagrams as an end product, without it's downsides in designing them from process models, it is possible to translate these from ORM.

As for now, the foundation has been laid out for the proposed method. The knowledge on the requirements engineering, process modelling and fact-oriented modelling congregates in the Fact-Based Business Process Modelling method by Balsters (2013). This method will be extensively described next.

2.4 PROCESS DRIVEN DATABASE DESIGN

Fact is that the previously described research domains (BPM, FBM, RE) exist primarily next to each other. The process-driven database design method that is used is the FB-BPM method and it essentially incorporates all three

research domains. The FB-BPM method starts from validated process models (capturing the intended usage of the system from the perspective of each stakeholder), and in a semi-automatic mode derives a data model from such a process model. By capturing both the relevant process- and data-properties of the system, the claim is that FB-BPM offers a full treatment –on the conceptual level- of the functional requirements needed for an EHR-system.

2.4.1 PROCESS MODELLING

Knowledge from RE learns that the perspective of the user, or stakeholder, needs to be taken as a starting point for the to be developed system. In this way, his data-use processes (DUP) and structured interviews are the basis for a step-by-step construction of a data model, supporting these scenarios. The DUP are pieced together in a BPMN notation, containing the whole business process aimed for in the project. This step starts with preparing and conducting interviews for each data-use stakeholder. From this information, the relevant DUP are transformed to conceptual process models. Hereafter, the models will be validated to the same stakeholders, resulting in validated process models. This step in the FB-BPM method runs somewhat simultaneously with the next, which is the transforming of the BPMN model to a fact-oriented data model.

2.4.2 DATABASE MODELLING

For each BPMN task, an ORM event will be made. A BPMN-task of for instance <Set: Temperature> will be translated to an ORM-event: [Temperature Setting: is logged]. The general format describing a BPMN task is: <Verb-phrase present tense: Noun-phrase>. The general format describing an ORM event is: [Noun-phrase Nominalized Verb-phrase: is logged]. The nominalization of the verb is in this case the change of a verb into a corresponding noun phrase. The in the event mentioned ‘is logged’ refers to the time stamping of that event. (Balsters 2013b)

To arrive at a certain data model fragment from a BPM event, we can ask some **fact-type identifying** questions (Balsters 2013b):

- Which entities are involved in the event as *participants*?
- At what *instant* (timestamp) does the event happen?

- How do we *identify* the event?
- What do we have as *input* for the event?
- What do we have as *output* for the event?

For answering these questions, domain experts from the company have to be involved. However, much information can already be drawn from the interviews held in the first part of the whole project. With the answers of these questions, we can then form an ORM data model event by event. To indicate that one event (e.g. Event2) is preceded by another (e.g. Event1) we need to write a rule (written in OLE: ORM-Logic driven English (Balsters 2012)):

for each Event1, Event2, Instant1, **and** Instant2:
if that Event1 has successor **that** Event2 **and that** Event1 is at **that** Instant1 **and that** Event2 is at **that** Instant2
then that Instant1 <**that** Instant2

Moreover, a more general view on the method which will be validated in this thesis can be formulated as the following roadmap (Balsters 2013b):

1. Transform a BPMN task into a desired ORM-event
2. Find a minimal model that realizes that event using our **fact-type identifying** questions
3. Transform the next BPMN task into a subsequent ORM-event
4. Find the minimal extension to the previous ORM model that defines that subsequent ORM-event
5. Repeat 1-4 until all events for all data-stakeholders are finished
6. At the end you will have created the complete corporate database, associated to the original business process

When this step is done, and the complete corporate database has been created, this corporate database can be validated with the end users.

3 PROPOSED METHODOLOGY

The research described in this thesis is a case study performed at a large teaching hospital in the Netherlands. This methodology section is divided in the research of the *overall project*, performed by Hoekstra (2014), Fischer (2014) and Spits (2014), and the *internal methodology* of this thesis. The aim of the *overall research* is to design a conceptual database from the process models and validating these models with the end-user. Next to this, the FB-BPM method that is used is validated by critically examining the proposed steps that should be taken according to that method and improving them where possible. The *internal methodology* consists therefore of the part where process models are transformed into data models using the FB-BPM method while also critically examining the proposed steps.

A proper method to design an EHR is desired by many stakeholders, but current methods lack successful results. This is typical for practical-knowledge problems, where there is a gap between how stakeholders currently perceive the world and how they would like to see the world. Stakeholders perceive the world as being incomplete by the lack of a proper method, and would like to see a proper method with which a successful database supporting an EHR can be designed. This is aimed at by design science (Balsters 2013a). The research is therefore structured according to design science, with requirements engineering as most important asset. This will be explained next.

3.1 OVERALL RESEARCH FRAMEWORK

A convenient starting point of the research is the regulative or engineering cycle from the requirements engineering domain, to fill the gap between theory and practice. This is described by the regulative cycle by van Strien (1997) and by the engineering cycle of Wieringa & Heerkens (2007). The overall project is structured according to the regulative cycle which is explained next.

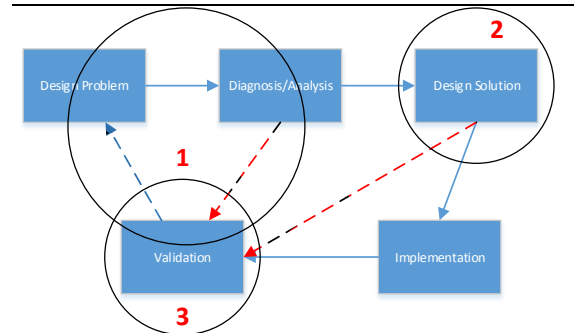


Figure 6 Adapted Regulative Cycle (van Strien 1997)

The regulative cycle is a looped process meaning that the validation can be linked back to the design problem (figure 6). The diagnosis/analysis and design solution are validated intermediary in the research. And as implementation is not part of our research due to time constraints, it is left out. Instead of validating the implementation, our research focuses on validating the diagnosis/analysis and design solution stages and improving the preceding stages with this information. The numbers indicate the projects of the different researches (Hoekstra (1), Fischer (2) and Spits (3)). The second stage in the regulative cycle answer the following questions, that can be used to structure the research (Balsters 2013a):

Design solution:

- Which solution alternatives are available?
- Can we assemble old solutions to build a new solution?
- Can (and must) we invent a new solution completely from scratch?

The following section has been divided in the methodology of the overall project and the internal methodology of the research described in this thesis. This is structured according to the previously mentioned regulative cycle.

3.2 OVERALL PROJECT METHODOLOGY

Here, the methodology of the *overall research project* is described. The combined research of Hoekstra (2014), Fischer (2014) and Spits (2014) will be performed according to the following steps.

3.2.1 BUILDING THE PROCESS MODELS

This first step will be performed by Hoekstra (2014) and is part of the design problem and diagnosis/analysis stages. Information will be systematically gathered from the stakeholders of the to-be designed database using structured interviews and current process schemes. The stakeholders technical and social CSF's and goals are hereby determined. The technical CSF's are of importance for the design of the database models and the social CSF's are of importance for the 'look and feel' of the to-be designed system. Here are also the functional and non-functional requirements of the CSF's of importance, as mentioned in chapter 2.1. This is therefore important when designing user interface mock-ups (chapter 3.2.3). The interviews will be focussed on the business processes of the stakeholders concerning the EHR. This information will then be transformed to business process models using BPMN (Business Process Modelling Notation). Systems thinking can be used for designing these models. This approach offers the ability to abbreviate processes in the form of sub-processes. In order to keep the models readable, sub-processes can be put into so-called black boxes which can be opened and the interactions and functions of those sub-processes can be studied (In 't Veld et al. 2011).

Next to this, the Fact Type Identifying Questions (chapter 2.3.2) are asked to the end user during the interviews, to receive as much information possible for the design of the data models which follows hereafter.

3.2.2 FROM PROCESS MODELS TO DATA MODELS

Once the process models of patient data are realized, the underlying data models supporting the process models can be designed, which will be performed by Fischer (2014). This is part of the design solution phase. The data models, which are the backbone of the entire EHR-system, will essentially be given context by the process models. The method that will be used is the FB-BPM method which is

developed by Balsters (2013). As explained before, this offers a consistent and complete method to translate all process steps into a corresponding data model. The modelling language used will be the fact-based language ORM. The outcome will be a database design which can then be validated with the end-user and which can be compared with the LBGm/TBGM designed by the LTHN. Possible changes in the models proposed by this feedback will also be incorporated into the models. Along the way, the steps of the FB-BPM method are critically examined. The exact methodology of this research will be explained further in the chapter 'Internal Methodology'.

3.2.3 VALIDATING THE DATA MODELS

Added to the FB-BPM method is a validation step. This final part is the validation of both the data models with the end user and the validation of the used method, by comparing the outcomes with the LBGm. This is part of the validation stage of van Strien (1997). The first part entails the designing of user interface mock-ups, based on the BPMN models and the FBM models. These are sketches of to-be user interface screens, based on the data model. The process models define the sequence of the 'screens' whereas the data model defines the content of the 'screens'. These will be validated with the end-user to see if the designed database models reflect the data use processes. The social CSF's of this end-user (who mostly is a non-technical domain expert) are checked if they are met by the system. The final part entails the comparison of the database models with the database models from the LBGm, to validate the outcomes of the method that is used in this whole project, with an database designed according to a different method. This stage will be performed by Spits (2014).

3.3 INTERNAL METHODOLOGY

In this chapter, the steps taken in this part of the *overall research* will be explained step by step. This is derived from the FB-BPM method. The research is structured according to the engineering cycle by Wieringa & Heerkens and entails the following stages:

- *Research problem investigation*: What is it we don't know and why do we want to know it? What are the research questions? (Chapter 1)
- *Research design*: How do we collect data and how do we analyse it? (Chapter 2.4 & 0)
- *Research design validation*: If we would perform the research as designed, would our conclusions be valid? (Chapter 2.4)
- *Do the research* (Chapter 3)
- *Evaluate the results*: What are the answers to the research questions? Is this a significant addition to our knowledge? Are there further questions to be answered? (Chapter 4,5 and Spits (2014))

The steps that will be taken to do the research and obtain and evaluate the results are mentioned next.

3.3.1 STEP 1: GET AN UNDERSTANDING OF THE UNIVERSE OF DISCOURSE

To make it possible to start working from the beginning of semester 1.2, the interviews taken for developing the BPMN models are visited. In this way, a first understanding of the UoD is given. This gives context to the to-be designed fact-based data model.

3.3.2 STEP 2: DESIGN FBM MODELS ACCORDING TO PRELIMINARY PROCESS MODELS

After the design of the first process models by Hoekstra (2014), these can be interpreted and transformed into FBM diagrams. This implies (1) specifying the associated ORM

event to a BPMN task and (2) finding the minimal model that realizes the desired ORM event using our fact-type identification questions. These steps will be repeated until all BPMN models for all data-stakeholders are finished. This is iterative, meaning as soon as new or improved process models are finished, the FBM diagrams are updated with the new information.

As a consequence of the short timeframe in which this research has to be conducted, not all aspects of the method can be considered. In Balsters (2013), every ORM model is accompanied by a rule in OLE (ORM-Logic driven English (Balsters 2012)) indicating that one event is preceded by the next. Because of the necessity of learning this language before being able to build such rules, this was left out of the research. This implies that for an implementation of the model these rules have to be added.

3.3.3 STEP 3: DESIGNING FINAL FBM DATA MODELS

The FBM diagrams and their relation to each other are modelled into one FBM model. This is the complete business database containing FBM diagrams. This results in a complete corporate database for the target processes.

3.3.4 STEP 4: CRITICALLY EXAMINING THE STEPS OF THE FB-BPM METHOD

This step implies critically examining the steps that were taken to reach the final database. It runs parallel to the previous steps. Questions can be asked e.g.:

- Are there steps missing?
- Do all the steps contribute in reaching the end product?

Next to this, the end-user of the to-be designed system will validate the method by giving his opinion on the end product which is done by Spits (2014). A non-satisfactory end product could mean a deficiency of the method.

4 RESULTS

In this chapter, the results of the internal methodology of the research are explained. As the research was conducted according to the four previously mentioned steps, these steps will be used for explanation of the results. As a guidance for this research, the sub-questions that were used will be answered with these steps. In these subparagraphs, difficulties and abnormalities that were encountered in practice are also mentioned.

4.1 STEP 1: GET AN UNDERSTANDING OF THE UNIVERSE OF DISCOURSE

In this step, the interviews that were conducted with the end-users were visited. This gave a first and clear understanding of the UoD. As the first process models were designed, the next step could be taken. The following overall business process was found in this step.

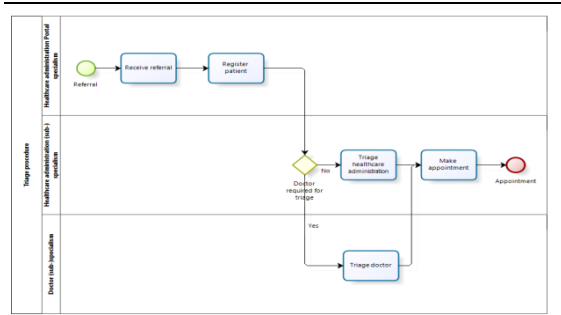


Figure 7 Triage Procedure (Hoekstra 2014)

4.2 STEP 2: DESIGN FBM MODELS ACCORDING TO PRELIMINARY PROCESS MODELS

This is an intermediary step which brought guidance to the research. The intermediary models are therefore not reported and the final models, which are most important, are shown in the next paragraph.

4.3 STEP 3: DESIGNING FINAL FBM DATA MODELS

In designing the first FBM models, the FB-BPM method was used. For every activity that was present in the BPMN model, the fact-type identifying questions were asked. There is a certain overlap in this part, as in Hoekstra (2014) answers to these questions (Balsters 2013b) had to be found and in this thesis they are used for designing the FBM models.

- Which entities are involved in the event as participants?
- At what instant (timestamp) does the event happen?
- How do we identify the event?
- What do we have as input for the event?
- What do we have as output for the event?
- Which data is needed for the event? (Hoekstra, 2014)
- Which entities are involved in the event as participant? (Hoekstra, 2014)

As the first interviews were conducted with end-users who are non-technical domain experts, not all of these questions could be answered by them. This requires knowledge from a technical domain expert. But, practice showed that even a technical domain expert does not have a good understanding on how these questions should be interpreted in practice.

For instance, the entities that are involved in the event as participants can be explained as: 'what is being recorded in each event?'. Or more specifically, 'what do you want to be recorded in your database in each event?'. The second question is a system action. This requires no need for a domain expert input, as for the EHR, all events have to be time stamped to attain a certain 'audit function' of the EHR. The third question can be discussed with a technical domain expert. This concerns questions such as: 'with what other events can one event be identified?'. Practice shows that these are often debatable questions which render precious information for the company, because many of them were never asked before. The fourth and fifth question answer what values go into the event and what the result is of the event. This could be e.g. that input is a correct Spe-

cialtyKind of the account (e.g. healthcare administration specialism) and the output is that there is a digital referral. This output determines what the next event in the sequence will be. Simplifying the questions made it clearer for both technical and non-technical domain experts. The questions that often could not be answered in the interviews of Hoekstra (2014) are as follows, and were answered in this research or were discussed by a technical domain expert. Further questions ORM (if they are not answered in the BPMN part)

- Additional Question: How can the participants be identified (Hoekstra, 2014)?

The method gave information on how an event should be identified, but not on how participants should be identified.

- At what instant (timestamp) does the event happen?
- How can the event be identified?

With the answers to all the previous questions in combination with the BPMN models, the first FBM models were designed. The substantive question ORM were answered during the design of the ORM diagrams, when they were not already answered in the BPMN interviews. Questions which could not be answered were discussed with the technical domain expert.

For the ORM diagram design the six steps roadmap mentioned in 2.3.2 was used initially. However, practice showed that these steps suffice for simple BPMN models with a pool with a single swim lane, while the BPMN model of the Triage Process of the LTHN as designed by Hoekstra (2014) is more complicated. As this model is distinctive for processes within a hospital environment where patient data needs to be recorded, a more elaborate roadmap needs to be designed. An example of this problem is given in the next section.

4.3.1 FORMING DATA MODELS FROM BPMN IN PRACTICE

As the BPMN model representing the triage process at the LTHN is complex and some similar activities can take place in different swim lanes, it is of importance to start with determining where the BPMN activities need to be in the dif-

ferent to-be FBM models. The contents of every ORM diagram based on a part of the BPMN diagram is based on every activity in BPMN up to a final activity. A final activity is an activity which is followed by a gateway and at the same time an activity to which all flows lead from the first gateway which started the diagram. If a gateway diverges into different swim-lanes, the diagram proceeds until the flows converge into the so called final activity. The BPMN where this is based on is shown in appendix 8.3. As this thesis is focussed on transforming the BPMN models to FBM models, the explanation of the BPMN model can be found in appendix 8.4.

In this paragraph, some examples are given on how the method of Balsters (2013b) works in practice. An explanation of the naming in the FBM models can be found in appendix 8.2 and the complete BPMN model and its corresponding FBM models can be found in appendix 8.3 and 8.5. A start is made on how a login is modelled in BPMN and how this leads to an FBM model. The names of the models are adapted from the main event that takes place in that model. The names in the models are adapted from the BPMN transformed into ORM events as the method prescribes. In BPMN, the following Login and authorization procedure for a healthcare administration of the pre-triage is modelled:

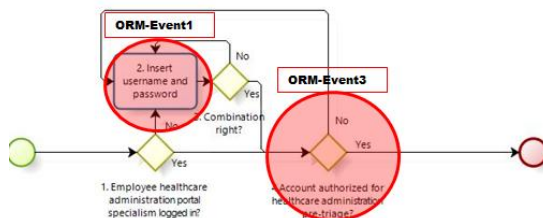


Figure 8 Login & Authorization check

Following the roadmap of the FB-BPM method, the following steps have to be taken:

- 1 Transform a BPMN task into a desired ORM-event

ORM-Event1: [Login: Is Logged]

- 2 Find a minimal model that realizes that event using our fact-type identifying questions

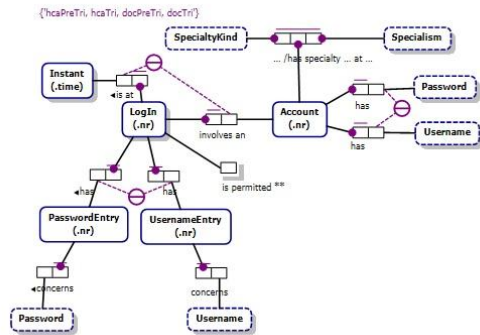


Figure 9 Login in ORM

- 3 Transform the next BPMN task into a subsequent ORM-event

As an example, the part 'RefTrans' of the BPMN is used to show the further translation of BPMN into FBM or ORM models. The BPMN models are supplemented with the data in appendix 8.4 for all the information needed for the FBM models. This starts with the input event 'ReferralCoupling', which can be seen in appendix 8.3. Here is also determined if a doctor is required for (sub-)specialism determination.

ORM-Event2: [ReferralCoupling: Is Logged]
 ORM-Event3: [SpecialtyKindCheck: Is Logged]
 ORM-Event4: [SpecialismDetermination: Is Logged]

If a doctor is required (indicated with an equality constraint in FBM, figure 10) then a Login takes place and if this is permitted, a SpecialtyKindCheck takes place. If the logged in account is of SpecialtyKind docPreTri (see appendix 8.2), a SpecialismDetermination takes place. From here, a referral is attached to a specific (sub-)specialism and only an account from that (sub-)specialism can treat that referral.

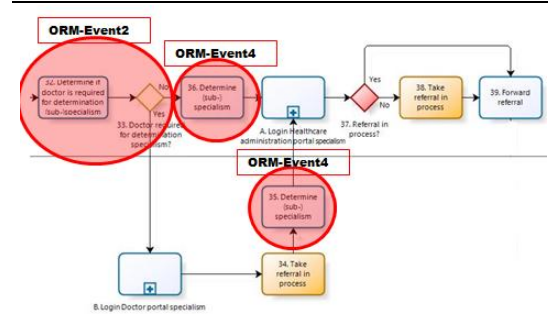


Figure 10 RefTrans Part 1

After the SpecialismDetermination, another Login takes place followed by a SpecialismCheck. This checks if the account of the referral is of the same (sub-)specialism as the referral. The corresponding ORM-event is therefore:

ORM-Event5: [SpecialismCheck: Is Logged]

The forwarding of the referral is a consequence of a health-care administrator or a doctor finishing the determination of the (sub-)specialism of the referral. Therefore there will be no record of this step in the FBM models. The other orange coloured activities also won't be a part of the database. These are steps that are not required to be recorded in the HER. The 'forward referral' event leads up to a login which is shown in Figure 11. In the swim lane of the (sub-)specialism in the triage part (read: not pre-triage) this is the 'determine if doctor is required' event. This event is also transformed into an ORM-event. This is the 'final' event which is followed by a gateway in the BPMN model. According to the rule, it means that this is the end of one model.

ORM-Event6: [TriageDoctorDetermination: Is Logged]

- 4 Find the minimal extension to the previous ORM model that defines that subsequent ORM-event.

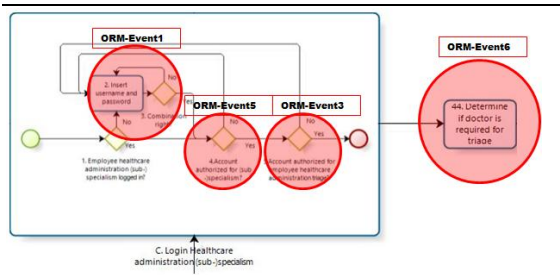


Figure 11 RefTrans Part 2

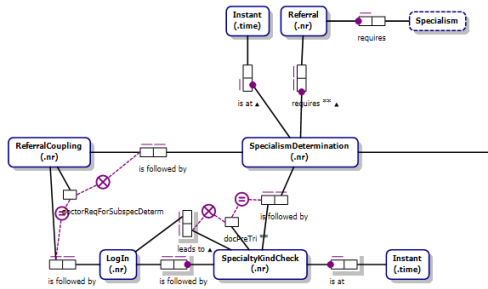


Figure 12 RefTrans Part 1 ORM

As the LogIn model is already modelled earlier, the LogIn in this model is shown as a copy by a shadow behind it. That it is followed by a SpecialtyKindCheck under certain conditions, is also shown as a copy. In the FBM models in the appendix can be seen that a LogIn can only be followed by a SpecialtyKindCheck if that LogIn is permitted. Else it leads to another LogIn. The repetition of events in the models is merely for explanatory purposes, as the relational database model will just mention one table. The ReferralCoupling is followed by a SpecialismDetermination if there is no doctor required. The account that is logged in is that from a healthcare administration from pre-triage, as indicated by the flow of the previous models (see appendix). In the SpecialismDetermination is indicated that for a certain referral one specialism needs to be inserted. This event can also be preceded by a SpecialtyKindCheck (which has to be a doctor for pre-triage) if a doctor is required for the determina-

tion. All activities are instantiated to log at which instant the activity has occurred.

As indicated by the BPMN, the SpecialismDetermination is followed by a LogIn, as shown in Figure 13.

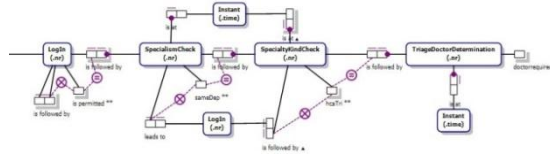


Figure 13 RefTrans Part 2 ORM

What can be seen here, is that first a non-permitted LogIn is followed by another LogIn (indicated with an exclusion constraint:). As there were no limitations set to the number of logins, this could result in someone trying many times to log in. The SpecialismCheck can only occur when a LogIn is permitted, indicated with an equality constraint: . A SpecialismCheck is followed by a SpecialtyKindCheck when the referral is from the same department as the account which has been logged in. When the referral is from a different department, another LogIn takes place as long as this criterion is met. Finally, if the account which has been logged in with is from the SpecialtyKind hcaTri (which is a value derived by the system from Account, indicated with **) the next event can take place, which in this case is the TriageDoctorDetermination. All events are at an Instant(time) which indicates that they are logged at a time when they take place.

5 Repeat 1-4 until all events for all data-stakeholders are finished

All the FBM models that are formed from BPMN can be found in appendix 8.5. For now, an example of difficulties in the modelling of the FBM models is shown.

6 At the end you will have created the complete corporate database, associated to the original business process

This database can be exported to a relational view and can be seen next. The LogIn event is translated to a relational view as follows:

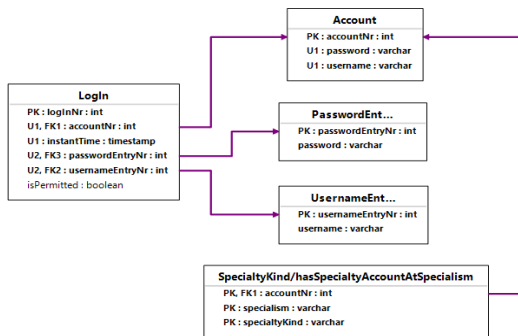


Figure 14 Relational view of LogIn

The relational view of RefTrans is shown in Figure 14. You can see that these relational views, or so-called data models follow the BPMN and the ORM models exactly. This is a major advantage in the design of a database, as now when the business process changes, these changes can easily be tracked down to the corresponding data models.

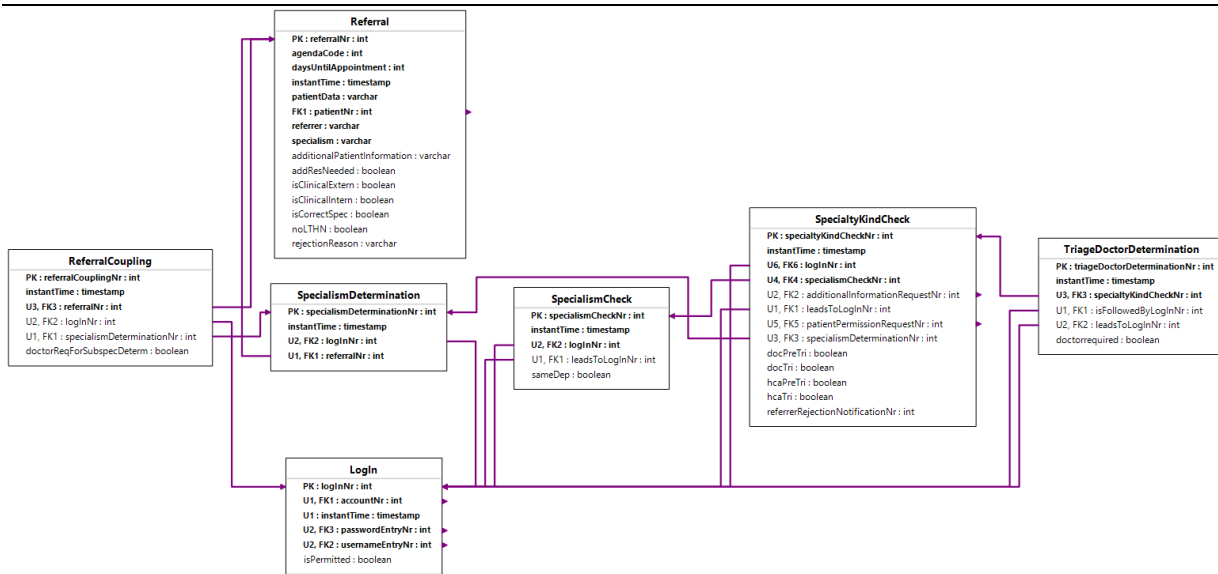


Figure 15 Relational view of RefTrans

4.4 STEP 4: CRITICALLY EXAMINING THE STEPS OF THE FB-BPM METHOD

For the final step of the research, the following questions concerning the FB-BPM method were asked:

- Are there steps missing?
- Do all the steps contribute in reaching the end product?

As said before, the BPMN model of the triage process of the LTHN is that complex, that a more detailed step-by-step explanation on how to transform this BPMN to FBM is required. One can imagine that the business process described here is characteristic for business processes in all hospitals.

The first question to be answered is 'are there steps missing?'. As already stated in chapter 4.3, the model that was designed for the EHR in the LTHN required one more step than was initially described in the method of Balsters (2013b). These steps can be summarized into the following roadmap, based on the design of Balsters (2013b):

Step 1:

Additional step: Determine which parts of the BPMN belong to one ORM model.

Step 2:

Specify the associated ORM events with the BPMN tasks from that.

Step 3:

Find the minimal model that realizes the desired ORM events using the extended fact-type identification method (Hoekstra 2014).

Step 4:

Determine which next parts of the BPMN belong to one model.

Step 5:

Transform that tasks into subsequent ORM events

Step 6:

Find the minimal extension to the previous ORM model fragments that defines the ORM events.

Step 7:

Repeat step 1-6 until all events for all data-stakeholders are finished.

The following general rules from the FB-BPM method, as can be extracted from the FBM models displayed earlier, therefore hold, as they have been proven to work in the modelling of ORM diagrams from BPMN models in this case (Balsters 2013b):
 A simple BPMN diagram is transformed to ORM subsequently:

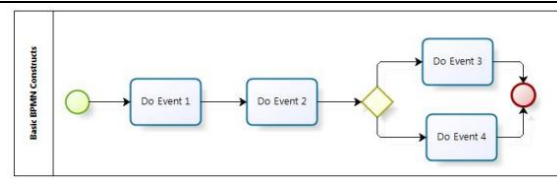


Figure 16 Simple BPMN diagram

A start event is transformed to:

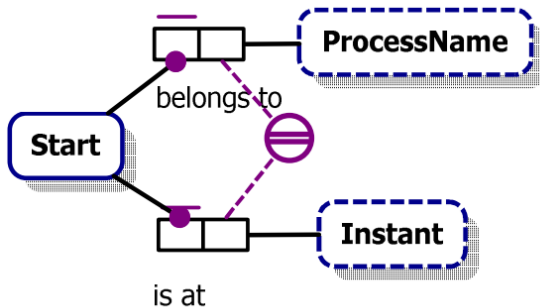


Figure 17 Start Event

The sequence flow of activity 1 that is followed by activity 2:

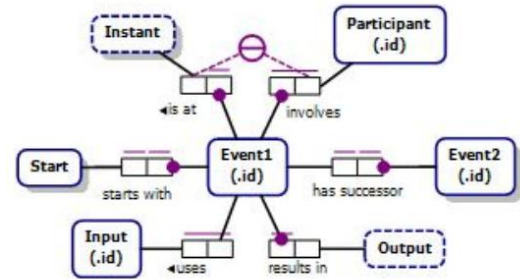


Figure 18 Sequence Flow

A gateway is modelled as follows:

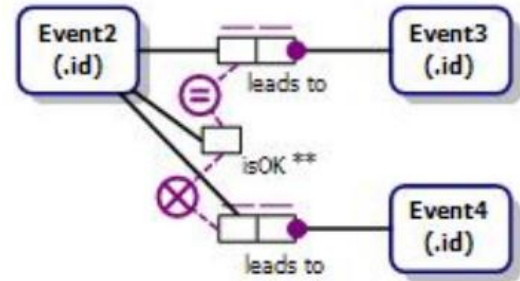


Figure 19 Gateway

And the converging to the stop event is modelled as follows:

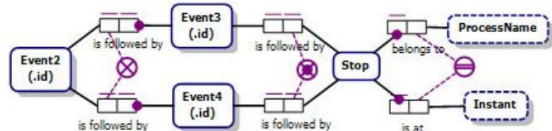


Figure 20 The final events

Following this roadmap leads to a complete corporate database associated to the original business process. Using this roadmap, even complex BPMN models can be made clear so that they can be transformed into ORM in a highly structured way. This also answers the second question 'Do all the steps contribute in reaching the end product?'. The

steps as proposed by Balsters (2013) are only supplemented and none of them were removed.

The validation of the models designed in this research by Spits (2014) was incorporated in the models. This means that the BPMN has been validated by Hoekstra (2014) and the data models by Spits (2014), offering a strong proof of the validity of the designed models.

It has also been proven that one cannot simply work from gateway to gateway in designing an ORM model based on a BPMN model. The designer needs to trace every gateway from the BPMN so that he ends up with complete ORM models. Next to this, in every new activity which had to be modelled in ORM, feedback was given on the BPMN. If for instance the BPMN model lacked information, or contained information which was not specific enough, this would come to surface in designing the ORM models. In the next chapter a reflection is made on how the research has been conducted in this thesis and its drawbacks.

5 DISCUSSION

In this chapter, the results of the research will be discussed. The research as how it has been conducted was done in a short timeframe and should therefore not be overestimated.

The method of Balsters as it was first introduced offered a good instrument in transforming BPMN models to ORM (or FBM) models. It is shown that with using requirements engineering as a starting point for combining BPMN and ORM, a process-driven database design method can work in practice. The critical success factors as formulated in cooperation with the stakeholders were taken into account, as the BPMN was designed according to them. Security issues (e.g. authorization and login) were designed in cooperation with technical domain experts so that also here the CSF's are met.

The ORM models were validated with the end-users using a user-interface mock-up. If the user is content with these mock-ups, one can say that the database as it was modelled fits with the processes of the end-user. This validation is described in Spits (2014). The validation step however only validates this single case of process-driven database design, and does not imply that this case is distinctive for a triage process in every hospital. Next to this, the user interface mock-ups that were made do not contain the entire ORM diagrams. For instance decisions made by the system and other background operations are not seen by the user in the user interfaces.

The research was conducted to receive an answer to the following research questions:

How to validate a proposed general method of process-driven database design in the context of designing a database supporting an EHR-system?

For the validation of the proposed method, two questions were formulated:

- Are there steps missing?
- Do all the steps contribute in reaching the end product?

These questions were answered using the following sub-questions:

- *How can the FB-BPM method be used for a systematic derivation of data models from process models in the context of designing a database supporting an EHR-system at the LTHN?*

This was shown in the chapter **Fout! Verwijzingsbron niet gevonden..** The method as it was developed by Balsters (2013), was used and improved where it could.

- *Are the steps from the FB-BPM method accurate and how can they be improved?*

This was shown in chapter 4. It was shown that the method needed an additional to make the method more clear. Furthermore, the research conducted was only focussed on the triage process of the LTHN. This is however a small fraction of the to-be designed EHR. The plausibility of the validation of the improved method could improve greatly with the design of an entire EHR using this method. If this method is to be scaled up further (e.g. national EHR design), other concerns come to mind. In that case, inter-organizational terms and activities should be standardized with all the parties involved.

6 CONCLUSION

This research is part of a larger research that was started with the aim of validating a proposed method for designing a corporate database concerning an EHR system according to process models. Current research in this field lacks a structural approach for designing a database, which often leads to failure of the database. In this part of the overall research, the proposed method of transforming BPMN models and their associated information to ORM models was used and critically examined. For this, two main validation questions were set up: 'Are there steps missing?' and 'Do all the steps contribute in reaching the end product?'. This resulted in an answer to the main and sub-questions of the research.

6.1 SCIENTIFIC CONTRIBUTION

With this research it is demonstrated that the FB-BPM method is applicable to complex BPMN models. This has not been shown before in any research. Next to this, the research has shown that with the design of the ORM models, feedback was given on the correctness of the BPMN. An incorrect BPMN could therefore not lead to a database and the feedback led to more accurate BPMN models. With the validation of the method, the applicability of it will greatly improve. It is however suggested, that further research focusses on applying this method with different hospital cases, or as suggested, in the design of a complete EHR. The expectation is that more improvements to the method can be made for even better applicability.

7 REFERENCES

- Balsters, H., 2013a. Lecture Slides of course: "Design Methods." University of Groningen.
- Balsters, H., 2013b. Mapping BPMN process models to ORM data models. In: *Lecture Notes in Computer Science*, LNCS 8186 (Springer Verlag).
- Balsters, H., 2012. ORM Logic-based English (OLE) and the ORM ReDesigner tool: Factbased Reengineering and Migration of Relational Databases. In: *Lecture Notes in Computer Science*, LNCS 7567 (Springer Verlag).
- Bridgeland, D.M. & Zahavi, R., 2009. *Business Modeling 1st ed.*, Morgan Kaufmann Publishers.
- Chinosi, M. & Trombetta, A., 2012. BPMN: An introduction to the standard. *Computer Standards & Interfaces*, 34(1), pp.124–134. Available at: <http://linkinghub.elsevier.com/retrieve/pii/S0920548911000766> [Accessed September 18, 2013].
- Doesburg, J. & Balsters, H., 2012. Fact-Based Specification of a Data Modeling Kernel of the UML Superstructure. In: *Lecture Notes in Computer Science*, nr. LNCS 7 (Springer Verlag).
- Van Dorrestijn, M., 2012. Schippers: "Specialisatie komt ziekenhuis ten goede." *Zorgvisie.nl*, p.1. Available at: <http://www.zorgvisie.nl/Personeel/Nieuws/2012/11/Schippers-Specialisatie-komt-ziekenhuis-ten-goede-ZVS015385W/> [Accessed September 24, 2013].
- Halpin, T., 2001. Object role modeling: An overview. white paper, (online at www.orm.net). . . ., (February). Available at: <http://courses.washington.edu/css475/orm.pdf> [Accessed October 9, 2013].
- Halpin, T. & Morgan, T., 2008. *Information Modeling and Relational Databases 2nd ed.*, Morgan Kaufmann Publishers.
- Hijlkema, J., 2013. EPR exchange system. Prototype development using design science. University of Groningen.
- Hoekstra, S.G., 2014. Validation of a process-driven database design method for an Electronic Health Record-system: developing the process models. University of Groningen.
- Insrán, E., Pastor, O. & Wieringa, R., 2002. Requirements engineering-based conceptual modelling. *Requirements Engineering*, pp.61–72. Available at: <http://link.springer.com/article/10.1007/s007660200005> [Accessed October 2, 2013].
- Modderkolk, H., 2013. Animo EPD valt tegen: nu pas miljoen patiëntgegevens verzameld. *NRC Handelsblad*, p.1. Available at: <http://www.nrc.nl/nieuws/2013/08/13/animo-epd-valt-tegen-nu-pas-miljoen-patientgegevens-verzameld/>.
- Patig, S., 2006. Evolution of entity–relationship modeling. *Data & Knowledge Engineering*, 56(2), pp.122–138. Available at: <http://linkinghub.elsevier.com/retrieve/pii/S0169023X05000339> [Accessed October 8, 2013].
- Post, S.A., 2013. EPR Exchange Systems: Development of a Conceptual Database Model Using Design Science. University of Groningen.
- Selvakumar, J. & Rajaram, M., 2011. Performance Evaluation of Requirements Engineering Methodology for Automated Detection of Non Functional Requirements. *International Journal*, 3(8), pp.2991–2995. Available at: <http://www.enggjournals.com/ijcse/doc/IJCSE11-03-08-083.pdf> [Accessed October 23, 2013].
- Spits, E., 2014. Validation of a process-driven database design method: Validation of the to-be Electronic Health Record-system data-model and the process-driven database design method. University of Groningen.
- Stephana, L., 2013. EPR exchange system: conceptual business process model development using design science. University of Groningen.
- Van Strien, P.J., 1997. *Towards a Methodology of Psychological Practice: The Regulative Cycle. Theory & Psychology*, 7(5), pp.683–700. Available at: <http://tap.sagepub.com/cgi/doi/10.1177/0959354397075006> [Accessed October 22, 2013].

- In 't Veld, J., In 't veld, M. & Slatius, B., 2011. Analyse van Bedrijfsprocessen, Noordhoff Uitgevers.
- Vereniging Zorgaanbieders voor Zorgcommunicatie, 2012. Het Landelijk Schakelpunt. VZVZ.nl.
Available at:
<https://vzvz.nl/page/Zorgconsument/Het-LSP>.
- Wieringa, R. & Heerkens, H., 2007. Designing requirements engineering research. Comparative Evaluation in Requirements Engineering, pp.36–48.
Available at:
http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4557817 [Accessed October 22, 2013].

8 APPENDIX

8.1 THEME CASE DESCRIPTION

Theme: An Electronic Health Record (EHR) system within a LTHN

Project: Validation of the logic business data model of the LTHN

Supervisor: Dr. H. Balsters

Prerequisites: Systems design, Business modeling, Basic IT-skills (process- and data modeling, databases), affinity with engineering systems

Situation

At a LTHN (Large Teaching Hospital in the Netherlands), a large migration project is in progress to replace the current Hospital Information System (HIS) with an advanced Electronic Health Record (EHR) system. Currently it is being investigated how to migrate some fifty source databases into one central target database system. This central database is a prerequisite for an EHR system such as developed at the LTHR.

This migration process is an extremely complex operation; not only is the overall size of the source data involved overwhelming, but in the process one has to also figure out which source data is relevant to the target system, and subsequently one has to figure out how to integrate and map the source data to the target system in a consistent manner. In the source systems, data may be redundant with respect to the to-be designed target system; the data may also be incorrect or out of date. Hence the migration process also offers an opportunity to cleanse existing data. Since hospital-oriented systems also lay a huge emphasis on reliability and safety, such a migration process has to be managed in a controllable fashion, meaning that each step in the migration process has to be meticulously defined and validated.

In the migration process we distinguish functional aspects and technical aspects. Functional aspects concern the desired functionalities of the system, while technical aspects concern the implementation of the functional aspects into a technically working system. Our theme concerns the design of a logical and technical model based on process analyses to fulfill the functional needs of a patient admission process in a EHR system.

The LTHR has already developed a data model for a central database environment to merge all hospital data from multiple source systems to a unambiguous defined data model, called the LBG (Logisch Bedrijfs Gegevens Model). This data model is based on the 'reference domain model for hospitals' (RDH) which is a reference model for the relationships between the various business processes and information objects within a hospital environment. The LTHR has expressed the need to validate the correctness of the LBG. In order to keep the scope of our validation project under control (our resources are limited: three students, simultaneously working within a period of three months), we shall confine the model to a part of the EHR system; those parts of the data model that are involved in the process of triage and routing of patients that are referred to the hospital for a specific medical problem.

Problem

Information about the patient is captured in so-called Hospital Information Systems (**HIS**). In order to cater to a hospital with a large diversity in users, a HIS tends to be very complex, even when one, for example, confines the system to, say, patient admittance within the hospital. These systems also deal with very privacy-sensitive medical patient data. As a result they are strictly locked-down to prevent unwanted leaks of these types of data.

In order to enter or transfer patient data within a hospital, a model to facilitating this patient data entry and exchange is envisioned. However, hospitals are struggling with constructing such a system: one has to figure out which source data is relevant to the targeted EHR-system, and subsequently one has to figure out how to integrate and map the source data to the target system in a consistent manner. On the other hand, Patient Data entry and exchange is often reasonably well understood as a process: we have information on how to process patient data, which checks are to be performed, in which order the process should take place, etc. In order to explicitly define such a process, we wish to design a business process model capturing all of the requirements necessary to perform patient data entry and exchange in a consistent and complete manner. Once such a process model of patient data entry and exchange is realized, we can move on to the design of

the underlying data model supporting the process model. This data model (a conceptual blueprint of a database) will offer the backbone of the a part of the EHR-system. One could say that the process model offers the context in which the data model is to be developed. Hence, the EHR-process is leading with respect to the EHR-data. The University of Groningen has already developed his own method to systematically map process models(specified in SIPOC) to data models. Our proposed research aims at realizing these same process- and data models supporting an EHR-system at the LTHN by using the BPMN to FBM method. BPMN stands for Business Process Modeling Notation, and constitutes an internationally accepted (ISO-) standard for modeling business processes. FBM stands for Fact-Based Modeling, and constitutes an ISO-standard for data modeling. In summary, we can state that our university wishes to investigate how data is currently entered and exchanged, what the requirements for an entry and exchange system are, and how to design a patient data entry and exchange system based on these requirements, resulting in a data model that could facilitate an EHR system. Once we have derived an FBM data model supporting patient admittance as part of the LTHN EHR, we can compare our FBM data model with the existing LBGGM data model to validate correctness of (part of) the LBGGM.

Project Description

Our research aims at solving the following question:

“How to validate the correctness of data models supporting an EHR?”

This leads us to the following sub-questions:

1. What is the state of the art concerning the entry and exchange of medical patient data within the LTHR?
2. Who is currently entering and exchanging data, what data is entered and how is the data exchanged?
3. What are the constraints on patient data entry and exchange?
4. What are the requirements of a functional design of an LTHN EHR-system according the admittance process?
5. How can we code these requirements into a process model?
6. How can we systematically derive a data model from the process model?

7. How does the existing LBGGM data model of the LTHN compare to our derived data model?

Organization

Three students will be working simultaneously on this project for a period of three months. Starting date will be September 1st, 2013.

8.2 NAMING IN THE MODELS

- Pre-triage: This refers to the part of the triage where the referral is on the portal specialism, until it is being sent to the right (sub-)specialism.
- Triage: This refers to part of the triage where it is on the (sub-)specialism where it has been sent to.
- docPreTri: This is an attribute of an account which is a doctor authorized for the pre-triage
- docTri: This is an attribute of an account which is a doctor authorized for the triage
- hcaPreTri: This is an attribute of an account which is a healthcare administrator authorized for the pre-triage
- hcaTri: This is an attribute of an account which is a healthcare administrator authorized for the triage.