Utrecht University

Master Thesis

Neonatal Care Admission Optimized

A data processing architecture for real time data, to manage the bed capacity at birth centres

Author:
Devika Jagesar

Supervisors:
dr. Verónica Burriel Col
dr. Sietse Overbeek
dr. Fabiano Dalpiaz

A thesis submitted in fulfillment of the requirements
for the degree of Master of Science

in the

Centre for Organization and Information
Department of Information and Computing Sciences

July 6, 2020

Thank you for always watching over me, and having my back Lord Krishna and Lord Shiva...

UTRECHT UNIVERSITY

Abstract

Faculty of Science
Department of Information and Computing Sciences

Master of Science

Neonatal Care Admission Optimized

by Devika Jagesar

The research presented in this thesis addresses the use of a data architecture creating a unified data environment to manage algorithms in real time. Many healthcare organizations are researching the added value of real time data, and machine learning methods to optimize the quality of healthcare, and contribute to several projects that research medical treatments. However, the existing data environments of the hospitals are not always defined in an architecture, resulting in many data flows that are created “on the fly” for several applications. The need of a unified data environment is relevant for efficiently managing data flows and extending the existing data environment with new applications. This thesis is centered around a data science initiative called Predict that uses algorithms to predict the amount of premature babies within a certain timeframe. The main objective of our research is to create a data architecture that guides the data experts in creating an efficient real time data flow for Predict. By observing and interviewing the data experts, we have identified the requirements and related tasks that will help create a data flow. The requirements are mainly focussed on managing the data processing activities like extraction, cleaning and integration in real time. The tasks that are derived from the requirements are classified in a layered architecture that can be extended with data models and principles the data needs to adhere to, and also helps data experts to choose a data processing method like ETL or data virtualization.

Acknowledgements

Thank you…

Dr. Veronica Burriel Col for being my supervisor and friend during this project. Your support and friendship provided me with smiles during this journey. Your knowledge and experience contributed to our research, and I will always be thankful to the confidence you had in me.

Dr. Sietse Overbeek and dr. Fabiano Dalpiaz for your feedback and support in the final phase of this project. I highly appreciate it!

Marc Bouma for supporting me, and this research project. Your knowledge and expertise were of great value.

Jolien Ketelaar for your guidance during the research process, and your encouraging nature towards me and the project.

Papa for always supporting me, and for being an amazing dad. You mean the world to me!

Darshana for being my best friend and little sister. Thank you for always bringing a smile on my face.

- Devika Jagesar, 5th of July, 2020

Content

List of Figures

List of Abbreviations

NICU	Neonatal Intensive Care Unit
WHO	World Health Organization
ADAM	Applied Data Analytics in Medicine
UMC	Universitair Medisch Centrum
PREDICT	PREgnancy Data analytics to Improve outcome & hospital CapaciTy
WKZ	Wilhelmina Kinderziekenhuis
ETL	Extract Transform Load
EHR	Electronic Health Record
PCC	Patient Centered Care
CRISP DM	Cross-Industry Discovery in Databases Model
KDD	Knowledge Discovery in Databases
AD Process	Algorithm Development Process
HIX	Healthcare Information X-change
BMA	Buro Medische Automatisering BV
IC	Intensive Care
EDW	Enterprise Data Warehouse
AI	Artifical Intelligence
EU	European Union
UML	Unified Modelling Language
HL7	Health Level Seven

Chapter 1

Introduction

1.1 About Neonatology

The term “Neonatology” was introduced in 1960 by Alexander Schaffer and refers to the care and treatment of premature infants [2]. A baby is considered premature when it is born under 37 weeks. In developed countries, prematurity and low birth weight (<2500 g) are the leading causes of death among babies. Babies that are born preterm are admitted to the Neonatal Intensive Care Unit (*next: NICU*), a department specialized in giving care to prematurely born infants (“*neonatals*”) at the Neonatology Unit. Babies that are admitted at this department suffer from several medical complications like breathing problems, heart problems or Sepsis which is a blood poisoning infection that leads to organ failure [5] [6].

In the past decades, the perinatal, neonatal and infant rates have decreased, however it is still considered an important health issue as the amount of deaths are still high [3]. According to the World Health Organization (*next: WHO*), 15 million premature babies are born worldwide, and this number is only increasing. The preterm complications that a mother suffers from during her pregnancy, caused a death of approximately 1 million children under the age of 5 years in 2015 [4].

This thesis is centered around the Applied Data Analytics in Medicine (*next: ADAM*) program of the Universitair Medisch Centrum Utrecht (*next: UMC*) [1]. ADAM consists of nine projects that are focussed on applying several data science techniques on the huge amount of data that resides within the information systems of the hospital, to optimize the quality of healthcare. This thesis is in collaboration with one of the projects within ADAM called PREgnancy Data analytics to Improve outcome & hospital CapaciTy (*next: Predict*).

The main objective of Predict is to optimize the bed capacity at the birth centre at UMC in an effort to ensure timely and effective help to expecting mothers reducing the risk of health complications to mother and child.

1.2 Data Analytics & Neonatal Care Admission

Predict revolves around four main deliverables that are required to optimize neonatal care admission. The project will require two algorithms, a dashboard and an architectural foundation to work on:

- Algorithm 1: A capacity algorithm that provides an insight in how many Intensive Care babies can be expected within a certain time frame
- Algorithm 2: An adverse outcome prediction where the algorithm tries to measure the probability of the pregnant woman developing complications during the pregnancy
- Architecture: Both algorithms require a mixture of (near) real time and historical data to work with. This data needs to be pulled from various information systems and databases

- Dashboard: Finally, a dashboard ties it all together operationalizing the newly created insights helping NICU staff in allocating beds to expecting mothers

For this research project we will focus on:

1. How to manage data flows once the algorithm goes into production

2. How the algorithm needs consolidated data to be analyzed by the algorithm

We will therefore focus on designing an architectural foundation for this project by only analyzing *the first algorithm*. The second algorithm and dashboard are out of scope.

When we use the term consolidated data, we mean data that is integrated, transformed and cleansed. Figure 1.A presents an overview of two data processes, the algorithm development and the algorithm in production. Data experts use historical data to create the algorithm and a lot of tasks are executed manually, resulting in a time consuming process.

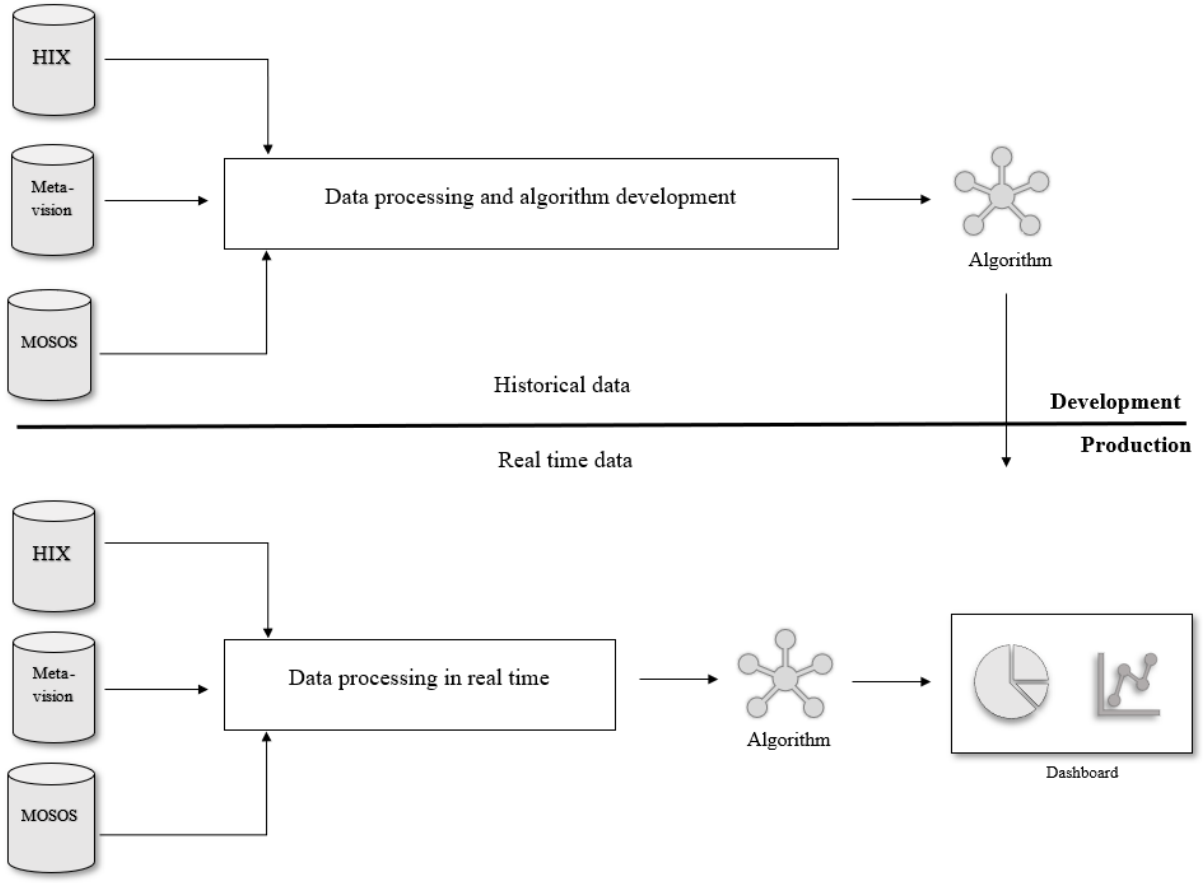

Figure 1.A Development and Production Algorithm [7] [8] [9] [15]

As indicated in figure 1.A, the data experts are expected to use three data sources for the algorithm development:

Healthcare Information X-change (*next: HIX*)

HIX is the Electronic Health Record (*next: EHR*) platform of the hospital and is developed by Chipsoft. The company is a marketleader in managing and providing service to healthcare systems that store patient information. They also provide software solutions for independent

clinics and nursing homes [82]. HIX is the hospitals primary datasource where they store the patients personal information (name, contact information and financial data), and data surrounding the treatment of the patient.

MOSOS

MOSOS is a software developed specially for obstetrics data. It is introduced by Buro Medische Automatisering B.V. (*next: BMA*). This company is a marketleader in the Netherlands and Belgium when it comes to IT solutions for the obstetrics department in hospitals. There are over 200 hospitals in the Benelux, France and England that use the IT products of BMA [83]. At the UMC, MOSOS was used to store pregnancy and delivery data for patients.

In order to create the algorithm, the data scientists are creating datasets for each woman that gave birth at the UMC from 2015 and onwards. To create this dataset, the data experts sometimes had to integrate MOSOS and HIX data to complete the dataset of a patient [8].

MetaVision

MetaVision is developed by iMDsoft and is defined as a Patient Data Management system for the Intensive Care (*next: IC*) and Operation Room units of hospitals. The company is founded in 1996 and is focussed on building clinical information systems that are used by over 400 hospitals worldwide [84]. MetaVision stores the data generated by the IC monitors, the IC equipment and the information that is manually added by the clinicians. The data experts have not yet explored the data that resides in MetaVision, but are considering using this data source in the future to modify the algorithm.

We will explain the development of the algorithm in more detail in chapter 4. Once the algorithm goes into production, Predict wants to use real time data to support the clinicial decision making process [15] [8].

1.3 Case Study: The NICU department at the UMC

The neonatology department looks different from the other departments in the hospital. There are incubators where premature born, or sick babies can be monitored and receive treatment [5].

There are two more departments related to the NICU, the high care department and the medium care department. When a baby is admitted to the NICU but is showing signs of progress, they can be moved to a post intensive care department called high care. This department can be located in the same hospital or a different one. At high care, the baby still needs regular observation and treatment [5].

When the baby is not ready to go home, but the treatment at a NICU is not necessary anymore a transfer to medium care will be the next step. The baby can shift from NICU directly to medium care, or has to stay at high care for some time before it can be admitted to the medium care department. The baby is still monitored, but it is not necessary to have a neonatologist present. A pediatrician is present at the department to continue the treatment [5].

The duration of stay at the NICU or the related departments can vary. Some babies are allowed to go home after a couple of days, unfortunately some babies have to stay at the intensive care for several months [5].

The neonatology department at the birth centre of Utrecht, called Wilhelmina Kinderziekenhuis (*next: WKZ*), has physical beds and operational beds. Physical beds are the amount of beds that are physically present at the department. Operational beds are the amount of beds that can actually be used depending on the nursing staff and available materials [5].

The medical staff

There are several medical experts active at the NICU:

- Pediatricians
- Neonatologists
- Specialists
- Nurses

The nurses in particular are very important when it comes to managing the capacity. The amount of babies that can be admitted to the NICU are dependent on the amount of nurses that are available at that moment. In other words, the nurses determine the capacity in the birth centre [5].

NICU in the Netherlands

There are a lot of hospitals in the Netherlands that have a neonatology department, but there are only ten hospitals that have a NICU department. Eight of them are located in our Academic hospitals:

- AMC and VUmc in Amsterdam
- Erasmus MC-Sophia in Rotterdam
- LUMC in Leiden
- Maastricht UMC+ in Maastricht
- Radboud UMC in Nijmegen
- UMC Groningen in Groningen
- UMC Utrecht Wilhelmina Kinderziekenhuis in Utrecht

Because the distance between some NICU's was still too large, they added two other NICU's in a general hospital located in Zwolle and Veldhoven. These ten hospitals are responsible for all the neonatal intensive care in the Netherlands [5] [43].

Patientflow NICU

Every NICU has a couple of regional hospitals they are responsible for. For example, the NICU in Utrecht is responsible for hospitals in Tilburg, Nieuwegein, Leidsche Rijn, Woerden, Amersfoort, Tiel, Apeldoorn and Deventer.

When a NICU baby is born in one of the abovementioned hospitals, the NICU department in Utrecht will pick up the baby from that hospital and start treatment at WKZ. If the birth centre in Utrecht does not have a place for the baby, the clinician will call other NICU departments in

the country to check their capacity, and the baby will be admitted in another hospital. Figure 1.B presents an overview of the in- and outflow of the baby [5].

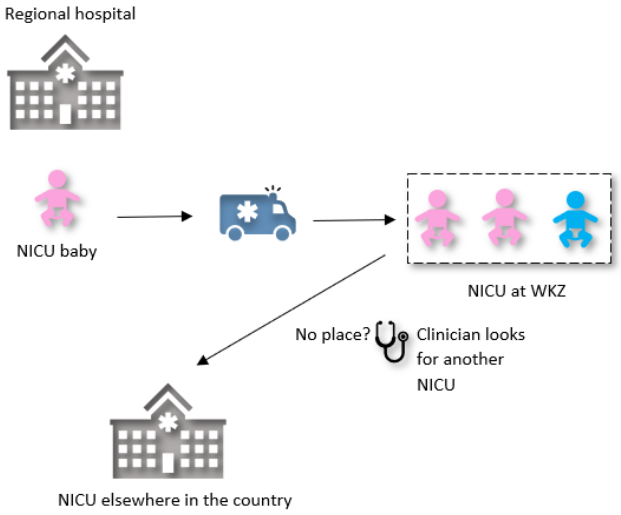

Figure 1.B In- and outflow of the NICU baby [5]

Predict & UMC NICU

The previous subsection provided a brief overview of the structure of the NICU at UMC and nationwide. Because of estimation errors about the moment of delivering the baby, the hospital is unable to fill beds, or make place at the right time. The hospital wants to optimize the facilities at the NICU and started project Predict [5].

Predict is a project that is executed at the NICU department. The main objective of Predict is to optimize the bed capacity at the NICU by using prediction models. Their end product is a dashboard that visualizes the output of the algorithm to improve the decision making process of the clinicians at the birth centre.

The NICU in Utrecht is currently coping with a capacity problem. Capacity in healthcare is a broad term. It can refer to physical beds, resources or staff members [10]. In this research, capacity is referred to the shortage of nurses at the NICU. At the moment, there are enough beds and materials to treat the babies, but not enough nurses to manage the current capacity. This shortage of staff members is restricting the NICU department to optimize their capacity, and use the beds that are available at the unit [5].

The question has been raised to assign more beds to a nurse, but that will take away from the quality of treatment. In order to maintain the high quality of care that the birth centre is aiming for, it would not be safe to assign more beds to a nurse. A nurse gets the amount of beds assigned according to the current situation at the NICU department, and it is a collaborative effort between the doctors and nurses to decide the amount of beds the nurse is in charge of [5].

The consequences of not having enough staff members is that the birth centre often has to reject patients or transfer them to other hospitals (pregnant women and babies). The transfer of a patient is defined as the moment a pregnant woman started treatment at the hospital in Utrecht but there is not a NICU place available at WKZ, and the mother and baby have to be shifted to another hospital. The rejection of a patient is defined as the moment a gynecologist from another

hospital contacts WKZ to check the NICU capacity at WKZ because they have a patient that might give birth to a premature baby but WKZ has no place to treat the mother and baby [5]. In this research, we will refer to the pregnant woman as the patient.

The complexity of NICU admissions lies in the moment of decision making, and not being able to estimate when the baby is going to be delivered during a preterm birth. The estimation of the moment of delivery is a tough task. When a patient is suffering from contractions or any other form of labour pain, it does not necessarily mean that she is going to deliver the baby however, she will be admitted to the hospital. This implies that there is a significant time interval between the moment the patient gets admitted to the hospital, and her actually giving birth to the baby. Figure 1.C provides an overview of the pregnancy timeline [5] [11].

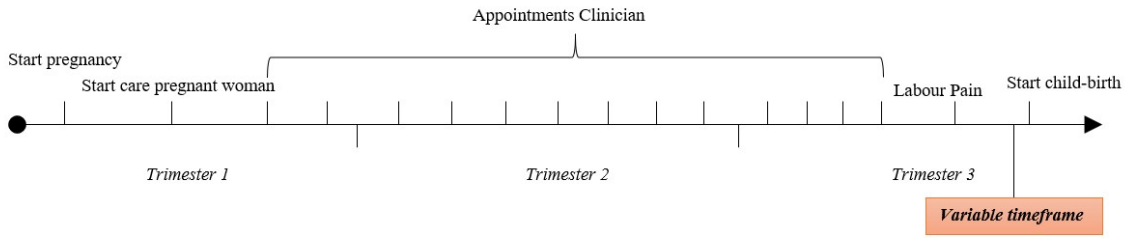

Figure 1.C Pregnancy timeline [11]

Professor E. Hans, an expert at operational planning in healthcare, supports Predict from the capacity planning perspective. According to him, capacity in healthcare is difficult to manage, primarily due to the unpredictability of the patient demand. The key to capacity planning is to manage the wide range of variability (patient demand) in the department by creating flexibility. In Predict, flexibility is created by gaining more insight in the incoming patients and babies [12]. In other words, they want to predict the time interval between the hospital admission of the pregnant woman, and her actually delivering the baby as presented in figure 1.C. Therefore, the output of the algorithm will provide an overview of which mother is going to deliver a NICU baby within x amount of hours [5] [13].

For example, if the clinician is aware that 3 patients are going to deliver a NICU baby between 6 and 12 hours, they can anticipate on the NICU demand. They can check their NICU department and shift babies to high care, if medically possible. It can also result in transfering patients with the baby in the womb, prior to delivery. The hospital is trying to prevent that the patient has to give birth, and the baby eventually has to travel to another NICU. The hospital would rather have the patient with baby in the womb placed in the right hospital, so neither the patient nor the baby has to travel elsewhere [5].

1.4 Problem Statement & Research Objective

As mentioned in the previous subsection, the main objective of the Predict team is to develop an algorithm that predicts the incoming NICU babies within a certain timeframe. This thesis will focus on designing an architecture to facilitate and support the aforementioned algorithm excluding the dashboard that will be used for decision support purposes by the NICU staff. See figure 1.D for an overview of an abstracted data process on which the proposed data architecture will be based.

Our project mainly focusses on getting the right data in the right format to contribute to the clinicians decision making process. The algorithm is a given, but we need to research what will be the best approach to support the algorithm when it starts running. According to the Predict team, one of the requirements we need to take into account is the need of real time data.

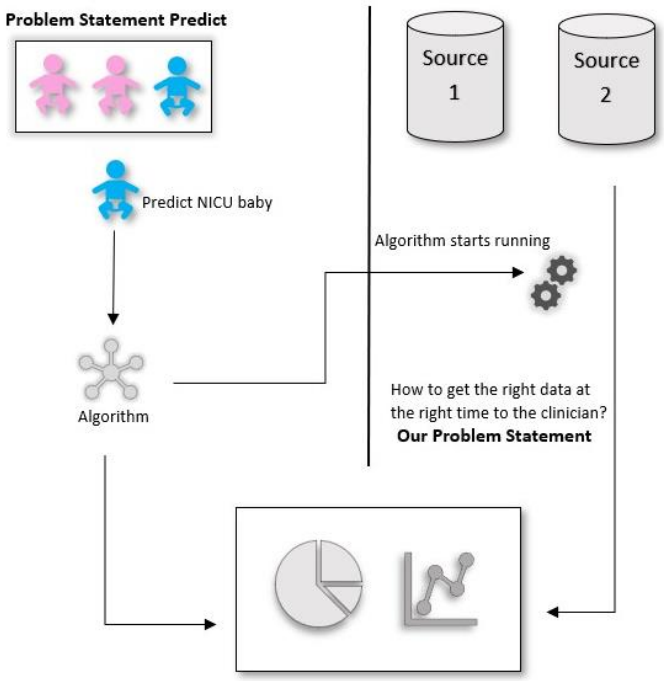

Figure 1.D Predict and our project

Moreover, they want to develop a dashboard that provides real time data that is instantly processed once a pregnant woman is admitted to the hospital. The algorithm is created manually, providing the data experts enough time to process the data (cleaning and transforming data). Once the algorithm goes into production, the data processing activities need to satisfy real time requirements like access to a real time database and real time data processing.

Another challenge to take into account is the integration of different isolated information systems that contain the data that needs to be processed in real time. Our main objective is to research what data processing characteristics would fit Predicts data process and how these characteristics need to be managed. To structure these characteristics and present an efficient data flow, we aim at conceptualizing a data architecture that guides data experts in creating an effective real time process that delivers consolidated data for the algorithm to process.

The advantages of having a solid architecture in place range from improving the data quality and providing reliable data and reporting, to a clear understanding of your data environment and how it correlates to your business objectives and processes [44].

1.5 Related Work

Our research is focussed on providing a data architecture that helps data experts create a real time process. This will eventually result in providing the right data set for an algorithm that calculates the risk of a pregnant women. The challenges at the NICU that we have addressed in

the previous chapter are not limited to our country. Researchers in India have developed a software solution to manage the several NICU challenges they identified. A couple of these challenges are listed below:

- Complete health monitoring
- Predictive analytics
- Reduced medication errors
- Pattern analysis
- Nutrition and medication calculators

They developed a software solution that functions as an integrated platform to support data flows that relate to the abovementioned challenges. This platform integrates cloud technology, Internet of Things and data analytics and results in providing:

- Integrated data
- Laboratory reports
- Notifications for clinicians
- Predictive analytics

The paper proposes an architecture that specifies an abstract data flow including software specifications. They have created a machine data integration layer that defines several clinical data standards that specify how the data should be cleaned and integrated, and the tasks are executed by Apache Kafka, an open source platform that supports streaming data. This research provides us with insights about how data is cleaned and integrated. For example, the paper presents examples of a machine data integration layer that includes several wrappers that specify clinical data standards the data needs to adhere to during the cleaning and integration process.

Moreover, this research can also be used by the Predict team to extend the dashboard functionalities of Predict. The paper provides several mockups of their own dashboard and its functionalities [107].

As we have mentioned before, our thesis is focussed on processing real time data in healthcare. Unfortunately, we could not find specific methods that dissect the dataflow and their several tasks. However, we did find some literature that specify several applications to manage the data process. There are researchers who have investigated the need of big data analytics in healthcare, and provide an architecture that explain how existing applications like Hadoop can be used to manage the several components of a data process (cleaning, integrating, data delivery). The architecture in this paper specifies how the data flows from the clients request, using meta data and the functionality Map reduce within Hadoop, to integrate the data and provide the right data to the client [70].

Another research explores how cloud computing can be used to collect and process patient data. The architecture created in this research specifies how a data process is automated by using sensors that are attached to medical equipements to exchange several services. Next, the data that is available in the cloud is processed and propagated towards the medical experts. The proposed solution supports real time data processing, and automating data processes by eliminating manual data collection. Moreover, it also provides a new perspective on how cloud technology can be used to create new real time data flows [74].

The abovementioned examples do not specify how the several tasks are managed during the process. They are more focussed on how the applications add value to the data process. However, they do propose similar challenges:

- Data integration
- Data quality
- Managing data latency

We aim at dissecting the several tasks of a data process and the challenges in more detail in the chapter 3. We will also focus on how they can be managed when executed in a batch process and real time process, therefore chapter 3 is an extension of our literature review.

Chapter 2

Research Design

2.1 Research questions

The data processing architecture that is going to be designed to predict the incoming NICU babies has to fulfill two general requirements:

- It needs to include the prediction algorithm
- It needs to process real time data

We have formulated our main research question as follows:

How can we develop a data processing architecture supporting prediction algorithms for NICU bed capacity management?

Our goal is to aid Predict in developing a real time data process to support the capacity problems at the NICU. We aim to do this by researching data processing characteristics and how these should be managed in a real time data environment by using an architecture to structure the data flows.

To answer our main research question we have created five sub questions:

1. How is the NICU capacity managed at the moment, and how does the capacity management impact the patient flow?

The first step is to become familiar with the NICU environment and existing workflows at the unit. This will provide us with a better understanding of the problem statement but more importantly, we get an insight in the existing challenges at the unit from a clinicians perspective. This will help us understand the problem domain, and the added value of the algorithm.

2. What is the current understanding that domain experts have about processing/managing data in a healthcare environment and how has this evolved?
2.1 How does data analytics influence the healthcare sector, and what are the challenges?

Before we dive into the data process of the algorithm development, we need to research how data processing has evolved over the years, especially with the introduction of real time data. We will look into the main characteristics of data processing and the challenges that business users face. It will help us gain more insight in how to manage the characteristics, especially in a real time setting. Another relevant topic to research is the progress of using real time data in the healthcare sector and explore their challenges.

3. What are the requirements that we need to take into account while designing the data architecture?

To create an architecture that provides support to the data process that is needed when the algorithm starts running, we need to understand how the algorithm is developed and the challenges that the Predict teammembers face while creating this algorithm. This data process will function as a guideline for IT, architects and data scientists to indicate what needs to be changed to create a functional data process for the algorithm.

4. What data architecture is needed to manage the data flows, and organize the data in the information systems?

Once the data processing foundations and the requirements are clear, an architecture can be developed that will structure the different tasks that need to be executed in the data process.

5. What are the advantages and disadvantages of the proposed data architecture?

The architecture will be validated by a data expert who is familiar with algorithm development and data management.

2.2 Research approach

Our research will be performed within the Design Science paradigm, which is primarily a problem solving research pardigm [14]. Design science is defined as *"A research method that seeks to create innovations that define the ideas, practices, technical capabilities, and products through which the analysis, design, implementation, management, and use of information systems can be effectively and efficiently accomplished"* [14]. The innovations refer to the artifacts that are created within Design Science and are broadly defined as constructs, models, methods and instantiations. These artifacts help to predict or explain a phenomena in a particular research area and are created to solve organizational problems [14]. The Design Science Paradigm fits this research project as we are aiming at creating and validating an architecture to improve the data processes in a healthcare environment. Figure 2.A presents the Design Science framework applied to our research.

To structure our research path we follow the Design Cycle. The first phase is the problem investigation phase where we define the problem statement and explore the problem domain. In the next phase called treatment design, we identify the requirements that are needed to build our solution and combine the knowledge of the first phase to construct a model. In the last phase, we evaluate the proposed solution by an expert who is familiar with applied data science in a healthcare environment.

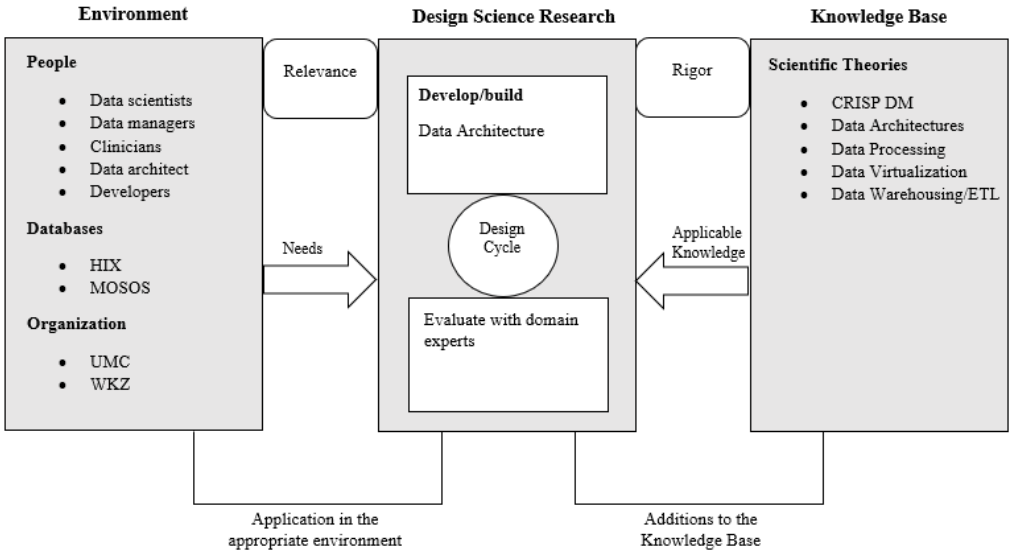

Figure 2.A Design Science Research Framework [14]

This project is divided in two phases. The first phase of this project was dedicated to explore the problem domain and perform an extensive literature study to establish the theoretical foundation of this project. During our literature study we tried to identify the gaps in the current literature and aim at providing new insights by constructing our solution. A semi – structured literature review is conducted by using the snowballing procedure [16]. This procedure refers to continuously searching and scanning the references of articles, books and other documents. The library of choice is GoogleScholar because of its wide variation of scientific literature in the domain of computer science. In the second phase, we started conceptualizing the artifact and eventually evaluated the artifact with an expert. The main activities of the project are presented in figure 2.B.

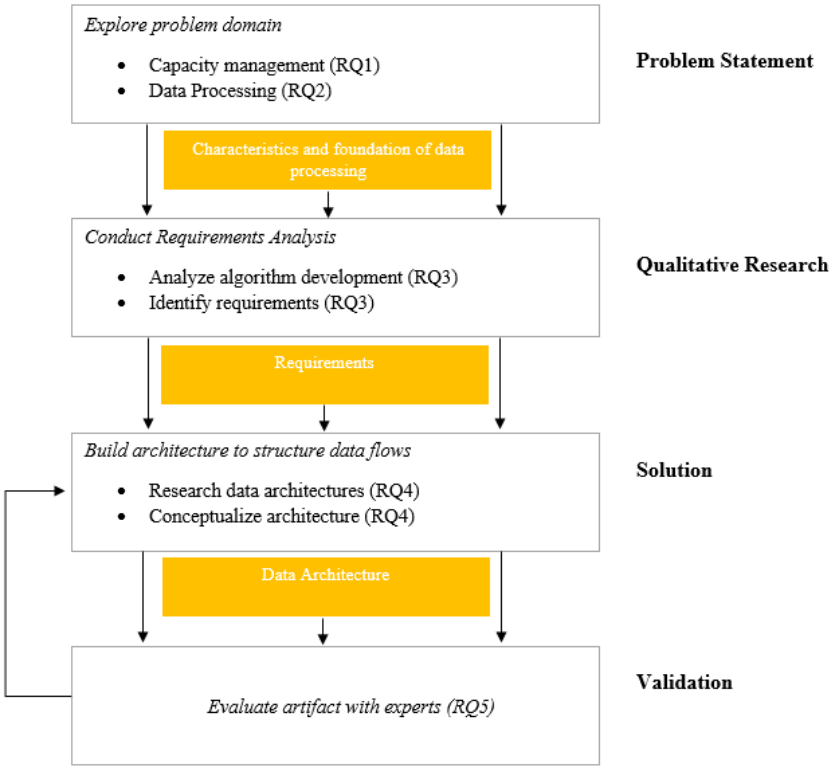

Figure 2.B Research Outline

2.3 Relevance

Scientific Relevance

The real time applications and the possibilities of applying machine learning methods in healthcare is growing. This phenomena requires an efficient data processing environment that guides data experts to create an efficient data flow. Almost every hospital has their own data environment that manages data flows, but practice learns that not every dataflow is created according to guidelines that should be predefined. These guidelines help data experts and the IT department to prevent creating several data processes that use their own technology, cleaning rules and integration rules. This research project aims at creating a unified data processing environment by dissecting different elements of a real time data proces, and structure them in

an architecture that can be used as a guideline to create dataflows. This unified data processing environment is important as it functions as a solid foundation for all the data projects, and can improve the collaboration between data experts and the IT department. Our architecture can also contribute to identify the data processing elements that need to be managed in a real time environment.

Social Relevance

This project is related to two social problems:

- Shortage of staff members in healthcare
- NICU capacity problems in the Netherlands

The NICU in Netherlands has been in the news since 2018. Pediatricians and gynecologists are worried about the capacity of the NICU's in the Netherlands. According to a news article published on the 28th of June 2019 [17] the pediatricians, gynecologists and other NICU specialists have met in the beginning of July last year, for an emergency consultation to discuss the current situation of every NICU in the country.

According to the medical experts in the city Hoorn, one pregnant woman a week (on an average) has to be admitted to another hospital because of expected complications with the baby, and the experts in Hoorn predict that almost 40% of the women can not be admitted to a NICU in their neighbourhood, thus they have to be admitted to NICU's far away. According to the chairman of the Nederlandse Vereniging voor Obstetrie en Gynaecologie (A Dutch association for Obstetrics and Gynecology) the magnitude of the problem is increasing by every year [17].

However, medical experts believe that the cause of this problem is not the NICU, but the shortage of staff members at the whole neonatology department. It is not easy to transfer a baby to high care or medium care if there are not enough staff members to take care of the babies [17].

In the Netherlands, the shortage of staff members in healthcare is only increasing. According to a news article published in november 2018 [45] there are 30.000 vacancies in healthcare and if nothing changes, in four years there will be a shortage of 125.000 people. This research project is related to Predict, that tries to optimize the capacity with algorithms. Our aim is to contribute to the back-end of Predict and help develop the real time data process.

Chapter 3

Data Processing foundations

Over the years, the healthcare sector has generated a large amount of data by storing patient care data and keeping records. However, a substantial part of this data is still stored in hardcopy form [18]. More healthcare providers are trying to adopt the trend of rapid digitization, and apply several machine learning techniques to derive new insights to improve the decision making of medical experts [18]. This transition to a more data driven healthcare environment will allow clinicians and data experts to embrace real time solutions, and use data analytics as an integral part of the care process [20]. However, there are some challenges the stakeholders need to take into account while moving towards data driven healthcare. For example, the datasets stored in the information systems are often so large and complex, that traditional data management tools are not capable enough to process the data [19]. These datasets are often stored in the EHR system of the hospital. Other challenges include the security of the data, the interoperability of the data sources and the collaboration between IT, data experts and clinicians [21].

In order to succesfully function in a data driven environment, stakeholders need a proper overview of how the data is processed in their organization, and how its managed in their organizational infrastructure. Data processing has evolved over time and in order to create the right method to manage the dataflows, several characteristics have to be taken into account that are listed below:

- Problem domain (case study)
- Business application the users need
- Current IT infrastructure
- Requirements stakeholders

In this chapter, we first discuss how data processing has evolved over the years, to help us identify the different tasks that need to be executed to create a sufficient data process. We research two ends of the spectrum and start off with an overview of the traditional method of data processing called data warehousing. While data warehousing is focussed on copying and migrating data physically, other methods focus on a more real time approach by processing the data while providing data to the business application with views like data virtualization. We also present an overview of the knowledge discovery model, Cross-Industry Discovery in Databases model (*next: CRISP DM*) to provide a more in-depth analysis about data processing activities. Next, we research the impact of real time demand in the healthcare sector and dive deeper into the operability challenges of the information sytems in healthcare that influence the data process methods.

3.1 Data Processing over time

3.1.1 Data Warehousing and Extract-Transform-Load

In this section we want to explore how data processing over time has changed. The reason behind researching this topic are threefold:

- We want to understand the current data environment of the NICU
- We want to research the tasks that need to be executed during the data process in a data warehouse setting
- We want to explore the options within data warehousing for Predict to eventually build and execute their real time data process

An overview of the data environment at NICU is presented in figure 3.A. As discussed in chapter 1, the primary data sources used at the NICU are HIX and MetaVision. For Predict, a third data source called MOSOS is included. This data source was used a couple of years ago to store pregnancy data and currently contains relevant data for the algorithm. However, it is not used actively in the current data environment and therefore not included in the figure.

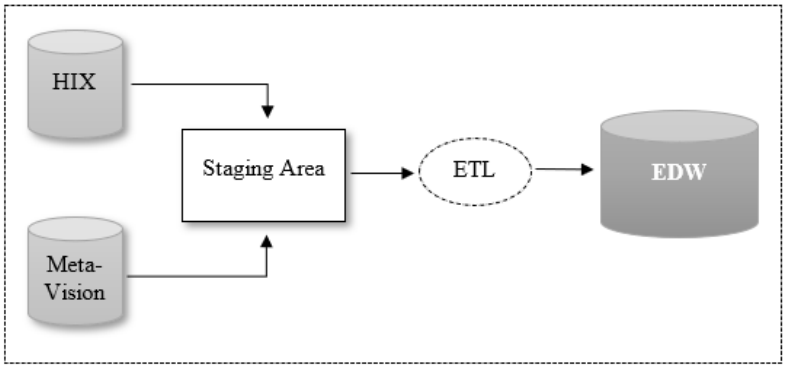

Figure 3.A Data environment NICU [7]

We discussed the data sources in chapter 1. The data environment of NICU also includes an Enterprise Data Warehouse (*next: EDW*). This data warehouse includes copies of the source tables. Data is extracted from HIX and MetaVision, and is first stored at the staging area. In this phase, the data is transformed according to the business rules using ETL. When the data adheres to the business rules, it is ready to be loaded into the EDW.

Extract-Transform-Load (*next: ETL*) is a traditional approach of data processing, and functions as a core aspect of data warehousing [22]. Data warehousing is defined as "*a subject oriented, integrated, time variant, non volatile collection of data in support of managements decision making process*" [36]. The concept of data warehousing started in the early 70's where the data was stored in early-generation databases and was used to create routine reports with the business applications at hand [23]. Over time, data warehousing has evolved by developing the several challenges that business users faced in the early days, like accessing and aggregating the data that are stored in different sources. One of the fundamental changes in data warehousing development was the decision to copy data, instead of directly accessing files and databases. This method creates a chain of databases where data migrates from one database to another while executing transformation tasks [24] [46] [48]. Every transformation process executes cleaning and integration tasks to eventually load the data into the next database. This process is known as the ETL process and is known to be long, complex and highly connected, making it a difficult task to timely implement a simple change in the data process [24]. Figure 3.B presents a more detailed overview of a general data warehousing system that includes the tasks executed by the ETL tool.

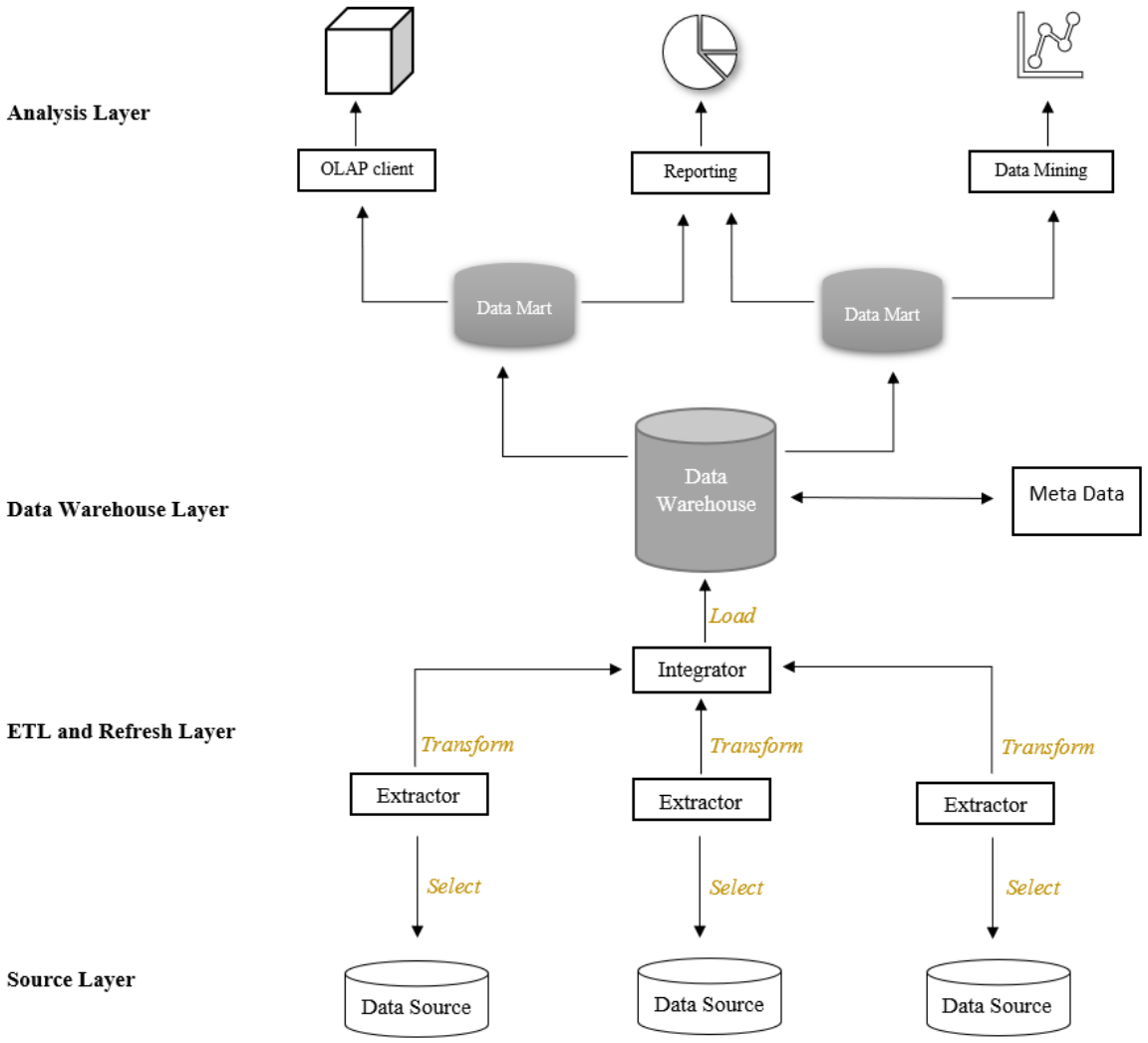

Figure 3.B Data warehouse framework [37]

The extractor executes the extraction and cleaning tasks while the integrator performs the integration of the data. The extractor also functions as a monitor that manages the source updates. There are two types of extraction, the first one is the initial extraction where the data is being extracted and loaded into the data warehouse for the first time. The second type is the incremental extraction, also refered as changed data capture. Incremental extraction is executed periodically according to the business needs and covers the changes in the datasources [22]. The data warehouse itself is a repository where the data is stored to support decision making and is defined as *"a system that retrieves and consolidates data periodically from the source systems into a dimensional or normalized data store. It usually keeps years of history and it is queried for business intelligence or other analytical activities. It is usually updated in batches, not every time a transaction happens in the source systems"* [25].

ETL and data warehousing are predominantly used to process batch data. ETL is used to clean and integrate data and migrates it to a physical database. However, Predict needs a data processing environment that supports real time processing of the tasks. ETL has modified its technology to satisfy the real time processing requirement with the introduction of near real time ETL. Before we dive into this topic we would like to address some changes in the data environments that affect building a new data process in healthcare.

Figure 3.B includes components like:

- Data sources
- ETL
- Applications for stakeholders

These components have changed over the years because of the massive growth of data, the increasing complexity of data sets and the demand of real time data. The amount of data that organizations generate these days has increased since the early 2000's and it even introduced us to a new term: Big Data. *"Big data is a term for massive data sets having large, more varied and complex structure with the difficulties of storing, analyzing and visualizing for further processes or results"* [26]. Big data also resulted in more complex datasets because the amount of data sources that were explored increased as well. Both structured and unstructured data had to be processed. Lastly, the applications that stakeholders use for decision making are also changing. There are occasions where the stakeholders want to explore the data in greater detail, and add machine learning techniques to their applications. Therefore, it is important for the business users to define the depth of the analysis, prior to the development of the business application because it can influence the data process.

As we have mentioned at the start of this chapter, the abovementioned changes are also noticable in healthcare. This sector is mainly still exploring the advantages of adding data analytics in their workflow, and modifying their data environment to extend the amount of data applications in the hospitals. Processing a large amount of data is not relevant in our project. However, we do have to take into account how to manage complexity in data sets, the cleaning and integration tasks, and including an algorithm in the new real time data process.

3.1.2 Data Processing tasks ETL

This subsection explores the tasks that are executed during the ETL process and how they should be managed within a data flow. These tasks need to be integrated in the architecture that we propose in chapter 5.

Transformation and cleaning

The transformation and cleaning process is executed at the staging area in a data warehousing system. The main objective of this phase is to make sure that the data matches the quality standards set by the business users. There is a difference between data cleaning and data transformation. Data cleaning means removing unwanted data from your data set in order to improve the quality of the data [33]. The data can include missing data or errors in the data, which are difficult to avoid because the root of the problem is at the front-end process. Data experts try to manage the inconsistency by adding restrictions to the fields for the front-end users. Moreover, other challenges include managing data sources that are integrated but produce duplicate records known as the merge/purge problem where different records in different data formats refer to the same entity. There is also a lack of knowledge when it comes to detecting and correcting errors [34] [35]. In Predict, the data sources that are used to create the algorithm also include missing data and errors. During the algorithm development phase, the data experts had enough time to manage this by creating new tables manually that contain the correct data. However, in the real time process, we have to take into account that these tasks need to be executed automatically and need predefined rules. The same goes for data transformation.

Data transformation refers to formatting and structuring the data in order to match the data warehouse schema [47]. A data warehouse schema is a logical representation of how the tables from the data sources should be structured in a data warehouse [36] [37].

Data quality

Data cleaning and data transformation are used to increase the data quality of the extracted data. Optimizing the data quality increases the reliability of information that is provided at the end of the data processing chain. However, the data in hospitals are primarily generated by the EHR, and the business users are the clinicians. Because this process includes the human factor, typographical errors or using different terminologies for the same concept are bound to be made [9]. According to a study, small data errors could affect the outcome of a data analysis, implying that the infrequency of data quality could compromise the operational decision making process [38].

Literature specifies different dimensions of data quality to measure the data. Table 1 provides an overview of these dimensions [39] [40] [41].

Dimension	Meaning
Accuracy & validity	Accuracy and validity refers to the syntactic and semantic correctness of the values, in other words, the values represent the real world object/ events in a correct manner.
Reliability	Reliability refers to the consistency of the data, in other words, an outcome of a data analysis should show the same results when repeated.
Completeness	Completeness refers to the amount of missing data in a data set and the impact of this occurence.
Currency & timeliness	Currency and timeliness refer to the amount of time it takes to update the data source and process the data to get the right information.
Usability & Interpretability	Usability & Interpretability refers to the representation of the values in such a manner that factors like ambiguity for example will not affect the interpretation of the data.
Availability & Accessibility	Availability & Accessibility refer to the amount of effort it takes to access the data, and if the correct data is also available in the data source when needed.

Table 1. Data quality dimensions [39] [40] [41]

There are several ways to manage the challenges that arise when upholding the data quality. An organization could develop procedures to monitor the data quality, or implement standards that focus on the documentation, processing and maintenance of data in healthcare. Therefore, some countries introduced minimum data sets and data dictionaries to develop these standards. An example of such a data set is the Uniform Hospital Discharge Data set, used in Australia and the United States of America to collect data elements [42] [40].

The main objective of a minimum data set and data dictionaries is to collect uniform data that multiple organizations can utilize [40] [42].

Literature also describes several methodologies for data quality assessment and improvement. We will not dive into this topic in too much detail for this research, but we will address some methodologies while creating our architecture in chapter 5.

3.1.3 Near real time ETL

As we have mentioned before, the changing environment healthcare organizations are operating in, is requesting for real time data. We want to explore the concept of near real time ETL to gain more insight in real time data processing. This will help us in identifying components to build our architecture. Another reason is researching whether this method is useful to eventually develop the data process for Predict.

In near real time ETL, the main objective is to shorten the data warehousing loading cycles. At the UMC, the loading cycle is executed at off peak hours to avoid overloading the data source. This usually results in a data latency of 24 hours or more. In near real time ETL, the aim is to move less data faster and a more frequent rate by allowing the applications that use ETL tools to perform light weight transformations. From a technical perspective, practice shows that it is not easy to develop this system and most solutions are created ad hoc and do not follow a unified approach [50] [51] [52]. Literature specifies a general architecture presented in figure 3.C to support near real time ETL [50].

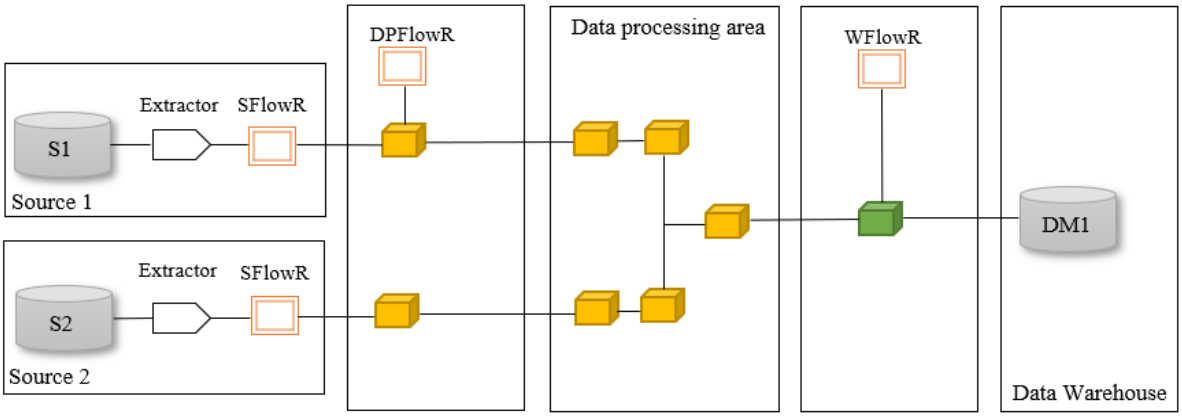

Figure 3.C Architecture Near Real Time ETL [50]

The yellow squares in the figure indicate the data that is transformed. Furthermore, the architecture mainly consists of three elements:

- The data sources
- The data processing area where the cleaning and transformation takes place
- The data warehouse

Each source includes a source flow regulator (SFlowR) that manages the relevant changes in the data source and propagates them towards the data warehouse at specific time intervals. Between the source and the data processing area resides the data processing flow regulator

(DPFlowR) that decides which data source is transmission ready, and by using a simple selection mechanism they decide the data that has to be processed by the ETL workflow. The data processing area is responsible for cleaning and transforming the data, and eventually the data is loaded into the data warehouse. A warehouse flow regulator (WFlowR) organizes the migration of data from the data processing area to the data warehouse. The flow regulator is based on the workload and the requirement of the business user regarding data latency [50].

However, there are still some technical issues that need to be resolved to efficiently execute near real time ETL. One of them is related to the sources that need to identify the data that must be passed on to the data processing area. In order to achieve this, a mechanism has to be developed that identifies changes and propagate data at a certain frequency, but at the same time does not overload the other responsibilities of the data source [50]. There are several applications on the market that offer near real time technology, or a variation like Distributed On-Demand ETL that combines several strategies to achieve near real time ETL [52].

In summary, ETL has evolved from physically migrating batch data to a method that fits the need of processing smaller batches of data in real time. The management of the cleaning and transformation tasks will be processed in our architecture. Furthermore, data experts need to take into account the changing data environment they operate in. Data sources, data structures and the business applications can change over time. We will try to take the changing nature of a data environment into account while conceptualizing our architecture. Lastly, we explored near real time ETL that seems like a convenient option for Predict once the algorithm starts running in a clinical environment. It still depends on the requirements of the data experts which technology to use, which we will explore in chapter 4.

3.2 Data Virtualization

As we mentioned at the start of this chapter, we research two ends of the spectrum regarding data processing. In this section, we will dive into data virtualization, a concept that has gained popularity within the Predict team because they prefer to use this technology to create the real time data flow once the algorithm starts running. Another reason for exploring this technology is to research how real time dataflows are created and managed.

Data virtualization is a relatively new term that encapsulates multiple (existing) concepts that have changed the architecture of data processing. The term is based on the word ‘virtualization’, a term that has been around for years. The basic idea of virtualization is that business applications, which we will refer to as data consumer in this section, can use resources without knowing where these resources are located or which platform the resources use [24]. A frequent use of virtualization in practice are *virtual machines* where software allows the hardware to emulate several operating systems by using an abstraction layer [53].

In data virtualization, the hidden resource is the data that is needed to feed the data consumer. An abstraction layer hides the details about how and where the data is stored [24]**.** Figuur 3.D provides a high level overview of the use of data virtualization. As indicated in the figure, the content of the sources can have different structures, and different technical details that need to be managed by the virtualization server to integrate and manage the data.

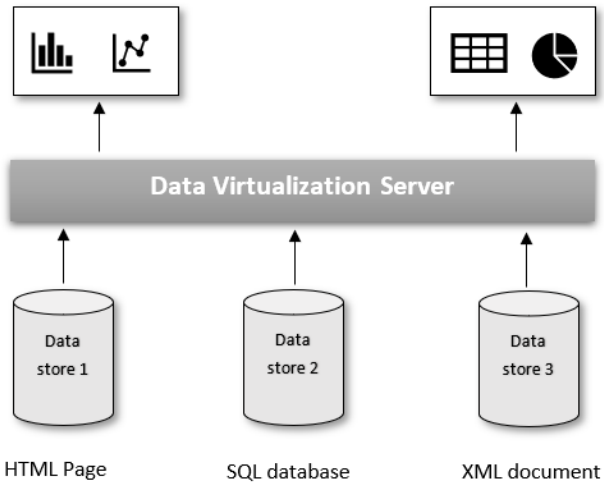

Figure 3.D High level overview data virtualization [24]

The basic idea of this technology is to let the data consumer view the data as one integrated source, and use a simple interface to access the data, through the virtualization layer.

Related concepts to data virtualization

As mentioned at the start of this section, data virtualization is an amalgamation of several concepts that provide tools and techniques to process data in a virtualized environment.

- Encapsulation (information hiding)

According to Rumbaugh *"encapsulation (also known as information hiding) prevents clients from seeing its inside view, where the behavior of the abstraction is implemented"* [55] [54]. When we apply this definition to data processing, it focusses on hiding technical details like where the data is stored, the format of the data or which API to use for data extraction. The technology also contributes to the independence of the data consumers development. In other words, when the structure of the data changes in the data source, it does not mean the data consumer has to change as well because the technology allows to seperate the business consumer and the process of extracting data from the data source. An example of encapsulation is the use of SQL. While writing a query, the user does not have to specify how the data must be retrieved but only specifies what data is needed [24].

- Abstraction

Abstraction is defined by D.T. Ross as *"a process whereby we identify the important aspects of a phenomenon and ignore its details"* [56]. Data abstraction in database management systems is decomposed in three levels as presented in figure 3.E.

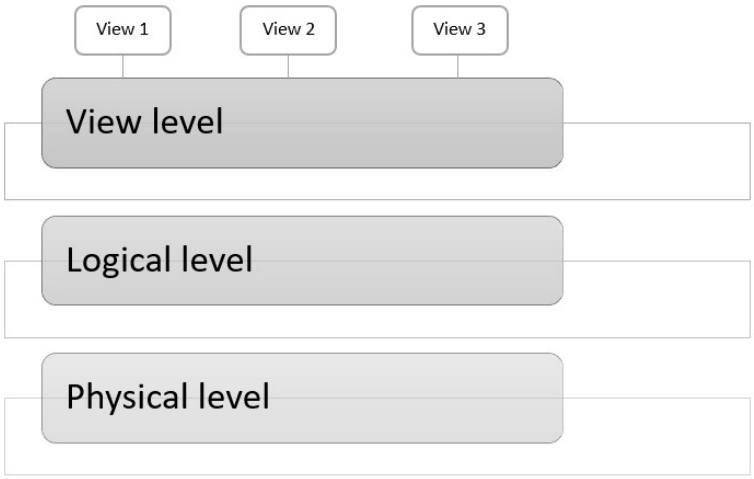

Figure 3.E Three levels of data abstraction

The physical level indicates the physical storage of the data. This could be data that is stored in files in a centralized database or stored in the cloud. The logical level is defined as the conceptual level of the data. In this level, the structure of the data is defined and the relationships among the different entities are specified. The view level is mainly important for the user. The view level provides the user with the data that has been requested through the interface. In data virtualization, the abstraction part is executed with views [24] [57].

Abstraction and encapsulation are closely related and both concepts create data independence. Creating data independence is an important concept to take into account while conceptualizing our architecture. The biggest advantage of creating data independence is that changes in the data processing area or data sources do not have to impact the whole data flow.

- Data integration and data federation

The last two concepts, data federation and data integration are often used as synonyms for data virtualization. However, both concepts are a part of data virtualization where in data federation the main objective is to create a virtual database and in data integration the main tasks are focussed on integrating heterogeneous data sets to create a unified view [24] [58].

Building blocks of data virtualization

Data virtualization is defined as *"a technology that offers data consumers a unified, abstracted, and encapsulated view for querying and manipulating data stored in a heterogeneous set of data stores"* [24]. We will introduce the building blocks of data virtualiation to provide an insight in how the technology works. We will specifically dive into the architecture of the data virtualization server which is the layer between the data sources and the data consumer.

A data virtualization server consists of three building blocks:

- Virtual table
- Mapping
- Source tables (in several formats like spreadsheets, HTML, CSV)

Figure 3.F presents the components of the data virtualization server. The virtual table is the returned view of the data consumers request for data. The source table is where the data actually resides and where the data is also accessed.

The source tables are not managed or defined in the data virtualization server. Our third building block is mapping. Mapping defines the transformation of the source table into a virtual table. In other words, the content of the virtual table is received by linking it to the source table through mapping. The tables are being accessed by using various API's.

Figure 3.F also includes a wrapper table. In order to let the source table assist the virtual table, it has to be imported first for the data virtualization server to become aware of this table. After importing the source table, a wrapper table is created by the server. The wrapper table is defined by the metadata of the source table. Important to note is that the data virtualization server checks whether values have to be transformed to a more standardized form while importing the source table. The content of the wrapper table is identical to the source table [24].

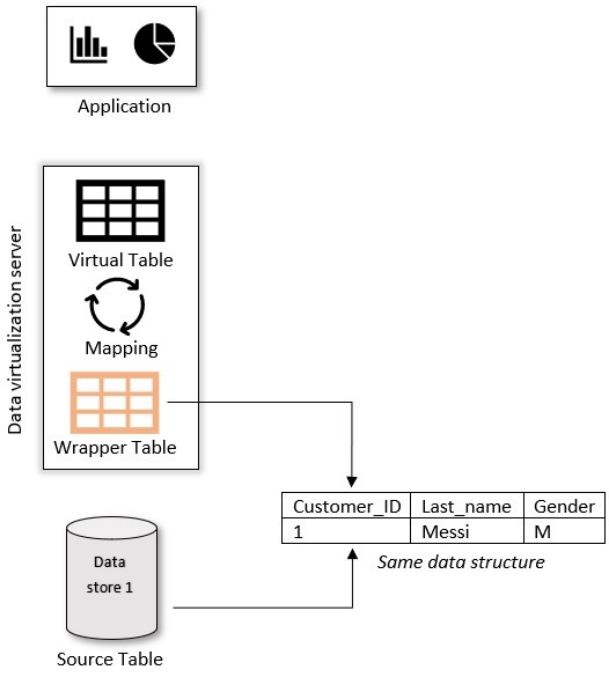

Figure 3.F Components Data Virtualization server [24]

Mapping

The wrapper table has the same data structure as the source table. But the application could have a particular requirement to see some tables joined or some rows aggregated. To satisfy this requirement, a virtual table needs to be defined on top of the wrapper table. Defining the structure of a virtual table is executed by mapping. Mapping decides what data, and how the data should be transformed while flowing from source table to virtual table. The mapping process is similar to the view concept in SQL, where in data virtualization the query definition (view concept) is called mapping [24].

Virtual Table

There are data virtualization servers that allow unbound virtual tables. These are tables without mapping or any link to a source table. In other words, the structure of the virtual table is defined first, the mapping can be added afterwards. The advantage of this top down approach is allowing the developer to become independent from the data structure of the source table. This approach is mainly effective when the data in the source tables are not well structured. Another advantage is increasing the flexibility of the virtual table, for example, when the data is moved to another database only the mapping has to be redirected [24].

In summary, dissecting the data virtualization method has provided us more insight about building a real time data flow. The method describes the elements that are used to build a data flow. The virtual table is used to define the table that contains the correct data to calculate the risks, and is specified by data maps that provide an overview of how the data is transformed. Both concepts are useful to incorporate in our own architecture. Lastly, when building our architecture, we want to explore the concept of seperating the dashboard from the data processing area to not only support the concept of information hiding but also reduce the impact of future changes on the whole data process chain.

3.3 Cross-Industry Discovery in Databases model

The previous sections have resulted in a better understanding of how to build a real time process and manage the several tasks during this process. The next chapter is going to discuss the development of the Predict algorithm mainly in terms of how the data flows. We have conducted interviews with data experts to research the algorithm development. In order to structure our interviews, and better understand how data is transformed from raw input to information, we have decided to use CRISP DM as a reference.

CRISP DM is a methodology developed for planning and structuring data mining projects. It includes tasks from the perspective of the data analyst. The methodology consists of a process model, also known as the reference model. This model consists of six phases and is presented in figure 3.G [73].

The CRISP DM process model presents the life cycle of a data mining project. The arrows in the model indicate the most frequent dependencies between the phases. However, according to the author you do not have to adhere to the order of the flows, as you can go back and forth between the phases [73].

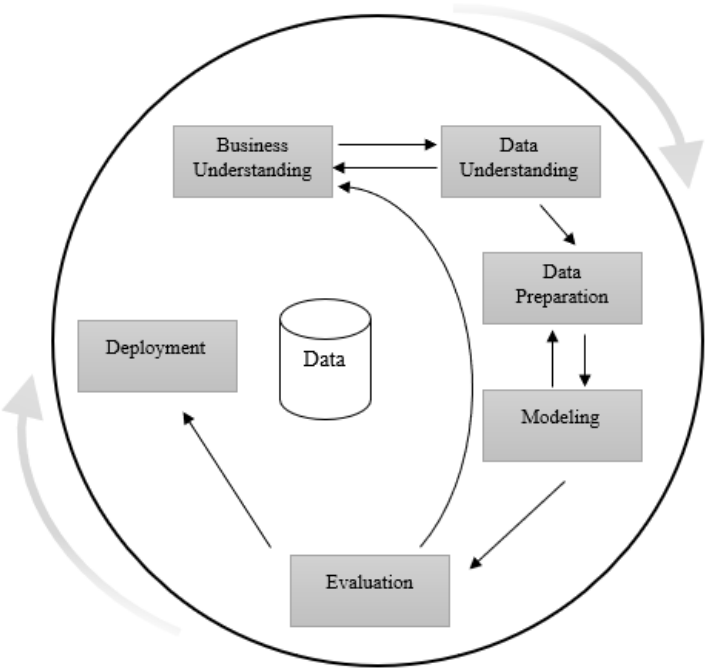

Figure 3.G CRISP DM Process Model [72]

As mentioned before, we want to use the process model to structure the content of our interviews. The main objective of our interviews is researching the process prior to creating the algorithm. In the CRISP DM process model, only the first three phases are focussed on this topic, therefore the remaining phases of the process model will not be included in this project. Literature provides an overview of the characteristics of the main phases in the model. Table 2 provides an overview of these characteristics.

Business Understanding	Data Understanding	Data Preparation
Determine Business Objectives *Background* *Business Objectives* *Business Success Criteria*	**Collect Initial Data** *Initial Data Collection Report*	*Data Set* *Data Set Description* **Select Data** *Rationale for Inclusion/ Exclusion*
Assess Situation *Inventory of Resources* *Requirements, Assumptions, and Constraints* *Risk and Contingencies* *Terminology* *Costs and Benefits*	**Describe Data** *Data Description Report*	**Clean Data** *Data Cleaning Report*
Determine Data Mining Goals *Data Mining Goals* *Data Mining Success Criteria*	**Explore Data** *Data Exploration Report*	**Construct Data** *Derived Attributes* *Generated Records*
Product Project Plan *Project Plan* *Initial Assessment of Tools and Techniques*	**Verify Data Quality** *Data Quality Report*	**Integrate Data** *Merged Data* **Format Data** *Reformatted Data*

Table 2. Generic Tasks main phases CRISP DM (specified for our project) [72]

Business understanding

The first phase of CRISP DM is focussed on understanding the project environment, defining the objectives and translating them into project goals taking the organizational resources into account. First, an assessment of the current situation needs to be performed. Exploring the data sources, the people involved (data experts, business experts and IT department) and analyze the technical aspects like software and hardware platforms or data mining tools that are available. Second, the project members have to specify data mining goals that relate to the business goals, and satisfy the business requirements. The last step is to produce a project plan that includes a time span of the different tasks that need to be executed, the resources that are available and a detailed description of every project phase [73].

Data understanding

The second phase of CRISP DM is focussed on understanding the data that is needed to create the algorithm. The first step in this phase is initial data collection and executing activities that are essential to become familiar with the data. Next, the data experts perform a quality analysis to research the current quality of the data, and how it will impact the output of the algorithm. Lastly, the data is also checked to identify other relevant insights that could contribute to the main objective of the project [73].

Data preparation

The third phase is the data preparation phase. This phase includes all the acitivies that are needed to make sure the data is constructed in such a manner that it is ready to be included in the algorithm. Tasks that are related to this activity are cleaning, integration and transforming data to create the right data sets [73].

We will use the abovementioned phases to describe the algorithm development in chapter 4.

3.4 Healthcare information systems & interoperability

Getting the right information at the right time is a valuable research area for hospitals to explore as it can increase the quality of healthcare. As mentioned before, hospitals generate a lot of valuable data on a daily basis and connecting this data through several IT solutions can be very beneficial for a hospital. It can reduce medical errors, encourage the collaboration between healthcare providers and develop personalized healthcare [76].

However, the data is stored in different information systems and they are not connected to each other. The same problem arises in Predict. Predict is mainly using HIX and MOSOS to extract the data for the algorithm development and real time process. In the future, they want to expand their data environment and add new data sources. These datasources are not connected to each other and during the algorithm development, the data had to be integrated manually. In this section, we want to explore the interoperability challenges proposed in literature.

Interoperability is defined as *"the ability of two or more systems or components to exchange information and use the information that has been exchanged"* [80]. Because of the lack of interoperability between several information systems in healthcare, data integration becomes a challenge. Data integration is defined as *"a set of techniques that enable building systems geared for flexible sharing and integration of data across multiple autonomous data providers"* [79]. Most of the data sources are developed independently, which results in differences in the

technical characteristics of these sources. The sources run on different platforms, they often have their own database schema that includes their own data specifications and the sources could contain structured, semi-structured and unstructured data. In healthcare, around 80% of the data is unstructured (image, signals, video) and not utilized properly for analysis [30] [31] [32]. Moreover, the amount of different files that are stored in the information systems are very diverse, for example:

- *Clinical notes*
  These could be handwritten or stored in a free text area in the EHR.
- *X-rays or image reports*
  Images in healthcare are mostly stored as DICOM files. DICOM stands for Digital Imaging and Communications in Medicine and is a common standard for conversing images in healthcare.
- *Heart video*
  For some treatments, the hospital also use video files.

This information is stored in a heterogeneous way looking at the terminology, syntax, semantics, data types and data format.

Furthermore, healthcare is a complex domain with a lot of medical experts that are involved in one treatment. They all have their own way of sharing information. The amount of multiple representations for the same clinical concepts are growing, therefore the need for a standard definition for terminologies is important. One common standard used in practice is called Health Level Seven *(next: HL7)*. HL7 is a worldwide standard that supports safe electronic data exchange [76] [77] [78] [104].

In summary, we have mentioned several examples in this subsection that address the challenges of data integration in healthcare. There are several data integration architectures, but generally they can be classified into two different approaches, warehousing and virtualization, which we discussed in the beginning of this chapter. In our final architecture, we need to take into account the technical differences between the data sources, and the several data types that need to be processed. When the algorithm is developed, the data integration is executed manually but when the algorithm starts running this task has to be automated. There are standards available that the data experts can use but after a thorough data analysis, a data expert can create their own standards customized to Predict data. Another important aspect is the language that is used while extracting data. In practice, SQL is the primary language while extracting data but often while using other sources like social media, the systems have to work with API's and internal data models to access and integrate data [30].

3.5 Real Time data processing in healthcare

Predict needs to process the most current pregnancy data to provide real time updates. Therefore, it is important that they have access to a real time data base that generates the most current data about the patients. Healthcare is also progressing when it comes to the adoption of real time data. In this subsection we would like to present the advantages of real time applications in healthcare for the patients and the clinicians. As we mentioned in the introduction of this chapter, healthcare organizations collect a lot of data stored in EHR's, but they also use other types like social media and sensor data that is generated during treatment. This has resulted in more advanced opportunities for the healthcare sector to analyze data and

transition to personalized healthcare and patient centered care *(next: PCC)* [63]. However, PCC has been around for a while. The advantages of adopting PCC are reducing healthcare costs, improve data exchange, a decrease in the average length of stay of the patients and overal a higher patient satisfaction. In personalized healthcare, the treatment, diagnosis and medicines are customized to the patient and their medical history [64] [65] [66] [67] [68]. Another way to contribute to more personalized healthcare is using wearable devices to monitor patients and help them with their diagnosis. The incoming data can be analyzed in real time and can be combined with their medical history to customize the treatment plan [63].

The abovementioned examples are patient centered. Real time data could also manage the workload of the medical experts. An example is speech technology in healthcare which can be used to convert speech to text without the use of someone who is transcribing the dictation, but only a medical expert who is dictating to a device [69].

However, there are some challenges the healthcare sector needs to solve to succesfully implement real time data analytics. There is still a lack of knowledge and experience when it comes to utilizing real time data analytics in a healthcare environment which sometimes leads to opting for short term solutions to make a project succeed. Another challenge is managing the data quality to make sure the right data is used for decision making [71]. Therefore, we aim to encourage the hospital to analyze their current data environment and start building a unified data environment that supports real time data projects in order for them to coexist with the batch projects. We also want to encourage the activities that need to be executed (cleaning and integration) to provide an efficient and qualitative data flow to make sure that the data at the end of the data flow chain is correct.

Chapter 4

Requirements Analysis

In order to understand and define what the data process should look like once the algorithm starts running, we are going to conduct a requirements analysis. The main objective of this analysis is to dive deeper into the several components of a data process like extraction, pre-processing and integration to find out the requirements that are needed in the algorithm production phase.

A requirement is defined as a condition or capability needed by a user to solve a problem or achieve an objective [81]. There are different types of requirements:

- Business requirements
- User requirements
- Software requirements

In this project we do not develop an information system, but we do want to specify requirements for the future system that manages the data process of Predict. Therefore, we will elicitate functional and non-functional software requirements. The functional and non-functional requirements describe what is expected from the real time process and how to maintain the quality of the process [106].

Furthermore, the main objective of the data architecture is to guide the data process to extract the right data, and process this data into the correct format to provide input for the algorithm. The clinicians perspective is left out of this analysis because their requirements are already covered by the Predict team while creating the algorithm and the dashboard. We will focus on the requirements from the IT and data experts (data manager and data architect) perspective.

The first subsection presents an overview of how the algorithm was developed, we will not go into details of the algorithm development because that is out of the scope of this research. However, it is required to explain the basic elements of the algorithm to better understand the process of data processing in the Predict environment. We eventually deliver an algorithm development process (*next: AD process*) that will help us in elicitating the requirements. The second subsection of this chapter will identify the different tasks that need to be executed in the data process, which will help us in chapter 5 when we conceptualize the architecture.

In order to execute the requirements analysis we conducted semi – structured interviews with the data architect, data manager and other data experts to gain their perspective on the development and production phase of the algorithm. To structure this interview we have used the activities described in CRISP DM as guidelines. The interview protocol can be found in Appendix A. In total, four interviewees were selected and they were interviewed more than once as the project was progressing.

The interviews were recorded, and the audio files were transcribed manually and coded. Coding the transcriptions contributed to organizing the data, and selecting the important topics for analysis. The transcripts are not added to this thesis because of privacy reasons. The four interviewees were selected based on their expertise and involvement of the Predict project.

Moreover, the interviews covered the following main topics:

- Algorithm development
- Data Extraction
- Data Integration
- Data Quality
- Data Management

4.1 Algorithm development

We will describe the development of the algorithm accordingly to the CRISP DM methodology as described in chapter 3. We will not take all the characteristics of this methodology into account. We will use the characteristics that give us sufficient understanding of how the algorithm is developed. A part of the business understanding phase is already presented in chapter 1. Figure 4.A presents an overview of the characteristics that will help us explain the algorithm development.

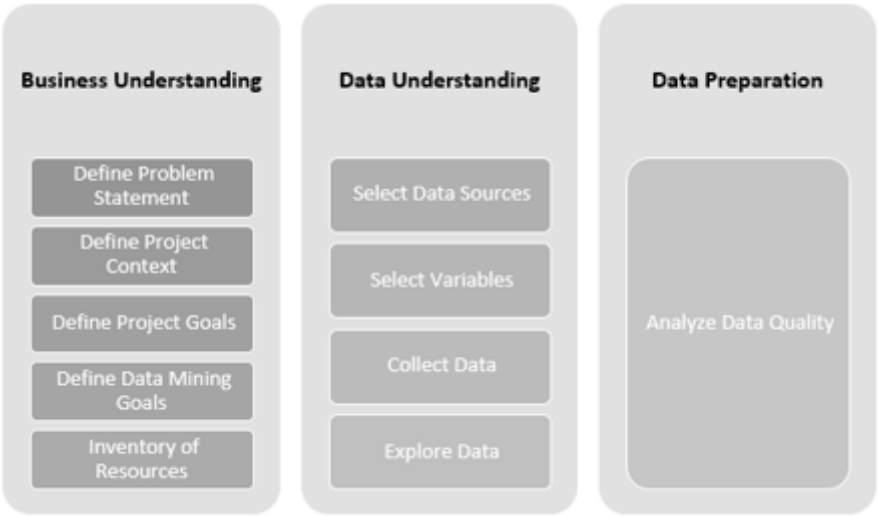

Figure 4.A Framework algorithm development

As we mentioned in the introduction of this chapter, the algorithm development is going to help us elicitate the requirements. The AD process will be used during interviews as a framework to help identify the tasks that need to be automated.

4.1.1 Business understanding

Data mining goals

There is a difference between project goals and data mining goals. For example, Predicts main objective from the clinicians perspective is to improve the capacity management in the birth centre. This could be identified as a project goal. To improve the capacity they want to use an algorithm that predicts the incoming NICU babies within a certain timeframe. Predicting the incoming NICU babies is a data mining goal.

The data mining goals are focussed on the specific output of the algorithm and how it contributes to the business objective. This part is complex because the data scientist has to be

very clear in how the output of the algorithm is going to provide the right information to the clinician. Exploring the project environment, and the clinicians work routine contributes to this step. When the data scientist is aware of the clinicians work environment, daily tasks and the challenges, it becomes easier to visualize how the output of the algorithm is going to have an effect on the clinicians decision making process.

At the moment, Predicts data mining goal is to predict the incoming NICU babies within a certain time frame. In the future, the data experts will research more possibilities regarding the algorithm and dashboard [13].

We have presented the inventory of resources in chapter 1 and 3. In chapter 1 we introduced the data sources:

- HIX
- MOSOS
- MetaVision

At the start of chapter 3, we provided an overview of the data environment of the NICU (figure 3.A).

Algorithm as a Medical Device

During the initial phase of the project, the data experts also investigated what type of predictive model would fit the nature of this project. In practice, when data experts develop algorithms to predict real time events, there is a possibility to feed the algorithm with real time data to improve the accuracy of the prediction over time. This is a form of real time machine learning. For example, an app could be created that uses machine learning techniques to gain more insight in the users behavior. While the app is generating real time data, this data could be fed to the algorithm simultaneously to make it improve itself [85]. An application that supports this method is Apache Kafka, an open source streaming data platform that provides streaming solutions to data experts [105]

However, there are certain rules regarding the use of Artifical Intelligence (*next: AI*) in healthcare. An algorithm, a form of AI, is considered to be a medical device within the healthcare sector. These devices have to adhere to specific rules before clinicians can use them. The rules for the European Union (*next: EU*) are specified in the EU Medical Device Regulation that consists of two regulations, the Regulation 2017/745 on medical devices (MDR) and Regulation 2017/746 on in vitro diagnostic medical devices (IVDR) [86].

According to the data experts of Predict, the algorithm can not be fed with real time data. When the algorithm is developed, it has to be checked and approved by an external committee before introducing it in practice. The reason for this approach is maintaining the traceability of the decisions the clinicians made based on the risk calculations. The clinicians will eventually work with a dashboard that provides them the results of the algorithm, and based on these results the clinicians take certain decisions. These decisions need to be traceable in the healthcare sector. Therefore, the algorithm can not include new data while it is running because the outcome of the algorithm could change over time if new data is included in real time. Taking this approach into account, figure 4.B presents a general overview of the data process including the algorithm [8].

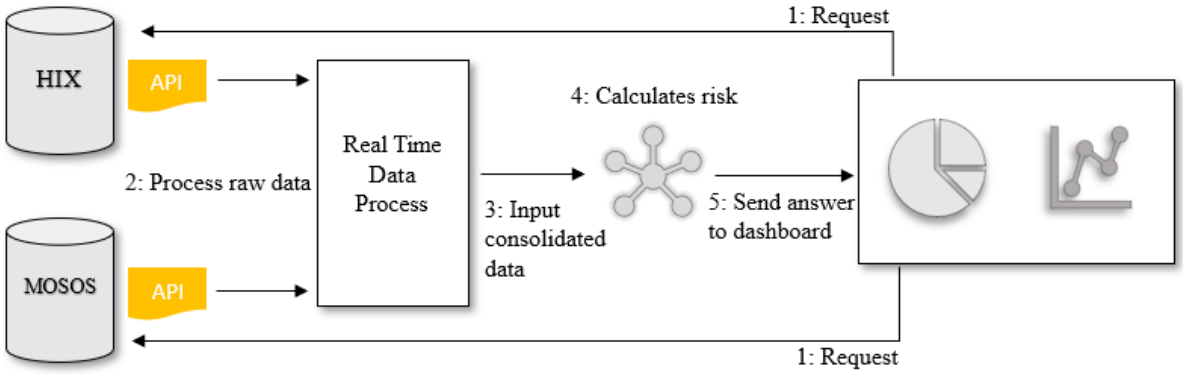

Figure 4.B Data process Predict

The dashboard calls the API's of the real time databases of HIX and MOSOS, requesting them for the variables. These variables are processed (cleaned and formatted) and result in a consolidated table (target table) that the algorithms needs to calculate the risks. The data process presented in figure 4.B will be taken into account while creating the architecture.

4.1.2 Data understanding algorithm

Within the Predict team, there are data managers and data scientists that collaborate to create the algorithm. In the first phase of the Predict project, the data experts collaborated with the clinicians to research what data is needed to predict the risks of the patient. The amount of data sources were also defined in this phase. The data experts decided to extract as much data as possible in the beginning of the project, and later on they would analyze the quality of the variables. This process was an iterative process. The amount of variables that are included in the algorithm increased as the project was progressing [7] [8] [9] [13].

The data that is needed to complete the algorithm is both structured and unstructured (data stored in free text). The data scientists decided to first extract the structured data and create the algorithm. They will modify the algorithm in the future with data mining methods for the unstructured data part. The data experts have already performed a quick analysis regarding the data and are aware of the importance of the unstructured data, and how it could improve the accuracy of the algorithm [7] [8] [9] [13].

The data needed to create the datasets are extracted primarily from HIX and MOSOS. Prior to extracting the data, the data experts defined the entities that need to be connected in order to create data sets [7] [8] [9] [13].

The next section elaborates on how the algorithm is developed in terms of dividing and executing the tasks. This results in the AD process that presents the data flow of the raw data into consolidated data sets, ready to be included in the algorithm.

4.2 Requirements

4.2.1 AD Process

The interviews with the data experts provide us with enough content to create a data process that presents the data flow of how the algorithm development is organized within Predict. This process is called the AD process and is presented in figure 4.C. The AD process will help us identify the requirements by analyzing the different tasks that are distinguished in the AD process.

The data managers extract the data from the databases and create the different tables according their datamodel. After that, the data managers pseudonomize the data before it is loaded into the data lake. Secondly, the data manager has to map the variables back to the original datasource. The aim of this activity is to create an overview of the data flow (data transformation and its original location) of the variables. Finally, the data managers are also responsible for a part of the cleaning process:

- Checking for odd values (test patients for example)
- Integrating the data
- Making sure the tables contain primary keys, and can be connected to each other by the data scientist.

The second phase of preparing the data is performed by the data scientist. They extract the data from the data lake, connect the tables and improve the quality of the data. The data preparation mainly consists of two tasks. The first one is related to checking the medical correctness of the values. The second one is analyzing values that can be included in features. Many algorithms work with features to analyze which values to include in the algorithm.

For example, the data scientist needs one value that contains a blood pressure measurement of a patient, but there are ten different blood pressure measurements at different time intervals registered. The data scientist needs to investigate if they need all the values, or calculating the average will be sufficient. To solve this problem, features are created to decide which values improve the algorithms accuracy the most. When the data sets are finalized, the algorithm is developed, trained and evaluated for production [7] [8] [9] [13].

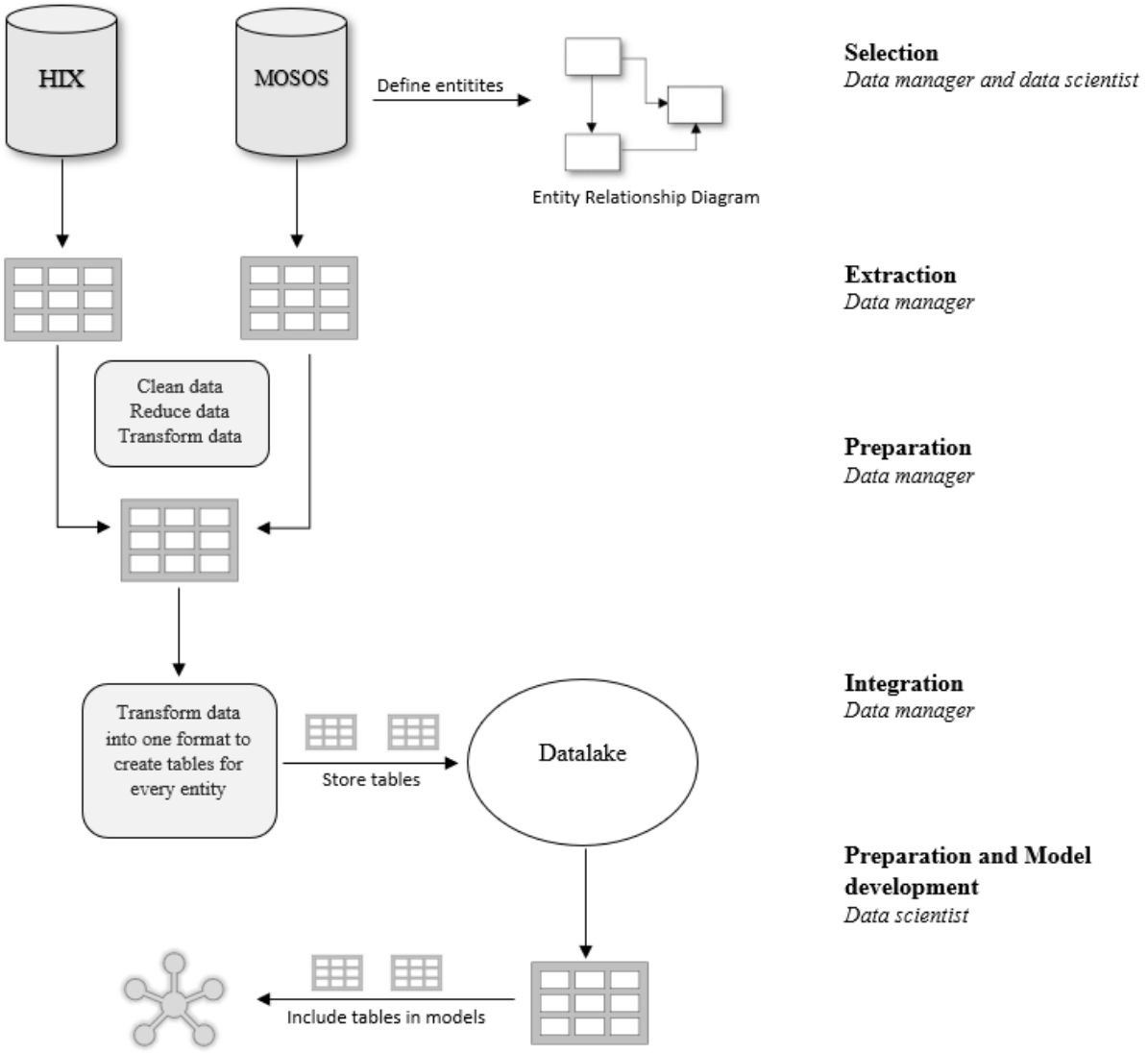

Figure 4.C AD Process [7] [8] [9] [13]

4.2.2 Requirements

Use Case

Before we dive into the requirements analysis, we want to explain a use case scenario of the expected real time process and how it will be executed in practice.

Scenario

The patient enters the hospital because of either complications with the pregnancy, or labour pain. The hospital registers the patients information and she gets admitted. When the patient is registered, the dashboard gets a notification that the API has to get the data from the sources to calculate the risk of delivering a NICU baby. This process happens in real time, in other words, the clinician needs the most updated data from the patient to be processed. The dashboard will eventually receive the information with the patients data, and the risk percentage of the baby in the womb. However, the API extracts the data of all the pregnant women that are currently in

the hospital at the maternity ward. This implies that the dashboard (that contains the data of all the women at the maternity ward) only refreshes when a pregnant woman enters the hospital. This method is also known as pull data. In the case of Predict, they use a pregnant woman entering the hospital as a trigger to refresh the dashboard. The opposite of pull data is push data where data is send to the user without them actually requesting for it. We will dive deeper into push and pull data further in this section [87].

Requirements

Based on the interviews and several meetings with the data experts, and data architect we have extracted eight requirements that are discussed below.

RQ1: Select variables that need to be extracted

Figure 4.C indicates the selection procedure as one of the initial phases in the algorithm development process. During the algorithm development, many variables had to be extracted and integrated. The data experts decided to create the algorithm in such a way that during the data process, the variables that are extracted for the algorithm do not have to be extracted again for the real time process. Thirteen variables are defined that will form the output of the extraction phase. In other words, these variables form one table and the values in the tables are the patients data. We will refer to this table as the target table. The target table contains consolidated data that will provide input for the algorithm.

RQ2: Extract the right data from a real time database

According to the data architect, the hospital has several data sources that contain copies of the HIX system. These copies can vary from weekly copies to a real time version with one second data latency. All these sources are used within the data environment of the hospital and result in dashboards and reports to manage the daily tasks of the employees. To satisfy the requirement of a dashboard update within a couple of seconds with the latest data, Predict needs access to the real time data warehouse of the hospital.

Moreover, to understand where the variables are stored in the datasources, data mapping is needed. Data mapping is used to create an overview of the origin of the values that are transformed and merged into a target table [88]. Figure 4.D presents a simple example of data mapping.

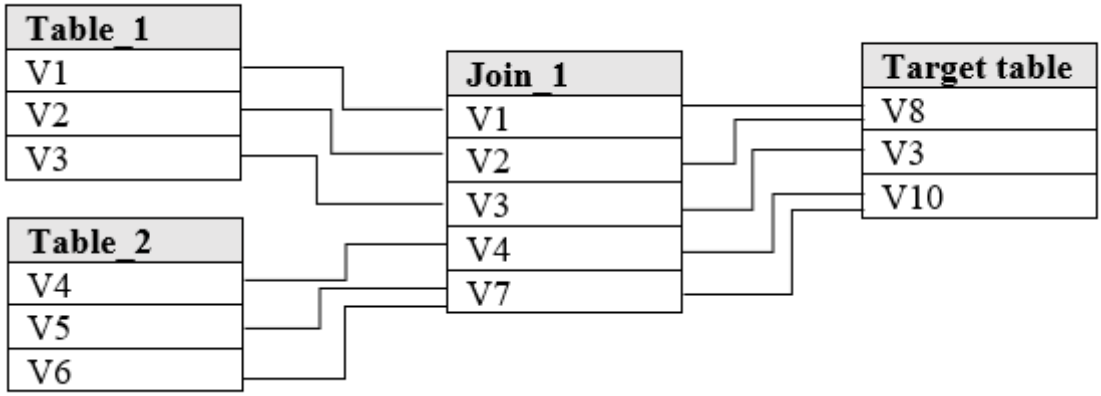

Figure 4.D Data mapping

The complexity of data mapping depends on the size of the data, the amount of sources and the structure of the data. Data mapping is essentially matching fields from data sources that contain raw data to target fields in a data warehouse for example. These data warehouses include a target data model that represents how the entities and variables need to be structured. In Predicts case, the output of the data extraction represents the desired data model that needs to function as input for the algorithm.

Data mapping has several advantages, it is mainly used to achieve data standardization and it is also useful to reduce the amount of data errors. Moreover, it supports data transformation and data integration. There are several techniques to perform data mapping. It can be executed manually by hand coding, but this approach becomes difficult as the complexity of the data, and the amount of data sources keep increasing. The next approach is semi-automated data mapping that involves building mappings between schemas. The last technique is automated data mapping, that revolves around data mapping tools that have built in features to convert data. In chapter 3 we discussed two different approaches of data processing, ETL and data virtualization. Data mapping is an essential part of these methods. Building a data map is a part of executing the ETL processes, and in data virtualization mapping is specified in the virtual table that needs to find its way back to the data source through the processing layers [24] [22].

RQ3: Extracted data needs to be integrated

The variables that need to be extracted reside in two data sources, HIX and MOSOS. In the future, new data sources will be added. HIX and MOSOS have different data models and during the algorithm development phase the integration was executed manually. Eventually the data integration tasks have to be executed automatically. Requirement 2 (mapping) could help the integration process, and bridge the difference in data models by specifying the transformation rules and analyzing the tables that need to be either joined, aggregated or filtered. To execute this task the data experts needs a thorough understanding of the structure of the data sources.

RQ4: Extracted data needs to be clean

In chapter 3 we have established the importance of data quality, and in this chapter we mentioned that one of the biggest challenges of the data experts was optimizing the data quality. The data experts were confronted with a lot of missing data and data errors. In order to improve the accuracy of the algorithm, the data needs to be clean and therefore usable to create input for the algorithm. Especially the missing data could become a big challenge. Extracting the right variables and creating the target table is mandatory for the algorithm. If one of these variables is not delivered, the algorithm can not calculate the risk. A part of this challenge is at the front end, the clinicians that register the data. This front end problem is difficult to manage for the data experts, however there are ways to manage cleaning tasks in the data process which we will discuss in the next chapter while defining the architecture.

RQ5: Algorithm and application need to be independent

The hospital is familiar with processing data in real time. At the moment, they have a couple of real time applications running in the hospital. However, working with algorithms in real time is new for the data, and IT experts. The difference lies in the responsibilities you assign to the algorithm and the application (dashboard) used by the clinicians. Figure 4.E presents an example of assigning a specific responsibility.

Figure 4.E presents four elements that could contain specific responsibilites like:

- Data cleaning
- Data integration
- Features

The aim of assigning these responsibilities is to make sure that future changes are efficiently incorporated in the data flow chain. For example, the responsibility of integrating data is included in the algorithm, and the structure of the data changes over time. This results in one of the variables needed for the target table, is now located in another table. Because the integration of data is included in the algorithm, the data experts have to change this component in order to create new data integration specifications.

This change will also influence the data processing activities, the API's and the data maps. In order to avoid modifying the algorithm, the integration process could take place in the data processing activities that are linked to the API's. By doing so, the impact of change is reduced to only the data processing activities.

Another example is extracting specific measurements of a patient to create the target table (for example measurements between january 2018 and september 2018). However, you do not need all the measurements that are registered in HIX.

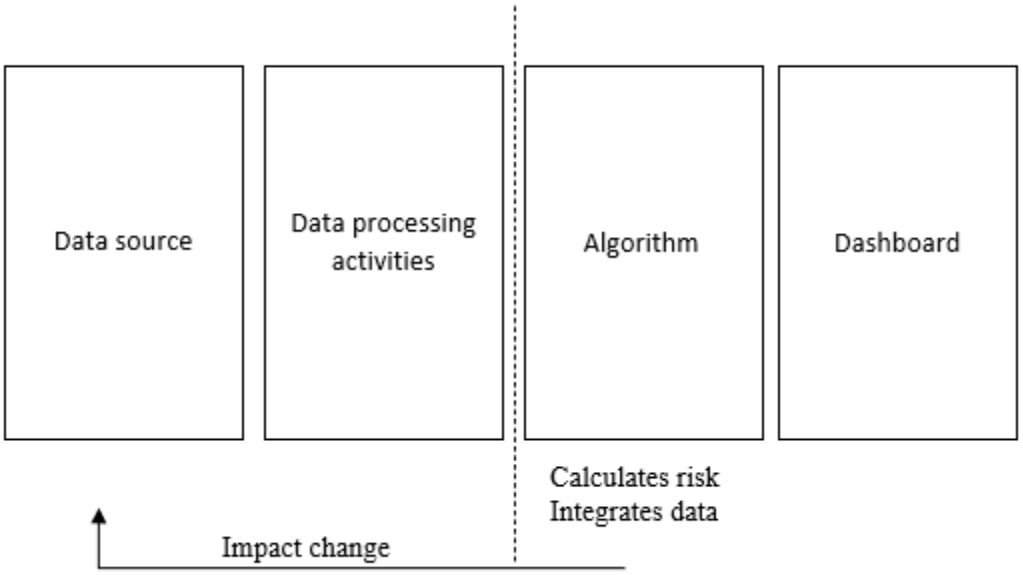

Figure 4.E Predicts main components

The data experts could program their algorithm in such a way that it receives all the measurements, but knows how to filter out the measurements that they actually need for the calculation. Another approach could be to shift this responsibility to the extraction phase and specify in the API's which measurements need to be extracted. These are design choices the data architect, and other data experts have to make in order to create an effective data flow.

Important to take into account during these design choices are the changes that could occur. Having a solid change management in place is vital for assigning these responsibilities. During

the interviews with the data experts, two elements were identified that could affect the dependency of the four elements when changes occur:

- Features
- Medical knowledge

We have explained how features work in section 4.2.1. The features decide the variables that are needed to calculate the risks. During the algorithm developement, the features are created by the data scientist. These features will have to remain in the algorithm because we are not aware if the extracted variables for the target table influence the predefined features in the algorithm. We have also explained the need of medical knowledge during the algorithm development phase. The question arises if the variables that need to be extracted for the target table need medical knowledge for interpretation. Unfortunately, we do not have enough information about the algorithm nor the variables that need to be extracted to assign these responsibilities, but we do want to mention this part of the data project, because it is important to think about the four elements in figure 4.E and what tasks you assign to these elements keeping future changes into account. What we do know is that the data integration and cleaning needs to be done before the data reaches the algorithm in the data process, and our architecture is focussed on this principle. Therefore, the current situation looks as follows:

- Data processing activities are responsible for data cleaning and data integration
- Algorithm is responsible for including features and calculating the risk factor

RQ6: Push/Pull data

Our approach to explore this requirement was dependent on numbers from the hospital that would give us an idea of how many women enter the hospital per day and per week, with pregnancy related complaints. Moreover, we also needed the clinicians view on how many dashboard updates they expect and need, to aid in their decision making process. Based on this data, we could indicate if the data had to be pushed or pulled. We discussed the difference between push and pull data briefly in the previous subsection. Both methods have their advantages and disadvantages. One of the advantages of push data is that the user does not have to worry about when and where to look for data. Another advantage is preventing the query performance getting overwhelmed with multiple requests. However, the disadvantage is that the user can receive data that does not correlate to the questions they asked the system, because they do not have control over the requested query. Unfortunately, we did not have access to the data we needed, hence we will discuss both methods.

The root of the problem lies in how many updates are necessary from a clinicians perspective. For example, there are eight pregnant women admitted at the ward, and the dashboard specifies information about them and their baby in the womb. It is important to know if their condition could change (during their stay at the ward) in such a way that it would impact the decision making regarding the NICU capacity. If the answer is yes, updating at a frequent rate would be mandatory. Secondly, the frequency of updates also depends on the clinicians requirements. If the clinician does not have any specific requirements regarding the update frequency, it has an impact on how many times per day the algorithm has to calculate risks.

Push data

Pushing data is receiving the information without involving the users request in the process. For example, the application could extract the patients data every 10 minutes resulting in a update every 10 minutes for the clinician [87] [89].

Pull data

When the data is pulled, Predict is using a trigger. The trigger in this case is a pregnant woman entering the hospital with pregnancy related complaints. The amount of pregnant women that enter the hospital can give Predict a better overview of how many times the algorithm should run. It is possible that both methods suit Predict, but it heavily depends on the clinicians requirements regarding the updates they need [87] [89].

RQ7: Store the dashboard results for traceability

We have mentioned that the dashboard is considered to be used as a medical device in the healthcare sector. A dashboard is a dynamic application that changes its interface when it is updated. However, the results of every update have to be stored, so the decisions made by the clinicians based on the dashboard results can be traced back to the origin. These results need to be stored in an operational datawarehouse.

RQ8: Enhance and maintain privacy & security

Maintaining the privacy of the patients, and the security of the data is a standard in almost every data architecture and should be mandatory in every data project. You need to protect the data against unauthorized users, especially when it comes to healthcare data which contains sensitive information about patients. Moreover, the protection of the data needs to be executed during the whole data lifecycle, from generating data to transforming it into information and using it in practice. We have provided an example of data protection in section 4.2.1 where we presented an overview of the distinction between the tasks of the data manager and the data scientist.

Privacy in healthcare can be defined as "*The right and desire of any patient to control or regulate the disclosure of personal health information*" [90]. Security is related to the physical and technological measures that need to be executed to protect the data. In practice, there are several ISO standards accross the world that define data confidentiality, authorization & authentication. Organizations need to define security goals that fit their business and adhere to these ISO standards [90]. Defining these goals are important these days because more hospitals use technology in their daily practice. In the Netherlands, every hospital uses an EHR, but there are also several treatments that involve sensor networks to monitor patients. Therefore, having a solid security plan in place, monitoring the security and updating the security tasks is important to maintain the patients privacy and the data security in the hospital [90].

Literature proposes several solutions to manage the security challenges [90]:

- Incorporate a role based access control where specific people are authorized to access resources. For example, when accessing an Electronic Patient Record, the clinicians and other medical staffmembers are allowed to read and edit but insurance providers are only allowed to read.

- The use of cryptographic encryption techniques. These techniques transform a plain message into an alternative form, and only the authorized people can decipher the alternative form to its origin.
- Use authentication algorithms like a digital signature to access information

4.3 Tasks Real Time Process

We derived several tasks based on the defined requirements in the previous section. These tasks form the foundation of our architectural layers that we discuss in our next chapter. Table 3 presents an overview of the tasks.

Requirement	Tasks
RQ1 Select variables that need to be extracted	Task 1: Define input for the algorithm
RQ2 Extract the right data form real time database	Task 2: Access a real time database Task 3: Data mapping
RQ3 Extracted data needs to be integrated	Task 4: Define integration rules
RQ4 Extracted data needs to be clean	Task 5: Perform quality analysis Task 6: Define integrity rules
RQ5 Algorithm and application need to be independent	Task 7: Algorithm (including features) has to calculate risks Task 8: Develop data change management plan
RQ6 Push/Pull data	Task 9: Data needs to be pushed/pulled
RQ7 Store dashboard results	Task 10: Database needs to store the dashboard results
RQ8 Enhance and mantain privacy & security	Task 11: Maintain data security

Table 3. Tasks real time process

Chapter 5

Proposed Data Architecture

5.1 Data Architecture

This section will describe the use of data architectures in general, and why they are important in an organization that wants to expand their data environment.

Literature and articles specify several definitions of what a data architecture implies:

- *"A data architecture is a set of rules, policies, standards and models that govern and define the type of data collected and how it is used, stored, managed and integrated within an organization and its database systems"* [91].
- *"A data architecture describes the structure of an organizations logical and physical data assets and data management resources"* [92].
- *"A data architecture is the process of planning the collection of data, including the definition of the information to be collected, the standards and norms that will be used for its structuring and the tools used in the extraction, storage and processing of such data"* [93].

The commonalities between these definitions is the management of data. Every definition is focussed on managing the data during every activity in the data process. Figure 5.A presents the elements that need to be managed.

Figure 5.A Data process activities

Almost every healthcare organization has data driven applications included in their business process, and this phenomenon has led to managing data processing activities full time. Additionally, the data experts and IT department have to collaborate to answer relevant questions that are related to maintaining the integrity and reliability of the data, and how to store, treat and manipulate data to provide users with the right answers. Organizations need an overview of how to manage the data processing activities. This overview could also function as a foundation to expand data driven applications and help organizations adapt to emerging technologies and environmental changes [93] [94].

The activities presented in figure 5.A can be managed in several ways:

- Data experts could create an overview of all the datasources including the data models to present their datastructure. This will help the extraction, cleaning and integration activities in the data process.
- Data experts could define the software used to execute all the tasks. This will provide the IT department and data experts with an overview of the technology behind the data flows and how it should collaborate to create an efficient data process.

- Data experts could define standards to transform the data. Standards could be defined based on structuring the data, the meaning of data or how to clean the data. These standards will benefit the cleaning and integration process. It can also help in adding new data projects as the standards can be reused.

The approach of managing the activities depends on the business goals. For example, your business goals could be related to applying more real time applications in practice. This decision will affect the software your company will use to process the data. Therefore, it is important that the architecture is alligned with the business goals. On the other hand, the architecture should not be static but flexible and easy to adapt and modify according to the changing scope of a project or the growth of an organization [93] [94]. For example, in Predicts case, the data experts need to take into account that the data experts want to include more data sources over time to optimize the algorithm.

The main objective of an architecture is creating a centralized data environment where systems are integrated and data processing activities are connected. At the moment, some companies cope with IT departments that function in isolation, and have their own data standards and architecture. Many applications are build based on requirements that do not allign with the architectural standards of an organization. This unfortunately has led to disjointed systems with several consequences like difficulties in troubleshooting production data issues, or problems assessing the impact of a change. A data architecture could provide clarity about several aspects of the data, which enables data experts and the IT department to work with proper data and solve challenging business problems [93] [94].

Our data architecture is more like a data blueprint on a conceptual level that provides an overview of the relevant elements of data processing, and also specifies the tasks that need to be executed to provide an efficient dataflow. The building blocks of our architecture are defined based on our literature review provided in chapter 3, and the requirements in the previous chapter. Our architecture is focussed on managing the data in every aspect of the data flow (from source to application) and connect the project requirements to the data processing tasks. Eventually, the architecture should lead to a better understanding on how to integrate systems and support an efficient dataflow for every data expert related to Predict. Furthermore, we wanted to keep the flexible data architecture principle in mind. Therefore, our data architecture is build in such a way that it can be easily extended and adapted to include more elements like software specifications and data standard principles.

5.2 Proposed Architecture Predict

Figure 5.C presents the proposed architecture for the Predict project when the algorithm starts running in practice and the dashboard needs real time updates. We have chosen a layered architecture pattern to develop our architecture. A layered architecture is a form of a software architectural pattern and is defined "*as n-tiered patterns where the components are organized in horizontal layers*" [95]. This architectural form is developed to identify multiple components in software development that need to collaborate. The main objective of this architectural form is to break down the software in tiers that include their own functionality to support reuse of these functionalites in the future. The strength of a layered architecture in our model is the seperation of roles in different layers. Every layer has their own responsibility in contributing to the optimization of the data flow, and anticipate on future change [95].

Our layers are stacked horizontally on top of each other where some layers coexist next to each other. The sequence of layering our components does not correlate to the importance of the layers because we do not provide a dataflow in the model. In other words, we do not identify a specific order in which the tasks have to be executed to transform the raw data into information. The architecture breaks down the tasks that need to be executed to build an efficient dataflow. Additionally, these layers can be used to build a data processing system which we will present in section 5.3. Every layer is responsible for one or more elements in the data processing cycle, while helping the other layers to execute their tasks. We have chosen a layered architecture mainly because of the clear overview it provides. The elements of the data process are visible and indicate their dependency on each other. A presentation of the data processing elements including tasks, aid experts to adapt, modify or extend the architecture with other applications or more datasources.

We will use a syntax related to package diagrams to conceptualize our data architecture. A package diagram is one of the structural diagrams of Unified Modelling Lanuage (*next: UML*). The diagram provides an overview of the arrangement of a system. It also presents dependencies between sub-systems by defining layers, which could be compared to the layered architecture form. Package diagrams are predominantly used to create more structure between high level system elements, and they organize a system by defining diagrams, models and documents per package. The syntax of the diagram mainly consists of packages and arrows that show dependencies. We will use the packages to define our layers and present the dependencies between them as indicated in figure 5.B.

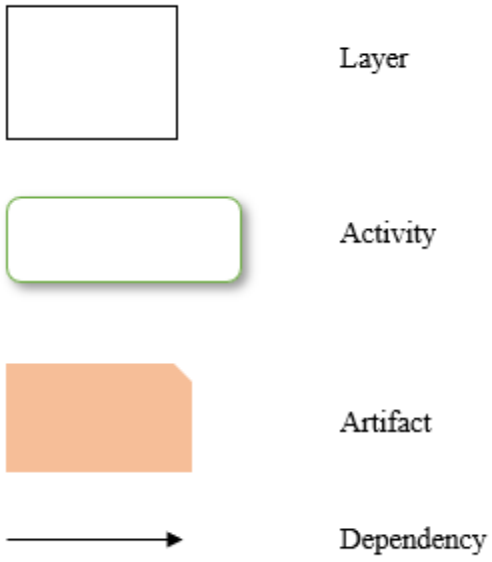

Figure 5.B Syntax data architecture

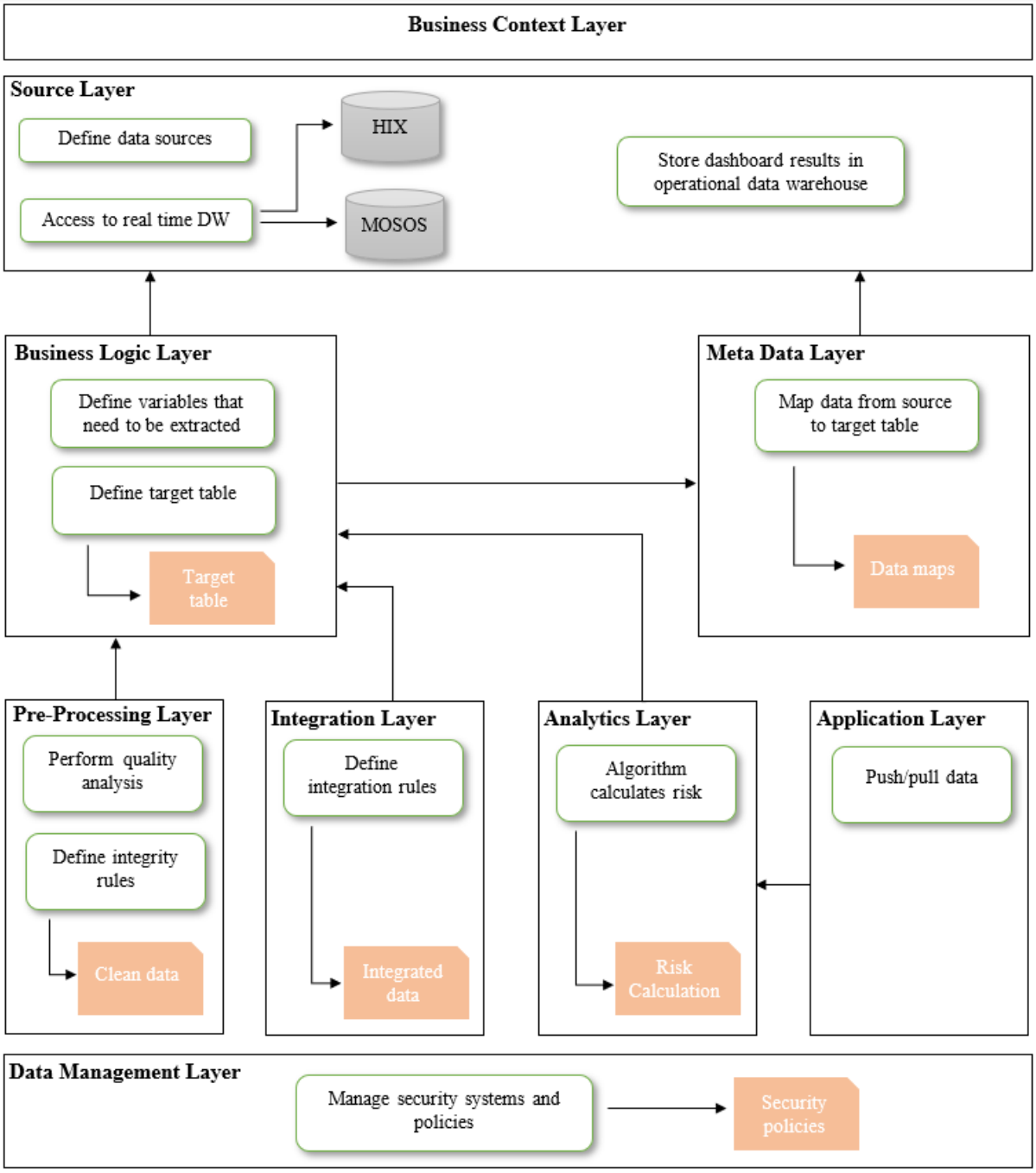

Figure 5.C Data architecture Predict

Figure 5.D provides a practical example of how the architecture can be used in the real world. The source layer contains the real time data sources. The variables that need to be extracted reside in different tables in HIX and MOSOS. The business logic layer contains the target table that the algorithm needs to calculate the risk. Therefore, it needs the meta data layer and the source layer to form the target table. Once the data sources are accessed, the data needs to be cleaned and integrated. The meta data layer combined with the pre-processing layer and integration layer are needed to execute the transformation rules. In practice, the data process can use queries to specify the cleaning and integration rules. Next, the target table is created and provides input for the algorithm. Finally, the result of the algorithm is presented on the dashboard.

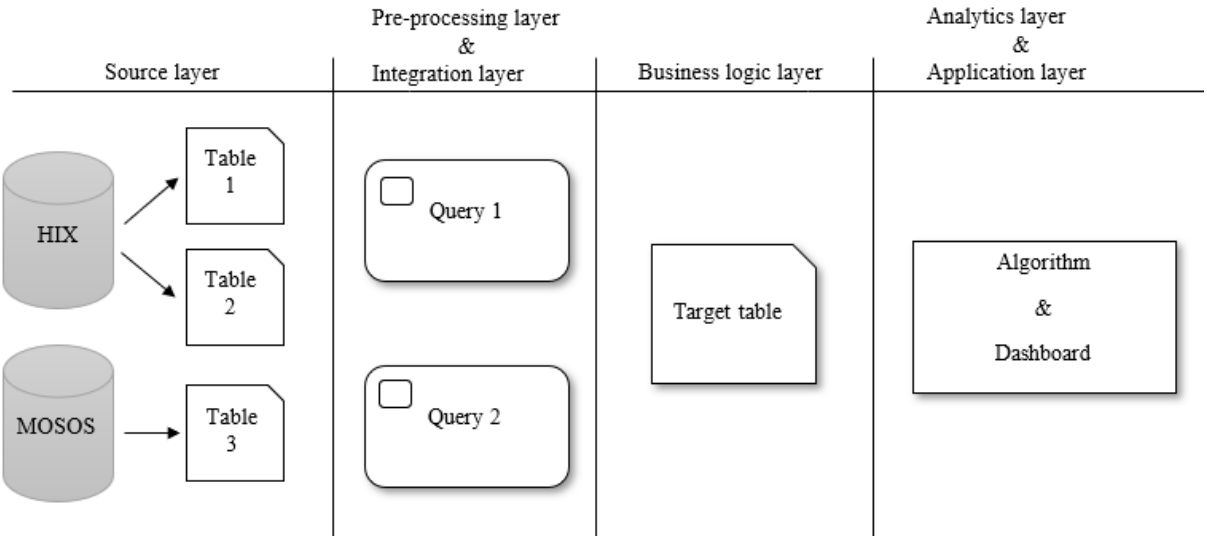

Figure 5.D Architecture in practice

Next, the layers will be discussed. Every layer contains an explanation of the tasks. Some layers also include a figure that describes the tasks that need to be executed by the teammembers of Predict or the IT department, to give the layer more depth.

Business Context Layer

This layer forms an integral part of the architecture and helps to build a solid foundation that functions as a guideline during the development of the solution. This layer supports brainstorming about the problem context, and the case study upfront to eventually progress to a mutual approved solution. In many agile software methodologies, the initial phases are focussed on project planning and elicitating requirements. Even in the data mining methodologies that we discussed in chapter 3, the main focus in the first couple of phases is planning, understanding the problem context and defining goals. In this phase the experts should ask questions such as:

- What are we building?
- Who are we building it for?
- Why are we building?

These questions are extremely important and the execution of these questions are critical when building a solution. In Predicts case, there are many more stakeholders involved in the project that need to be taken into account during the project. For example, the clinicians that eventually use the application, but also the data experts that need to develop the algorithm and the IT experts that need to make sure the algorithm and the data process surrounding the model keeps running. These stakeholders all have their own problem statements that are related to each other. Creating an overview of these stakeholders and their main objective will improve the collaboration between the experts and provides more clarity. Figure 5.E presents a model to structure this phase.

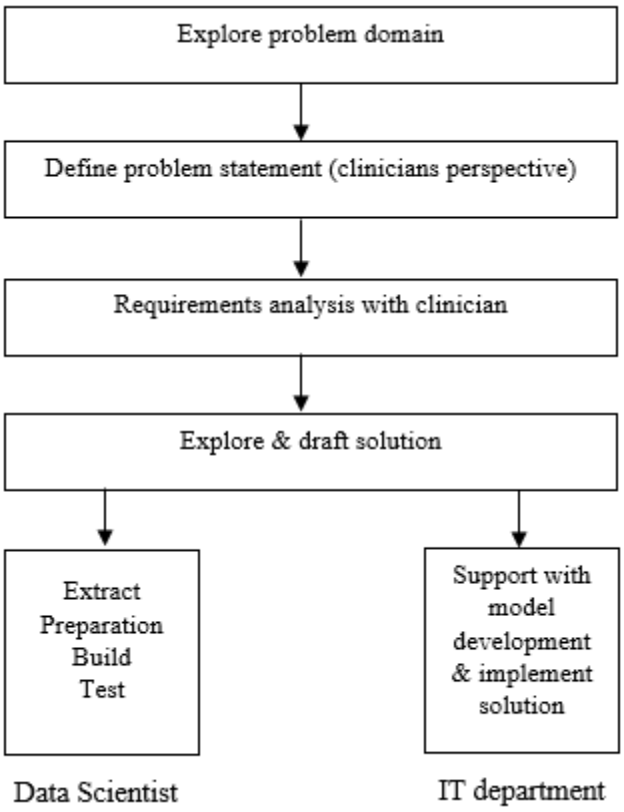

Figure 5.E Business context model

You start off by exploring the problem domain and defining the problem statement from the clinicians perspective. This problem statement is related to the capacity challenges at the NICU. The clinicians have their own perspective regarding this problem and the solution. Next, the data experts perform a requirements analysis with the clinicians which is mainly focussed on two elements:

- The risk calculation to develop the algorithm
- The application that the clinicians will use

During this phase it becomes clear which datasources the experts need, and the variables that need to be extracted. Moreover, they have information about the data visualization which relates to the variables that need to be extracted. You first need to know what you are going to present, and after that you know which variables you need to present the right information. The data experts explore and draft a solution that is related to the dashboard, and they also research which algorithm to use (for example, neural networks, random forest). The data scientist and managers develop the algorithm, while the IT experts such as the data architect and data warehouse experts support the data scientist from a data processing perspective. They have extensive knowledge about the data processes in the hospital, the applications the hospital uses to extract data and the programming languages that are used to create queries. Even the several design choices that need to be made, like we mentioned in requirement 5, should be discussed with both the data scientist and data architect. The main objective of this layer is to improve the collaboration between the different teammembers of the project and to make sure everyone is one the same page and up to date. Moreover, the tasks described in figure 5.E are agile, they can be performed simultaneously, and its important to revisit them time and again to update the information while the project keeps progressing.

Business Logic Layer

The main objective of the business logic layer is to define the output of the processing activities that result into the target table. This layer has one task:

Task 1: Define input for algorithm

The data scientist is in charge of creating the algorithm, hence they have the knowledge about how the algorithm works and what it needs to calculate the risks. Moreover, they are the ones that need to define the final output, and the raw values that need to be transformed to create the desired output. In order to define the output, the data experts need the specifics of the algorithm and the problem context. Moreover, this layer uses the business rules defined in the pre-processing layer and integration layer that define the format and values of the target table. Figure 5.F presents a figure that specifies the tasks to structure this layer.

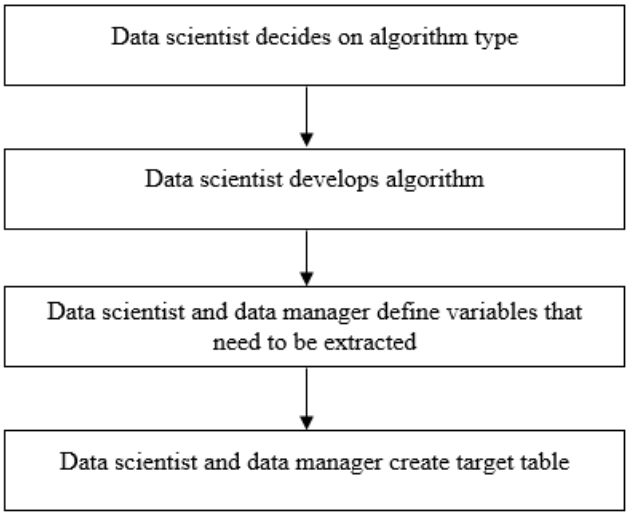

Figure 5.F Business logic model

Source Layer

The main objective of the source layer is to manage the datasources and store the dashboard updates.

The source layer has two tasks:

Task 2: Access to real time database

The application (dashboard) needs to get access to the API's of the real time data base of HIX and MOSOS to extract the right data.

The main objective of the source layer is to manage the sources that are needed to extract data from, and ensure that the API's deliver the data. These sources could differ from the sources needed in the algorithm development phase.

Task 10: Database needs to store dashboard results

A dashboard is considered as a medical device in the healthcare sector. A dashboard is a dynamic application that changes its interface when it is updated. However, the result of every update has to be stored in order to trace the decisions that are made by the clinicians based on the dashboard results. These results need to be stored in an operational data warehouse.

Pre-processing Layer

The main objective of this layer is to improve the quality of the data that is extracted for creating the target table. Two tasks are identified in this layer:

Task 5: Perform quality analysis

First, the data experts needs to execute a data quality analysis. We have established some data quality dimensions in chapter 3. Literature presents several techniques to execute an effective analysis, and they have also specified methodologies that are focussed on improving data quality [96]. We have identified several elements from these methodologies, and created an abstract model for the data experts to perform a quality analysis on the data presented in figure 5.G.

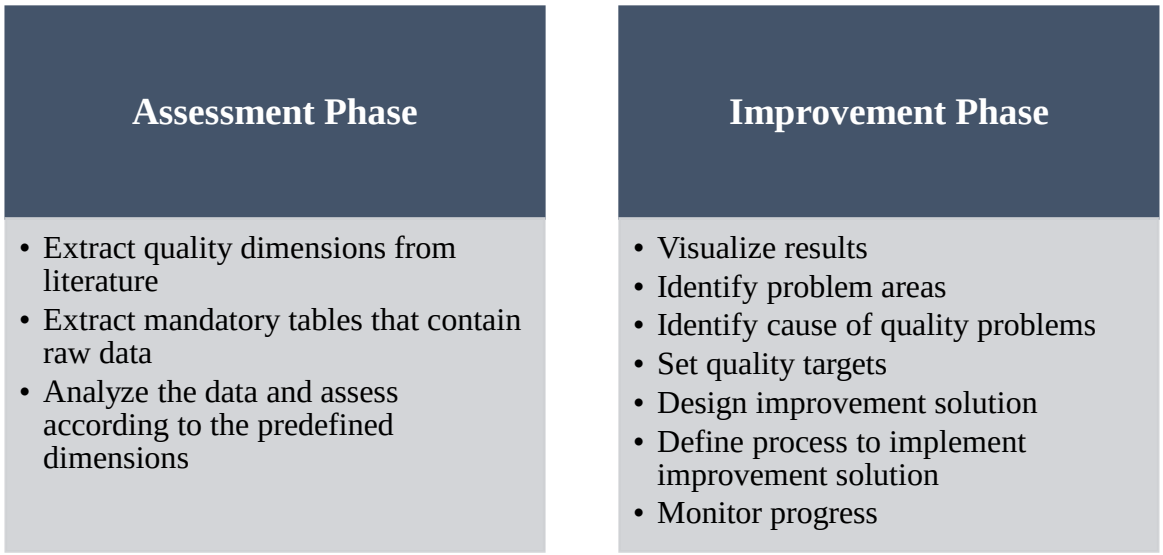

Figure 5.G Model quality analysis [96]

Figure 5.G starts off with the assessment phase where the data experts analyze the quality of the raw data according to the predefined dimensions. The main objective of the improvement phase is to identify the areas that need improvement and define improvement solutions. For Predict, the quality analysis for extracting real time data can be performed on the variables that need to be extracted to create the target model. In the future, when the algorithm is modified or a new algorithm is created, all the variables that are included in the algorithm can be taken into account. Additionally, figure 5.G includes designing an improvement solution. One of these solutions can be setting up business rules to improve the quality. An example of these business rules are integrity rules.

Task 6: Define integrity rules

After performing the quality analysis, data experts will get a clear overview of the problem areas that needs improvement. Like we mentioned before, the quality challenges are partially caused by the front end, hence not all quality problems can be solved at the back end. However, there are methods that can be implemented in the real time process to manage the majority of the quality challenges.

Data experts can specify rules (integrity rules) that are related to the quality of the data. The data should adhere to these rules in order to be processed. For example, one of the data experts found a medically incorrect blood pressure value in the records. To avoid these values to be processed, the data expert could restrict the range of the extracted values by specifiying that blood pressure values can only be between x and y, or the values always have to be < x. Data that does not adhere to these rules are automatically identified as incorrect and will not be processed [24].

Integration Layer

The main objective of this layer is to make sure the data is efficiently integrated in a real time process. This layer has one task:

Task 4: Define integration rules

To execute this task the data experts needs a thorough understanding of the structure of the data sources and the data. The integration rules can be specified in the data maps. The data managers are a good option to define the integration rules because of the amount of knowledge they possess. Moreover, in the preparation phase of the algorithm the data experts created new tables to integrate the data, in the real time phase they can work with queries that define the structure of the data. Figure 5.H specifies the tasks to structure the integration layer.

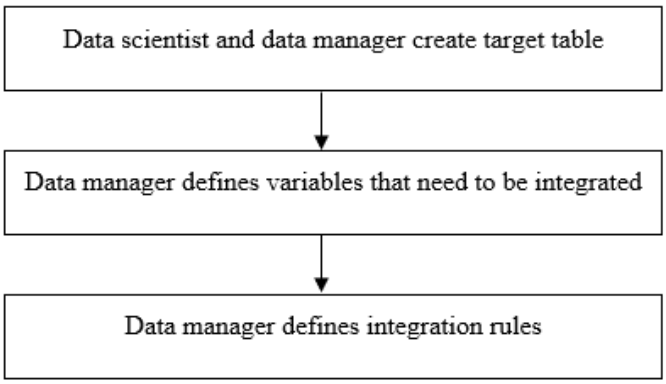

Figure 5.H Integration model

Analytics Layer

The main objective of this layer is to manage the algorithm that is created to calculate the risk. This layer contains the algorithm created by the data experts. It has one task that needs to be executed by the data experts in the Predict team.

Task 7: Algorithm (including features) needs to calculate risk

The main task of the algorithm is to calculate the risk when it receives the target table. The dashboard provides an overview of all the patients at the pregnancy ward. The clinicians can click on every patient to receive an overview and detailed information about the pregnancy per patient, and the risk calculation of the baby in the womb. To create the algorithm, the process as describes in figure 5.E can be used.

Application Layer

The main objective of this layer is to manage the application that is used by the clinicians to help them in their decision making process. The application layer is responsible for developing the dashboard and its functionalities. The application layer has one task.

Task 9: Data needs to be pushed or pulled

According to the requirements of the clinicians regarding how many dashboard updates are needed, and the data about the amount of pregnant women that enter the hospital on a daily basis, Predict can use the push or pull method to extract data.

Another element in this layer is the definition of data latency. Latency is defined as the time required to answer a single query. The data experts need to discuss and monitor how long it takes to get the consolidated data from the datasources [97]. Figure 5.I specifies the tasks to structure the application layer.

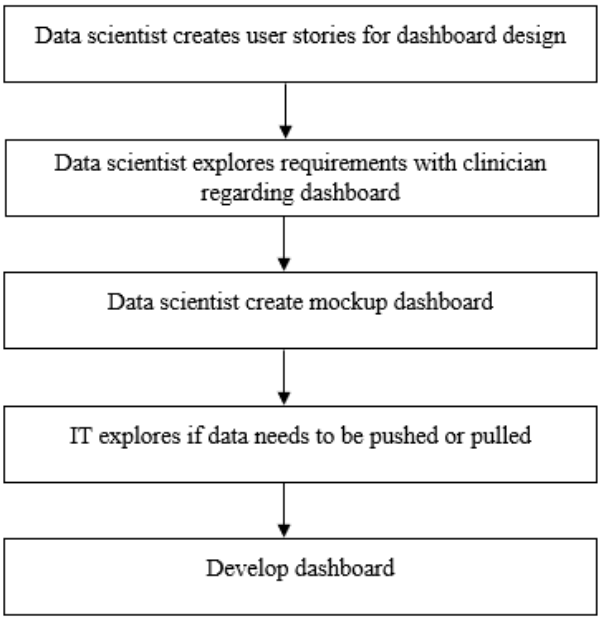

Figure 5.I Application model

Meta Data Layer

Meta data is defined as "*data about data*" [98] and functions as an important aspect in the data processing activities. Meta data provides information about certain characteristics of the data which results in more efficient data processing, and improving the interpretation of data. Meta data is an extensive term and embodies several categories like:

- Technical metadata
- Administrative metadata
- Descriptive metadata

We will not dive into the specifics of the use of meta data in this research, because our conceptual architecture views the data process in a more abstract manner. In general, meta data can be used to improve several aspects of data processing. In the context of our project, we will use the layer to define data maps. The layer consists of one task:

Task 3: Data Mapping

The data needs to be mapped to the data source to support an efficient data flow between the source data and the target data. From all the data experts, the data manager would be the most appropriate choice for this task. The data managers are the ones that extract and integrate the data in the first phase of the algorithm development. They possess accurate knowledge about the structure of the data sources, and the data itself.

The main objective of this layer is to create and store the data mapping definitions of how raw data is transformed to the target table. This information is essential for the extraction process, but also for future changes. For example, when the variables are registered in a different record in the same table, the mapping specification needs to be modified.

Data management layer

Task 8: Develop data change management plan

The main objective of the data management layer is to focus and maintain tasks that do not directly relate to the data processing part. Examples are change management and security. We have mentioned in requirement 5 (algorithm and application need to be independent) why change management is an important part of a data project. Most organizations practice change management when it comes to implementing new software. However, there is not a clear plan regarding change management for data. There are several examples of changes in data that could influence a data process. For example, changes in data models or adding new sources.

Developing a change management plan should involve all the data experts (data scientist and data managers), the data architect and the IT department. These experts have accurate knowledge about several aspects of the data project, to understand how to cope with changes. A well written change management plan consists of an analysis of the impact of possible changes, and how to manage them in the future [99].

Task 11: Maintain data security

UMC is a big hospital and they have been working with data projects for a while, demanding them to have their security systems in place. However, new projects require revisiting the security models and improve or modify them where needed. We have already mentioned how data projects include different phases. Each phase (from extraction to providing information) should be managed by the security systems and policies. For example, when the dashboard is developed by an external party and testing needs to happen, the hospital has to think about testing the application within the privacy & security rules that are predefined for outsiders. Figure 5.J specifies the tasks to structure the data management layer.

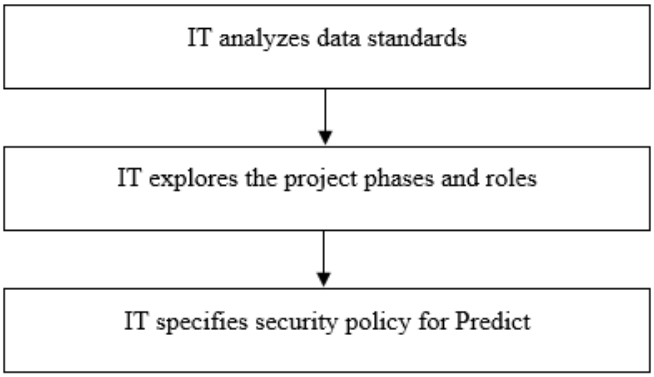

Figure 5.J Data management layer

5.3 Virtualization or Near real time ETL

Section 5.2 presented the architecture that specified the tasks that need to be executed in order to provide the algorithm with consolidated data. In chapter 3, we discussed how data processing has evolved, resulting in ETL developing to several forms to adapt to the changing business needs that require real time data. In ETL and near real time ETL the data is copied from source systems, and constantly migrated while performing transformations. In data virtualization, the data is not migrated but the users request is queried in real time using views that return the consolidated data to an application. We have already presented the architecture of near real time ETL in figure 3.C. Figure 5.K presents an architecture of using data virtualization in practice. The figure does not present a general data virtualization architecture, but is defined by a company that we anonymized due to privacy reasons. This company uses a data virtualization tool called Tibco to manage their real time projects. In their architecture, the interpretation component is managed by Tibco. The company uses batch data and stream data to manage their projects. The stream data is managed by Tibco [100].

Dataflow virtualization method

When Predict decides to use data virtualization to manage the data processing activities, they will need a general architecture that specifies how to manage the raw data and transform it into the target table. Predict could make their own virtualization tool and connect this to their real time data warehouse, or use existing tools on the market like Tibco or Denodo [102] [103].

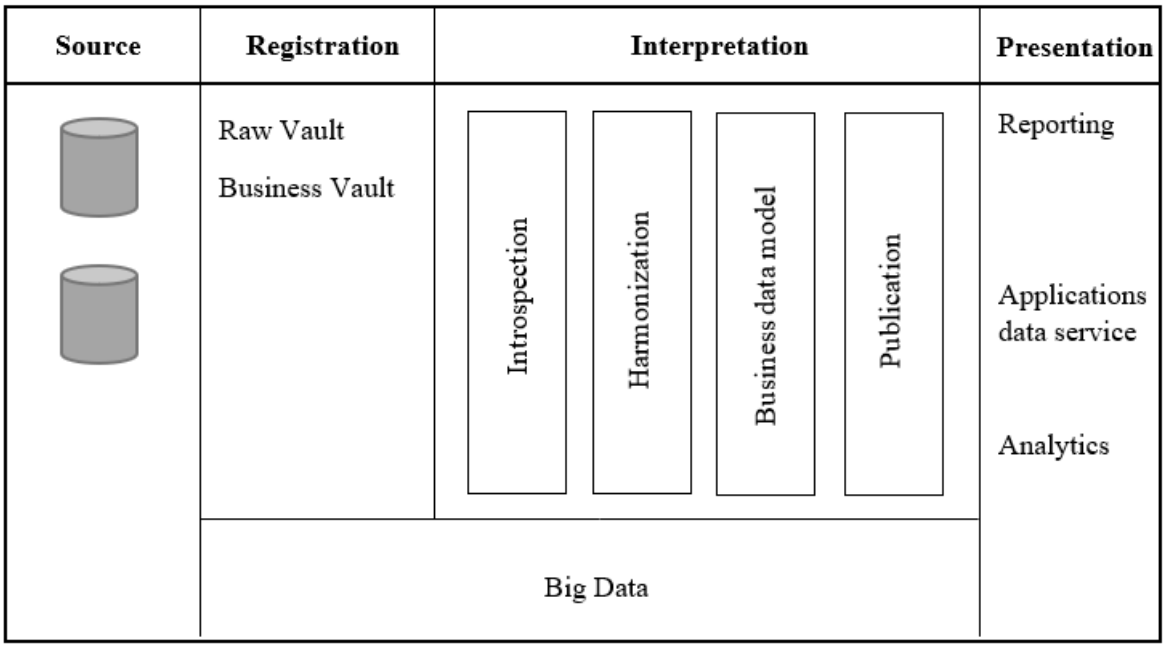

Figure 5.K Architecture data virtualization (Anonymous company) [101]

We will use Tibco as an example to process the data for Predict. Tibco is an American company and provides software that enables organizations to optimize the potential of their real time data to improve the decision making. They do so by connecting intelligence platforms to applications and data sources, to provide companies with a virtualized unified datasource to enhance data management. One of the products they offer is the Tibco data virtualization tool [102]. The tool includes a layered approach, and provides the data expert with the freedom to add as many layers as they want to manage the data processing activities. This approach is included in the architecture we present in figure 5.K. They have chosen four layers to organize their virtualization tool. The first layer cleans the data, the second layer manages the data integration, the third layer is responsible for creating the target tables, and the fourth layer manages the real time applications. Our data architecture presented in figure 5.C specified the mandatory tasks to provide input for the algorithm. The following layers of our architecture are directly related to the data processing part:

- Pre-processing layer
- Business logic layer
- Integration layer
- Meta data layer

In order to structure the data processing area of the Tibco tool, we have created figure 5.L that presents an abstract architecture of a possible solution using data virtualization by Predict. According to the data experts and project requirements, layers can be added or removed. Our architecture will help the data expert with defining the layers.

The data is accessed in a real time data warehouse, and processed in the layers of the data virtualization tool. The first layer is in charge of cleaning the data, the second layer integrates the data and the final layer creates the target table. The layers are accessed by views that are created according to the transformation rules. The mapping layer provides an overview of how the data is transformed between source and target table.

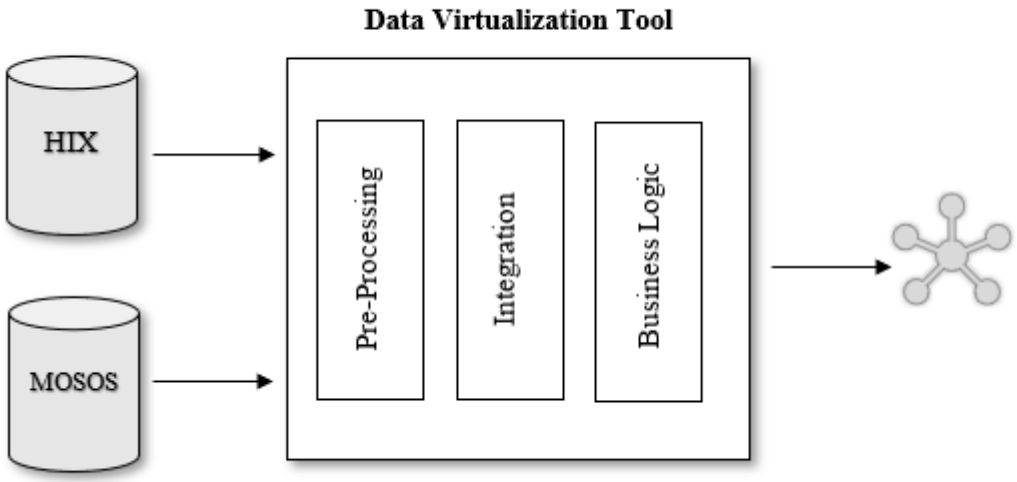

Figure 5.L Data flow using data virtualization [7] [8] [9] [13]

Dataflow near real time ETL method

In near real time ETL the data is stored in the data warehouse. An advantage of this approach is that the hospital does not have to purchase a new application, and the data architect and IT experts can mainly use their existing tools to create this dataflow. In the abovementioned explanation of the virtualization dataflow, we presented the four layers that contain components from our architecture that need to be processed in real time. Figure 5.M provides an overview of a general architecture using near real time ETL. We are inspired by the near real time architecture that is presented in chapter 3.

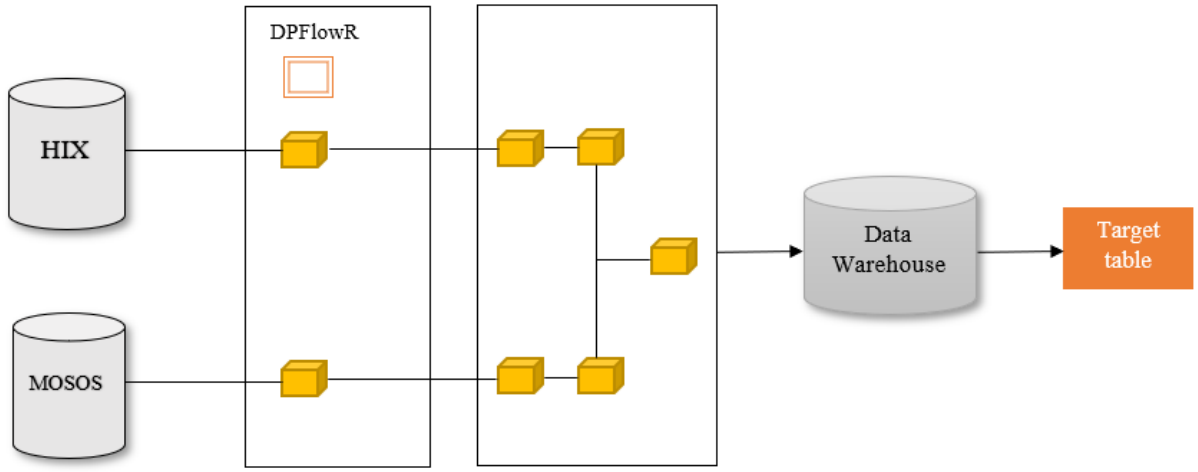

Figure 5.M Data flow using near real time ETL

The architecture includes the real time data warehouses where raw data is stored and needs to be processed. The yellow squares respresent the data that is transformed. The source flow regulator manages the values that need to be extracted of every patient at the pregnancy ward, and propagates these values to the data warehouse. This could happen periodic or by using a trigger. The data processing area cleans an integrates the data. Finally, the data is loaded into a database residing in a datawarehouse. The data model of the data warehouse is the target model that defines the input for the algorithm.

In summary, when Predict is expecting to add new sources that contain unstructured data or they expect the existing sources to change, it could be more useful to use a data virtualization tool. However, acquiring new tools and maintaining them will cost time, money and effort. The key to creating a succesful data processing environment is to centralize and standardize the dataflows as much as possible. In order to achieve this, it is important to analyze all the real time data flows and identify their characteristics, and how they are executed at the moment. Based on this information, they should conduct a research if data virtualization would help improve the real time applications or maybe enhance their existing technology and warehouses to develop efficient data flows. The hospital already has an information infrastructure that defines data processes, therefore the experts need to take into account the existing infrastructure and needs of the hospital [103].

Chapter 6

Model Evaluation

As stated in chapter 1, we aimed at designing a data architecture that would help Predict create an efficient real time data flow. In this chapter, we will evaluate the effectiveness of the elements of our data architecture as presented in chapter 5 by using an expert opinion as a validation method. Initially, we wanted to use a focus group and invite several data experts from ADAM to evaluate our architecture.

A focus group would fit the nature of the project as we are also aiming at bringing IT and data experts together to think about the algorithm in the production phase. We wanted to start a discussion about the need of a unified data environment. Moreover, we wanted to explore the several tasks that we describe, and who should be in charge of executing them to improve the collaboration between data experts and IT professionals. And more importantly, we wanted to explore what elements are missing to create the right dataflow. Unfortunately, because of the current social circumstances in our country we had to change our validation method. We will now use a data expert that evaluates our architecture to improve the model. Our data expert has experience with algorithms and managing data in a data process. The validation was executed using interview questions. Appendix B presents the interview protocol for the evaluation of the model.

6.1 Goal validation

The main objective of the validation phase is to present the strengths and weaknesses of our architecture, and explore whether the architecture could help Predict in building a real time data process.

The first phase of our interview is focussed on the strengths and weaknesses of our architecture, and the second phase provides room for general feedback. The most important topics of our interviews are listed as follows:

- Strength/weakness architecture
- Collaboration data experts and IT
- Main objective Predict
- General feedback

6.2 Expert opinion

Strength/weakness architecture

According to our data expert, the architecture features a clear breakdown, specifically on an operational level where each activity and layer can be bound to different data engineering/science sub specialities. These are common in medium to large organizations such as (academic) hospitals.

The downside of this architecture is from a visual perspective, The data flow by itself is not at the forefront of this design, and thus difficult to follow [108].

Collaboration data experts and IT

Our data expert is positive about our architecture improving the collaboration between data scientists, data managers and IT professionals at the hospital, but it depends on the specific problem at hand. In the context of Predict, it makes sense to involve data scientists and data managers that developed the algorithm to help model the target table, and specify transformation rules for the production phase [108].

Main objective Predict

The main objective of Predict is to have an algorithm that calculates risks with the most current data sets of the patients at the maternity ward. According to our data expert, the proposed design can clearly support this end goal of reporting on NICU capacity including the predictive element. Our data expert also believes that our architecture can help the Predict members build a real time data flow only when real time data bases are available.

We also asked our data expert if he thinks that this architecture could be applicable for other projects in the hospital to create data flows. According to him, the current design reflects a level of abstraction where it can be generally applied to other problem domains [108].

General feedback

When we asked if our data expert could change anything about the architecture, he answered that he would center the design around concepts such as data flow, and the concrete data providers and data consumers. However, he does feel that the downside of this approach is that the solution would be less generic and perhaps from an academic perspective, less interesting [108].

In summary, our data expert is mainly positive about what we wanted to achieve with the architecture. He does have some remarks about the design of the architecture as we are not presenting a data flow making it visually a bit confusing. During our design phase, we have thought about including a data flow. We eventually chose not to present a data flow because we wanted to keep the architecture as abstract as possible. We were focussed on presenting the tasks that need to be executed to create an efficient data flow. These tasks can be executed even during the algorithm development phase. Once the transformation rules and prefered software to manage the data flows are in place, the data flow can be build. Lastly, we are happy that our expert realised that the architecture has achieved the right level of abstraction because we want the architecture to be a tool for future data projects of the hospital. Therefore, we have decided to not change our data architecture.

Chapter 7

Conclusion

In this chapter we will present our conclusion to the main research question:

How can we develop a data processing architecture supporting prediction algorithms for NICU bed capacity management?

First, we will discuss the subquestions that we formulated to structure our research process, and we will close this chapter with an answer to the main research question.

7.1 Conclusion of sub-questions

Question one

How is the NICU capacity managed at the moment, and how does the capacity management impact the patient flow?

Babies that are born preterm of suffer from medical complications are admitted to the NICU. The NICU departments in the Netherlands are coping with a capacity problem at the moment. The capacity is related to the amount of nurses that are active at the department. Often, when the NICU babies are born, they have to be transfered to other hospitals because of the shortage of staff members. This unfortunately, has become a national problem and in some cases babies have to be transfered to Belgium to manage the increasing number of NICU babies. The hospitals do not share their capacity data with each other. Therefore, when a baby has to be transfered, clinicians spend a lot of time searching for other hospitals. Moreover, the beds at the NICU are not utilized properly. Mainly because of the complexity of the decision making that lies in estimation problems about the moment of delivery. The NICU department has rules defined to help them improve the decision making, and these rules predominantly revolve around analyzing the medical situation of the patient, but the capacity problems still remain a big challenge.

Question two

What is the current understanding that domain experts have about processing/managing data in a healthcare environment, and how has this evolved?

We have explored how data processing has evolved over time to meet the changing business requirements. We started off with the most prominent data processing method, ETL. The method is focussed on batch processing and plays an integral part of data warehousing. Because of the changing nature of the users requirements, that are related to processing more data at a more frequent rate, the data processing activities had to evolve as well. Other tools entered the market like data virtualization, that is focussed on virtualizing the data and transforming it in a real time flow without migrating data. Another change in the users requirement is applying machine learning methods to explore the data from several perspectives and in more detail. An example of this trend is the use of predictive algorithms. These algorithms eventually run on a data warehouse or need real time data. We wanted to dive deeper into the data processing activities and researched a knowledge discovery model, CRISP DM, to gain more knowledge

about algorithm development and the related activities. The activities are predominantly related to the cleaning and integration activities. During this chapter we have gained more understanding about how several characterisics of the data processing activities need to be managed.

a) How does data analytics influence the healthcare sector and what are the challenges?

Real time data processing in healthcare is a growing phenomenon. Healthcare has collected a lot of valuable patient data over the years, partially stored in the EHR's. With the use of real time applications and machine learning, the healthcare sector has entered an environment where clinicians could use the data in their research to different treatments. Moreoever, they can use real time applications to aid in their decision making process or regulate other activities in the hospital. We have mentioned some examples of these applications in section 3.5. However, there are some challenges related to the use of real time data analytics. One of the most important challenges is the lack of interoperability when it comes to healthcare systems, and the quality of the data that resides in these systems. Managing these aspects are important because they can affect the outcome of the data processing activities.

Question three

What are the requirements that we need to take into account while designing the data architecture?

We have executed extensive interviews with the data scientist, data managers and the data architect to gain more knowledge about the algorithm development and their perspective on the real time data flow. These interviews, and the knowledge that we have acquired in chapter 3 resulted in eight requirements. These requirements are focussed on the data processing activities that need to be executed during the dataflow, but are also related to activities that help develop the algorithm and manage the data processing environment in general. Related to the requirements, we have identified eleven tasks that need to be executed to create an efficient data flow. The requirements and their related tasks are listed in section 4.3.3.

Question four

What data architecture is needed to manage the data flows and organize the data in the information systems?

We first focussed on dissecting the definition of a data architecture and what elements we can apply to the kind of architecture we want to create. The commonality between the definitions was the management of the data. According to us, a data architecture presents how the characteristics of the several data processing activities are managed. There are different ways to present this. The architecture could include data models to present the data that is needed, or data standards that define how the data should be processed. We observed that Predict is still in the early phase of creating a real time process. That is why we wanted to conceptualize our architecture at an abstract level that defines the activities that need to be executed to create the new data process.

Question five

What are the advantages and disadvantages of the proposed data architecture?

Our data expert evaluated our data architecture and was predominantly positive about the result. According to the evaluation, the architecture can help Predict build a new data proces for the algorithm in production and might even be useful in similar problem domains. However, there were some remarks about the visual approach by not using a dataflow as a centered approach. The reason behind this is the abstraction level we wanted to maintain, and keeping in mind that one of the objectives was to develop an architecture that is applicable to other projects.

7.2 Conclusion of main question

How can we develop a data processing architecture supporting prediction algorithms for NICU bed capacity management?

The main objective of this thesis was to analyze data processing foundations and create an architecture that would guide the data experts of Predict to create an efficient real time data flow. In order to realize this goal, a thorough review of data processing foundations has been executed. Next, the algorithm development has been researched to understand the challenges the data experts had to face. After establishing the activities that need to be managed, the requirements had to be identified. These requirements are focussed on what the real time data process should look like, and how it can be maintained. The requirements resulted in tasks that need to be executed to create a data flow. These tasks are the foundation of the architecture. We aimed at specifying the tasks that need to be executed to create a dataflow. We structured our architecture according to a layered approach. This helped separating the responsibilities of the tasks and who they are assigned to in different layers. Each layer contributes to the dataflow and the data management surrounding the dataflow. The architecture can also help in making decisions about what data processing approach to use when the algorithm starts running. We have discussed two methods in section 5.3, near real time ETL and data virtualization. When the activities are worked out in more detail it can help in making a choice between these two methods. For now, Predict can use either one of them to realize their real time data flow. The data that needs to be extracted is not that much and not complex, and can therefore be managed within both methods.

Chapter 8

Discussion

8.1 Limitations

Our intended validation method was to organize a focus group with several data experts from ADAM and some IT experts within the hospital, to evaluate the architecture and discuss the relevance and benefits of having a solid data architecture in place. Unfortunately, due to the nationwide lockdown and the shift in the hospitals priorities, we were not able to bring the right experts together. However, we hope that in the future our architecture will be a topic of discussion between the data experts of ADAM, the IT department and the current data architect, to improve their collaboration, encourage new data projects that include algorithms in a real time environment, and improve the data processing environment in the hospital to support real time and batch projects.

Another limitation has been the access to real world data in the data sources, and gain more insight in how the data is transformed in practice. Unfortunately, we had limited access to the data. Therefore, all the examples in this thesis are created on an abstract level, not using any real world data.

8.2 Future research

First, we have investigated one data project within the ADAM program. However, ADAM consists of nine project that use predictive algorithms in a real time environment. The other projects overlap when it comes to the use of similar data. In other words, the extracted data for one project can partially be used for other projects. We believe that our architecture can help the other projects implement their algorithm in the production phase. However, we are not aware of the characteristics of the other projects, hence we can not establish that this architecture could function as foundation for implementing the algorithms within ADAM.

Furthermore, we have explained the importance of creating and maintaining a centralized data environment where batch processing and real time applications can coexist and be fully utilized. However, in practice managing these two data processing methods becomes a complex process. We believe that with the right infrastructure and data architecure, the two methods could complement each other. To execute this, data experts have to look at a much bigger architecture which is mainly focussed on the data infrastructure of the whole hospital. Creating an overview of the batch- and real time applications including specifications (software, data models, data standards) will help the hospital to work towards a unified data driven environment that supports both batch and real time projects.

Finally, this research also presented new research areas for every activity in the data process. For example, the standards that are used in healthcare to optimize the data quality. Research can be conducted to the quality of data in healthcare and how standards can be used or modified to support the quality of the data.

Bibliography

[1] Data Analytics Projects, https://www.umcutrecht.nl/nl/Over-Ons/Wat-we-doen/Data-analytics/Data-analytics-projecten

[2] Philip, A. G. :The evolution of neonatology. Pediatric research, 58(4), 799 (2005).

[3] Buonocore, G., Bracci, R., & Weindling, M. (Eds.).: Neonatology: a practical approach to neonatal diseases. Springer Science & Business Media (2012).

[4] Preterm birth, https://www.who.int/news-room/fact-sheets/detail/preterm-birth

[5] Interview: Clinician 1 (2019)

[6] Camacho-Gonzalez, A., Spearman, P. W., & Stoll, B. J.: Neonatal infectious diseases: evaluation of neonatal sepsis. Pediatric Clinics of North America, 60(2), 367 (2013).

[7] Interview: Data architect (2019)

[8] Interview: Data Manager 1 (2019/2020)

[9] Interview: Data Manager 2 (2019)

[10] Akcali, E., Coˆté, M. J., & Lin, C.: A network flow approach to optimizing hospital bed capacity decisions. Health Care Management Science, 9(4), 391-404 (2006).

[11] de Groot, D., Bras, S.: Ortec (2019)

[12] Interview: Prof dr. Hans, E (2019)

[13] Interview: Data Scientist 1 (2019)

[14] Hevner, R. Alan von et al.: Design science in information systems research. In: MIS quarterly 28.1, pp. 75–105 (2004).

[15] Anbeek, N: Predict (2019)

[16] Wohlin, C.: Guidelines for snowballing in systematic literature studies and a replication in software engineering. In Proceedings of the 18th international conference on evaluation and assessment in software engineering (p. 38). ACM (2014).

[17] Zorgen om volle intensive care pasgeboren baby's maandag spoedoverleg: https://nos.nl/nieuwsuur/artikel/2291074-zorgen-om-volle-intensive-care-pasgeboren-baby-s-maandag-spoedoverleg.html

[18] Priyanka, K. and Kulennavar, N.: 'A survey on big data analytics in health care', IJCSIT, 5(4), pp. 5865–5868 (2014).

[19] Feldman, B., Martin, E. M., & Skotnes, T.: Big data in healthcare hype and hope. Dr. Bonnie, 360, 122-125 (2012).

[20] Murdoch, T. B. and Detsky, A. S.:'The Inevitable Application of Big Data to Health Care', Jama. American Medical Association, 309(13), p. 1351(2013).

[21] Hersh, W. R. et al.: 'Caveats for the Use of Operational Electronic Health Record Data in Comparative Effectiveness Research', Medical care. NIH PublicAccess, 51(August), pp. S30–S37 (2014).

[22] El-Sappagh, S. H. A., Hendawi, A. M. A., & El Bastawissy, A. H.: A proposed model for data warehouse ETL processes. Journal of King Saud University-Computer and Information Sciences, 23(2), 91-104 (2011).

[23] Sharda, R., Delen, D., & Turban, E.: Business intelligence: a managerial perspective on analytics. Prentice Hall Press (2013).

[24] Van Der Lans, R.: Data Virtualization for business intelligence systems: revolutionizing data integration for data warehouses. Elsevier (2012).

[25] Kimball, R., L. Reeves, M. Ross, et al. 2008. The data warehouse lifecycle toolkit. Expert methods for designing, developing and deploying data warehouses. New York: Wiley (2008).

[26] Sagiroglu, S., & Sinanc, D.: Big data: A review. In 2013 International Conference on Collaboration Technologies and Systems (CTS) (pp. 42-47). IEEE (2013).

[27] Anuradha, J.: A brief introduction on Big Data 5Vs characteristics and Hadoop technology. Procedia computer science, 48, 319-324 (2015).

[28] Kitchin, R., & McArdle, G.: What makes Big Data, Big Data? Exploring the ontological characteristics of 26 datasets. Big Data & Society, 3(1), 2053951716631130 (2016).

[29] Anuradha, J.: A brief introduction on Big Data 5Vs characteristics and Hadoop technology. Procedia computer science, 48, 319-324 (2015).

[30] Castle, E.: 7 signs you're dealing with complex data: https://www.sisense.com/blog/7-signs-youre-dealing-with-complex-data/

[31] Kong, H. J.: Managing unstructured big data in healthcare system. Healthcare informatics research, 25(1), 1 (2019).

[32] Blumberg, R., & Atre, S.: The problem with unstructured data. Dm Review, 13(42-49), 62 (2003).

[33] Rahm, E., & Do, H. H.: Data cleaning: Problems and current approaches. IEEE Data Eng. Bull., 23(4), 3-13 (2000).

[34] Maletic, J. I., & Marcus, A.: Data Cleansing: Beyond Integrity Analysis. In Iq pp. 200-209 (2000, October).

[35] Müller, H., & Freytag, J. C.: Problems, methods, and challenges in comprehensive data cleansing. Professoren des Inst. Für Informatik (2005).

[36] Inmon, W. H.: What is a data warehouse?. Prism Tech Topic, 1(1), 1-5 (1995).

[37] Lechtenbörger, J.: Data warehouse schema design (Vol. 79). IOS Press (2001).

[38] Ward, M. J., Self, W. H., & Froehle, C. M.: Effects of common data errors in electronic health records on emergency department operational performance metrics: A Monte Carlo simulation. Academic Emergency Medicine, 22(9), 1085-1092 (2015).

[39] Jayawardene, V., Sadiq, S., & Indulska, M.: An analysis of data quality dimensions (2015).

[40] World Health Organization.: Improving data quality: a guide for developing countries. (2003)

[41] Strong, D. M., Lee, Y. W., & Wang, R. Y.: Data quality in context. Communications of the ACM, 40(5), 103-110 (1997).

[42] Wager, K. A., Lee, F. W., & Glaser, J. P.: Health care information systems: a practical approach for health care management. John Wiley & Sons (2017).

[43] De Couveuseafdeling en de NICU, www.kleinekanjers.nl

[44] Alderson, R.: Enterprise Data Architecture, https://medium.com/@rusty.alderson/enterprise-data-architecture-c5c579b54abe

[45] Zorg komt steeds meer personeel tekort we moeten nu nee verkopen: https://nos.nl/nieuwsuur/artikel/2257408-zorg-komt-steeds-meer-personeel-tekort-we-moeten-nu-nee-verkopen.html

[46] Kimball, R., Ross, M., Thornthwaite, W., Mundy, J., & Becker, B.:The data warehouse lifecycle toolkit. John Wiley & Sons (2008).

[47] Bansal, S. K., & Kagemann, S.: Integrating big data: A semantic extract-transform-load framework. Computer, 48(3), 42-50 (2015).

[48] Terpeluk Moss, L., Atre, S.: Business Intelligence Roadmap the complete project lifecycle for decision-support applications. Addison-Wesley Professional (2003)

[49] Jörg, T., & Dessloch, S.: Near real-time data warehousing using state-of-the-art ETL tools. In International Workshop on Business Intelligence for the Real-Time Enterprise (pp. 100-117). Springer, Berlin, Heidelberg (2009).

[50] Ranjan, V.: A comparative study between ETL (Extract, Transform, Load) and ELT (Extract, Load and Transform) approach for loading data into data warehouse. viewed 2010-03-05, http://www. ecst. csuchico. edu/~ juliano/csci693/Presentations/2009w/Materials/Ranjan/Ranjan. Pdf (2009).

[51] Jörg, T., & Dessloch, S.: Near real-time data warehousing using state-of-the-art ETL tools. In International Workshop on Business Intelligence for the Real-Time Enterprise (pp. 100-117). Springer, Berlin, Heidelberg (2009).

[52] Machado, G. V., Cunha, Í., Pereira, A. C., & Oliveira, L. B.: DOD-ETL: distributed on-demand ETL for near real-time business intelligence. Journal of Internet Services and Applications, 10(1), 21 (2019).

[53] Garfinkel, T., & Rosenblum, M.: A Virtual Machine Introspection Based Architecture for Intrusion Detection. In Ndss Vol. 3, No. 2003, pp. 191-206 (2003)

[54] Parnas, D. L.: On the criteria to be used in decomposing systems into modules. Communications of the ACM, 15(12), 1053-1058 (1972).

[55] Rumbaugh, J., Blaha, M., Premerlani, W., Eddy, F., & Lorensen, W. E.: Object-oriented modeling and design (Vol. 199, No. 1). Englewood Cliffs, NJ: Prentice-hall (1991).

[56] Ross, D. T., Goodenough, J. B., & Irvine, C. A.: Software engineering: Process, principles, and goals. Computer, 8(5), 17-27 (1975).

[57] Maier, D., Rozenshtein, D., Salveter, S., Stein, J., & Warren, D. S.: Toward logical data independence: a relational query language without relations. In Proceedings of the 1982 ACM SIGMOD international conference on Management of data (pp. 51-60). ACM (1982).

[58] Van der Lans, R: "Clearly Defining Data Virtualization, Data Federation and Data Integration", https://www.r20.nl/artikelen.htm

[59] https://searchcio.techtarget.com/video/How-data-virtualization-tools-work

[60] Bekker, A.: A comprehensive guide to real time big data analytics, https://www.scnsoft.com/blog/real-time-big-data-analytics-comprehensive-guide

[61] Barlow, M.: Real-time big data analytics: Emerging architecture. " O'Reilly Media, Inc." (2013)

[62] Shiff, L.: Real time vs Batch processing vs Stream processing, https://www.bmc.com/blogs/batch-processing-stream-processing-real-time/

[63] Price, P.: What's the potential for real time analytics in healthcare, https://www.rtinsights.com/whats-the-potential-for-real-time-analytics-in-healthcare/

[64] Charmel, P. A., & Frampton, S. B.: Building the business case for patient-centered care. Healthc Financ Manage, 62(3), 80-5 (2008).

[65] Becker's Hospital Review: 5 Hospital Benefits of Patient-Centred Care, https://healthmanagement.org/c/hospital/news/5-hospital-benefits-of-patient-centred-care

[66] Bresnick, J.: Benefits and Challenges of the Patient-Centered Medical Home, https://healthitanalytics.com/news/benefits-challenges-patient-centered-medical-home

[67] Shaller, D.: Patient-centered care: What does it take? (pp. 1-26). New York: Commonwealth Fund (2007).

[68] Radkiewicz, S.: Why Real-Time health data is a requirement in today's health system, https://hitconsultant.net/2019/08/14/why-real-time-health-data-is-a-requirement-in-todays-health-system/#.XjcyanvvJhE

[69] Luchies, E., Spruit, M., & Askari, M.: Speech Technology in Dutch Health Care: A Qualitative Study. In HEALTHINF pp. 339-348 (2018).

[70] Archenaa, J., & Anita, E. M.: A survey of big data analytics in healthcare and government. Procedia Computer Science, 50, 408-413 (2015).

[71] Kent, J.: Real-Time data use presents Workflow, Quality Challenges, https://healthitanalytics.com/news/real-time-data-use-presents-workflow-quality-challenges

[72] Wirth, R., & Hipp, J.: CRISP-DM: Towards a standard process model for data mining. In Proceedings of the 4th international conference on the practical applications of knowledge discovery and data mining (pp. 29-39). Citeseer (2000).

[73] Chapman, P., Clinton, J., Khabaza, T., Reinartz, T., & Wirth, R.: The CRISP-DM process model. The CRIP–DM Consortium, 310, 91 (1999).

[74] Rolim, C. O., Koch, F. L., Westphall, C. B., Werner, J., Fracalossi, A., & Salvador, G. S.: A cloud computing solution for patient's data collection in health care institutions. In 2010 Second International Conference on eHealth, Telemedicine, and Social Medicine (pp. 95-99). IEEE (2010).

[75] Raj, J.: A beginner's guide to dimensionality reduction in Machine Learning, https://towardsdatascience.com/dimensionality-reduction-for-machine-learning-80a46c2ebb7e

[76] Iroju, O., Soriyan, A., Gambo, I., & Olaleke, J.: Interoperability in healthcare: benefits, challenges and resolutions. International Journal of Innovation and Applied Studies, 3(1), 262-270 (2013).

[77] Renner, S. :A community of interest approach to data interoperability. In Federal database colloquium Vol. 1, p. 2 (2001).

[78] Heubusch, K.: Interoperability: what it means, why it matters. Journal of AHIMA, 77(1), 26-30 (2006).

[79] Doan, A., Halevy, A., & Ives, Z.: Principles of data integration. Elsevier (2012).

[80] Lupşe, O. S., Vida, M. M., & Tivadar, L.: Cloud computing and interoperability in healthcare information systems. In The First International Conference on Intelligent Systems and Applications pp. 81-85 (2012).

[81] Wahono, R. S.: Analyzing requirements engineering problems. In IECI Japan Workshop (2003)

[82] www.chipsoft.nl

[83] www.bma-mosos.com

[84] www.imd-soft.com

[85] Todi, M.: Real Time Machine Learning, https://medium.com/analytics-vidhya/real-time-machine-learning-7aa55dafb2b

[86] Ordish, J., Murfet, H. & Hall, A.,: PHG foundation making science work for health, algorithms as medical devices, PHG Cambridge (2019)

[87] Franklin, M., & Zdonik, S.: "Data in your face" push technology in perspective. In Proceedings of the 1998 ACM SIGMOD international conference on Management of data pp. 516-519 (1998, June)

[88] Xplenty: Data Mapping: An overview of Data Mapping and its Technology, https://medium.com/xplenty-blog/data-mapping-an-overview-aa804bb10998

[89] Khoumbati, K.: Handbook of research on advances in health informatics and electronic healthcare applications: global adoption and impact of information communication technologies. Y. K. Dwivedi, A. Srivastava, & B. Lal (Eds.). Medical Information Science Reference (2010)

[90] Pandey, S., Pandey, R.: Medical (Healthcare) Big Data Security and Privacy Issues. International Journal of Scientific & Engineering Research, Volume 9, issue 2 (2018)

[91] Healthcare Information and Management Systems Society.: HIMSS dictionary of health information technology terms, acronyms, and organizations. CRC Press (2017).

[92] Harrison, R.: TOGAF Version 8.1.1 Enterprise Edition Study Guide. Van Haren Publishing, Zaltbommel (2007)

[93] Grillo, I., Scavone, B.: Why is data architecture important for your business?, https://medium.com/dp6-us-blog/why-is-data-architecture-important-for-your-business-eebcd5a79f9a

[94] Shen, S.: What is the Data Architecture we need?, https://towardsdatascience.com/what-is-the-data-architecture-we-need-72606e71ba0c

[95] Walpita, P.: Software Architecture Patterns – Layered Architecture, https://medium.com/@priyalwalpita/software-architecture-patterns-layered-architecture-a3b89b71a057

[96] Batini, C., Cappiello, C., Francalanci, C., & Maurino, A.:Methodologies for data quality assessment and improvement. ACM computing surveys (CSUR), 41(3), 1-52 (2009).

[97] Marz, N., Warren, J.: Big Data Principles and best practices of scalable real-time data systems, Manning Publications (2015)

[98] Baca, M. (Ed.).: Introduction to metadata. Getty Publications (2016).

[99] Hayes, J.: The theory and practice of change management. Palgrave (2018)

[100] Interview: de Klein, H (2019)

[101] Anonymized company: An Agile Data & Analytics platform for the future (2019)

[102] www.tibco.com

[103] Data Virtualization and ETL, https://community.denodo.com/kb/view/document/Data%20Virtualization%20and%20ETL?tag=ETL

[104] Oemig, F., & Snelick, R.:Healthcare interoperability standards compliance handbook. Cham: Springer international publishing AG (2016).

[105] www.apache.kafka.org

[106] Swart, N.: Handboek Requirements brug tussen business en ICT. Eburon Business, Delft (2010)

[107] Singh, H., Yadav, G., Mallaiah, R., Joshi, P., Joshi, V., Kaur, R., ... & Brahmachari, S. K.: INICU–Integrated neonatal care unit: Capturing neonatal journey in an intelligent data way. Journal of medical systems, 41(8), 132 (2017.)

[108] Interview: Data expert 1 (2020)

Appendix A: Interview protocol Requirements Analysis

Date:
Time:
Location:

Interviewee background
Interviewee name:
Interviewee occupation:

Introduction

- Introduction of student
- Introduction of the interviewee
- Introduction of the project (explain reserach goal)
- Interview elements:
    - Algorithm development
    - Data Extraction
    - Data Integration
    - Data Quality
    - Data Management

Start Interview

A. What kind of data is needed to predict the right bed for a pregnant woman?

Q1: Prior to the project, did you know exactly what data you needed to extract to develop the algorithm?

Q2: Did you use scientific research that explains capacity predictions in general as a reference?

Q3: Is all the data that you need to create the algorithm, included in the datasources?

Q4: Do you also use operational data?

Q5: What is the ratio between unstructured and structured data?

Q6: Do you want to extend the amount of data sources in the future?

B: Data Extraction

Q6: What does the data extraction process look like?

Q7: What applications do you use to extract the data?

Q8: Do you use the same extraction process for every data source?

Q9: What are the challenges while extracting the data?

Q10: Could you tell me more about HIX, MOSOS and MetaVision?

Q11: Could you tell me more about NICU's data environment in general?

Q12: How do the data managers and data scientist collaborate?

C: Data Integration

Q13: When the data is extracted, what is the next step?

Q14: What does the data look like when it is extracted?

Q15: What does the integration process look like?

Q16: How did you divide the tasks with the other data manager?

Q17: Do you have a datamodel that helps integrate the data?

Q18: What applications do you use to integrate the data?

Q19: What are the challenges while integrating the data?

D: Data Cleaning

Q20: Do you check the data for errors and other quality problems?

Q21: Who is in charge of cleaning the data?

Q22: What does the data cleaning process look like?

Q23: What are the challenges while cleaning the data?

Q24: What are the most common quality problems that you find in the data sets?

D: Data Management

Q25: How is the data stored and you could tell me more about the main data sources in the hospital?

Q26: Is there a difference in storing data that is generated internally and externally?

Q27: Is the data that is generated from an external data source (app) connected to the patients data in HIX?

Q28: How do you manage and maintain the different data processes in the hospital?

Q29: Does the hospital work with real time applications? How is the real time data flow managed?

Appendix B: Model Evaluation

Introduction

Q1: Could you provide us with a brief overview of your professional background?

Q2: Do you have any experience with developing or using data architectures?

Phase 1: Strengths/ weaknesses of our architecture

Our architecture is focussed on guiding Predict to create an efficient real time data process. The architecture specifies the tasks that need to be executed to create the required dataflow. We have first executed a literature review to identify the data processing activities that need to be managed during the dataflow. After that, we researched the requirements that need to be taken into account when we design how the data processing activities need to be executed.

Q3: What do you think are the strenghts of our architecture?

Q4: Do you think that the activities in the architecture can solve the current problems in the dataflow of Predict?

Q5: Do you think that the architecture can improve the collaboration between IT and data scientists?

Q6: What do you think are the weak elements of our architecture?

Q7: Are there any elements that you miss in the architecture?

Q8: Do you think the architecture can help the Predict team to create a real time data flow?

Q9: Do you think the architecture could help other real time projects in the hospital to create a real time dataflow? If not, what elements should change in the data architecture?

Q10: If you could extend the architecture, what elements would you add?

Phase 2: General feedback

Q11: If you could change anything about the architecture, what would that be?

Q12: Do you have any other remarks about our architecture that could help improve the dataflow of Predict?