UPPSALA
UNIVERSITET

UPTEC STS 19012
Examensarbete 30 hp
Juni 2019

Using Work Domain Analysis to Evaluate the Design of a Data Warehouse System

Axel Iveroth

UPPSALA
UNIVERSITET

Teknisk- naturvetenskaplig fakultet
UTH-enheten

Besöksadress:
Ångströmlaboratoriet
Lägerhyddsvägen 1
Hus 4, Plan 0

Postadress:
Box 536
751 21 Uppsala

Telefon:
018 – 471 30 03

Telefax:
018 – 471 30 00

Hemsida:
http://www.teknat.uu.se/student

Abstract

Using Work Domain Analysis to Evaluate the Design of a Data Warehouse System

Axel Iveroth

Being able to perform good data analysis is a fundamental part of running any business or organization. One way of enabling data analysis is with a data warehouse system, a type of database that gathers and transforms data from multiple sources and structures it in the goal of simplifying analysis. It is commonly used to provide support in decision-making.

Although a data warehouse enables data analysis, it is also relevant to consider how well the system supports analysis. This thesis is a qualitative research that aims to investigate how work domain analysis (WDA) can be used to evaluate the design of a data warehouse system. To do so, a case study at the IT company Norconsult Astando was performed. A data warehouse system was designed for an issue management system and evaluated using the abstraction hierarchy (AH) model.

The research done in this thesis showed that analysis was enabled by adopting Kimball's bottom-up approach and a star schema design with an accumulating snapshot fact table. Through evaluation of the design, it was shown that most of the design choices made for the data warehouse were captured in the AH. It was concluded that with sufficient data collection methods, WDA can be used to a large extent when evaluating a data warehouse system.

Handledare: Patrik Backentoft
Ämnesgranskare: Anders Arwestrom Jansson
Examinator: Elísabet Andrésdóttir
ISSN: 1650-8319, UPTEC STS 19012

Sammanfattning

Att uppnå förbättrad effektivitet inom offentlig sektor är ett eftersträvansvärt mål för alla samhällen. Eftersom verksamheter som exempelvis vård och omsorg, rättsväsendet, kollektivtrafik och infrastruktur bedrivs med hjälp av skattemedel är det av stor vikt att varje krona används på bästa möjliga sätt för att främja effektiviteten. Ur ett globalt perspektiv är den offentliga sektorn i Sverige särskilt stor och där stat, landsting och kommun har betydande roller i hur landet styrs. I och med digitaliseringen använder dessa aktörer idag en rad olika digitala verktyg för att bedriva sitt dagliga arbete. Några exempel på detta är Stockholm stads e-tjänster för olika typer av ansökningar, Arbetsförmedlingens tjänst Platsbanken och digitala vårdtjänster. IT företaget Norconsult Astando utvecklar och underhåller liknande typer av tjänster för offentligt finansierade verksamheter där vägnätsanknuten information används. Ett sådant system är ISY Case, ett ärendehanteringssystem som används av bland annat kommuner för att hantera ärendeprocesser kopplade till gator och vägar.

Gemensamt för ovannämnda verktyg och tjänster är att de genererar en stor mängd värdefull information som kan användas som beslutsunderlag, upphandlingar eller för att på andra sätt styra och planera de offentliga verksamheterna. Emellertid är den data som genereras i vissa fall inte lämpad för att stödja dessa typer av analyser och måste därför förberedas, hanteras och lagras för att göra analysering möjlig. En metod för att möjliggöra analysering av data är genom ett *data warehouse*, en typ av databas som sammanställer och strukturerar data från flera datakällor i syfte att underlätta analysering. Det är starkt förknippat med *business intelligence* och idag lagrar många företag och verksamheter sin data i data warehouses för att använda det som stöd vid olika beslutfattningsprocesser. Liksom med databaser finns det flertalet alternativ när det kommer till design och utformning av ett data warehouse.

Även om ett data warehouse möjliggör för analysering är det även relevant att undersöka hur lämpligt ett sådant systemet är för att besvara de analytiska kraven. Eftersom ett data warehouse kan klassificeras som ett socio-tekniskt system kan en arbetsdomänanalys (WDA) utföras för att evaluera systemet. WDA är huvudsakligen utformad för att beskriva befintliga system och därför kan det anses okonventionellt att använda WDA som ett evalueringsverktyg. Trots det finns det viss tidigare forskning som pekar på dess fördelar vid evaluering av system.

Syftet med denna kvalitativa studie är att undersöka hur WDA kan användas för att evaluera en design av ett data warehouse system. En fallstudie utfördes på Norconsult Astando. Ett data warehouse system designades från ISY Case och evaluerades genom WDA. Studien använde litteraturstudie, semi-strukturerade och ostrukturerade intervjuer samt WDA. Framställningen av data warehouset följde vattenfallsmodellen, Pentaho Data Integration (Kettle) användes som ETL verktyg och data warehouset lagrades i en PostgreSQL databas. Resultatet visade att kraven för analysering uppfylldes genom att följa en ”nedifrån-upp” metodik förespråkad av Kimball med en stjärnschema design. Genom att utföra WDA kunde systemet evalueras för att kartlägga vilka delar av designen som uppfyllde systemets krav och syfte. Slutsatsen i denna studie var att WDA kan användas för att evaluera data warehouse system i en stor utsträckning, givet att lämplig metodik används. Vidare forskning innefattar att jämföra olika designer för data warehouse och att utforska hur WDA kan användas som evalueringsmetod för andra data warehouse system.

Acknowledgement

This master thesis is the result of the last project assignment in the Master Programme in Sociotechnical Systems Engineering at Uppsala University. The project was done in collaboration with the company Norconsult Astando in Stockholm during the spring semester in 2019.

First of all, I would like to express my utmost gratitude to the kind people at Norconsult Astando who assisted me in every possible way and let me work with them. A special thank you to the supervisor of this thesis, Patrik Backentoft, for encouraging me and guiding me through the work. Finally, thank you Anders Arwestrom Jansson for taking on the task as subject-reader for this thesis and for providing valuable expertise and feedback.

Axel Iveroth

Uppsala, June 2019

Table of Contents

1. Introduction

Hans Rosling once said: “Good analysis is very useful when you want to convert a political decision into an investment. It can also go the other way and drive policy” (Barone, 2007). With his passion for data and statistics, the late physician often found new ways to describe our society and eliminate obsolete beliefs of it. Rosling believed that data analysis could be used to shape policy and ultimately make the world a better place.

Data analysis can be defined as examining information to find something out or to help with making decisions (Cambridge Dictionary, 2019). It is indubitably a fundamental part of running any type of business or organization because the information it provides can be used to lower costs, make processes more efficient and generate a higher profit. Therefore, it is of no surprise that the concept of the data warehouse, a special type of database made for data analysis, has become popular in recent years. Through a complex computing procedure, a data warehouse gathers current and historical data from multiple sources and structures it with the goal of simplifying the analysis process. In doing so, businesses and organizations are able to use the data warehouse as a support when planning their work and when making important business decisions (Elmasri and Navathe, 2016).

Norconsult Astando is a Stockholm-based IT company that specializes in navigation, road databases, and GIS, and their clients are predominantly municipalities. The company’s vision is to improve society through the use of information technology. To reach this vision, Norconsult Astando develops and distributes multiple sustainable IT-solutions that provide support in matters where data from traffic- and road networks is used. One such system is ISY Case, an issue management system which handles processes for different types of street- and roadwork (Norconsult Astando, 2019). The system records information about how much time is spent on different work projects, which companies are involved in the work, and how resources are allocated between different projects, among other data. Although this data is valuable to the municipality, they still have no way to visualize, interact with or aggregate the data. This is where the concept of data warehouse comes into play and with it, the municipalities using ISY Case would be able to utilize their data.

A data warehouse can be classified as a dynamic system, which includes social dimensions and heterogeneous perspectives, and is defined by Vicente (1999) as a complex sociotechnical system. Led by the previous work from Rasmussen et al. (1994), Vicente (1999) developed a framework in the late 1990s for analyzing such systems. This became known as Cognitive Work Analysis (CWA). The framework consists of five phases, the first of which is Work Domain Analysis (WDA). In broad terms, WDA describes the environment in which a system operates and is used to create an understanding of where actions take place. Vicente (1999) illustrates WDA with

1978 Nobel laureate Herbert Simon's metaphor about the ant on a beach. The path of an ant as it zigzags on a beach to avoid obstacles and reach its destination can look complicated and be difficult to describe. But the complexity is really in the surface of the beach, rather than complexity in the ant. So, in order to fully understand a complex system, it is necessary to study the territory or the environment in which it is performed (Vicente, 1999).

WDA is generally used to describe existing systems and acquire knowledge about their domain. However, there has been some research where WDA has been used for evaluating systems (Carlson, 2018; Jenkins et al., 2011; Naikar and Sanderson, 2001). Analysis can be enabled by designing a data warehouse system. But by evaluating its design, a measurement is provided which indicates how well the system supports analysis. And as Hans Rosling stated, being able to perform good analysis is of high value when driving policy.

1.1 Aim

The aim of this thesis is to investigate how WDA can be used to evaluate the design of a data warehouse system. To do so, a case study at Norconsult Astando was performed. The thesis sets out to achieve two goals: to design a data warehouse of Norconsult Astando's system ISY Case and to evaluate its design by performing WDA.

1.2 Research Questions

The goals have been formulated into the following specific research questions:

- What data warehouse design can meet Norconsult Astando's requirements for enabling analysis of ISY Case?
- To what extent can WDA be used to evaluate the design of the ISY Case data warehouse system?

1.3 Delimitations

The concept of the data warehouse is detailed and complex. It can take several years of designing and developing before a fully functional system can be attained (Inmon, 2002). Therefore, some delimitations were made in this thesis regarding the data warehouse design. First, the ISY Case system supports several different types of work (cases) that follow different processes. The data warehouse constructed in this thesis was for one type of work, namely excavation work (Sw. schakt). This meant that analysis was made possible only for a part of the ISY Case system. However, because of the similarities in the data structure, the design choices that were made are applicable to other work types as well. Second, the focus of this thesis was on the design and usage of the data warehouse. Therefore, it did not prioritize the performance of extracting, transforming and loading (ETL) data into the data warehouse.

Regarding the CWA framework, this thesis included only the first phase out of the five, due to the scope of the thesis. Moreover, the WDA that was performed is specific to Norconsult Astando and ISY Case. Consequently, it may not be directly applicable on other businesses or issue management systems.

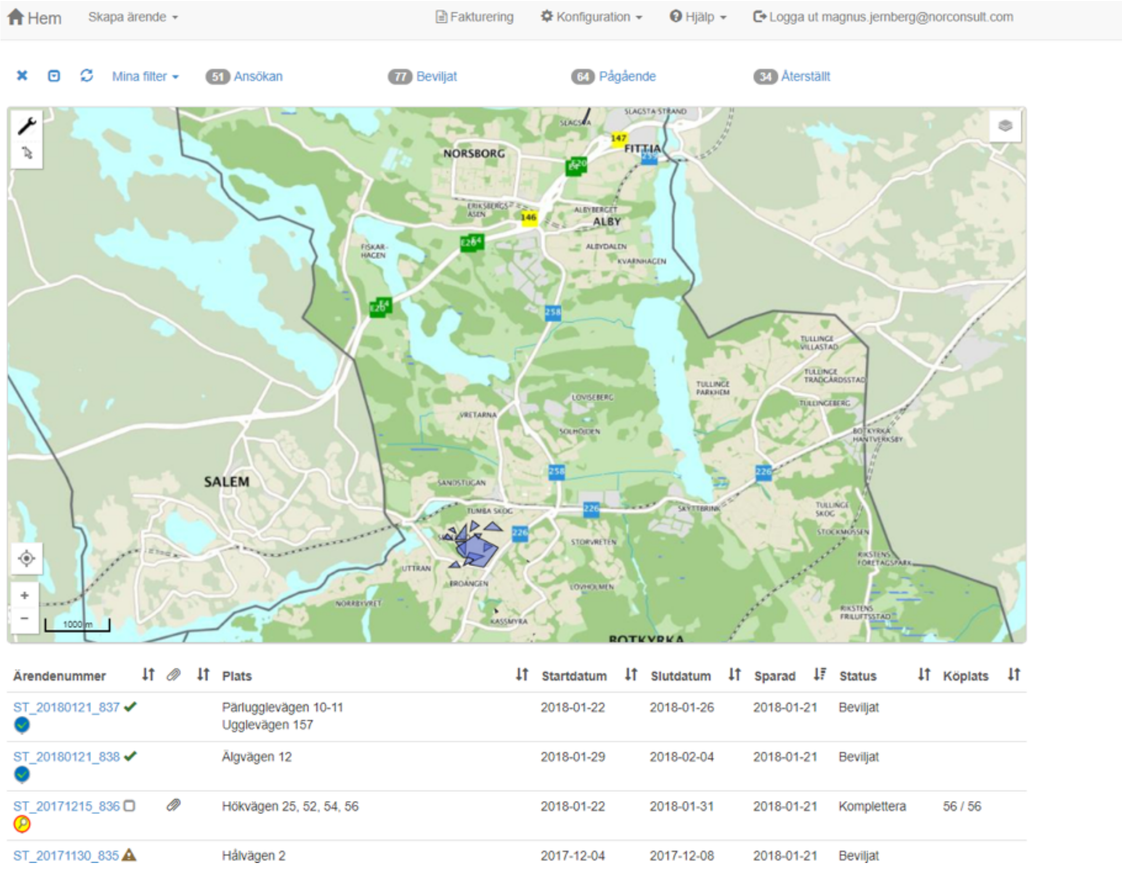
2. Background

This chapter will present the relevant background information about the ISY Case system. Section 2.1 examines ISY Case and goes into detail about the schakt module and the process of a schakt case. Section 2.2 explains the motivation for analyzing cases and section 2.3 compares ISY Case to a similar system used by Trafikkontoret, Gatuarbete Webb.

2.1 ISY Case

Imagine that a new house is to be built on a piece of land. In order to provide stability and structure to the house, a foundation needs to be constructed. This is done through excavation work. But before the actual digging can start, an application needs to be issued to the municipality to assure that all rules and regulations are followed. Once the municipality grants the application, each step in the excavation process has to be approved and a final inspection has to be done once the digging is complete. These are just some parts of the complex process that includes several people and operations. Norconsult Astando has developed a web client-based system, ISY Case, for managing these kinds of processes. It handles different kinds of *cases* other than earthwork and supports the complete process from issuing an application to completion of case. By using ISY Case, those involved in the excavation work for the residential area have an accessible system where they can issue a case, search for ongoing cases or view the status of a case (Norconsult Astando, 2017).

Other than earthwork, ISY Case supports several other kinds of processes including traffic arrangement (e.g. setting up traffic signs), ground leasing (e.g. setting up a stand) and traffic incidents. These types of cases are referred to as *modules*. Through the lifetime of a case, it is assigned different *statuses* that indicate what stage of the process the case is at. For instance, a case with the traffic incident module can have the following statuses: received, ongoing, finished, finished without action or canceled. When logging in to ISY Case, a page with search options is shown as the one in Figure 1. From here, the user can filter cases on modules (“verksamhet”), current status, date, location or other attributes. The results are shown both on the map and in a list (Norconsult Astando, 2017).



Ärendenummer		Plats	Startdatum	Slutdatum	Sparad	Status	Köplats
ST_20180121_837 ✓		Pärlugglevägen 10-11 Ugglevägen 157	2018-01-22	2018-01-26	2018-01-21	Beviljat	
ST_20180121_838 ✓		Älgvägen 12	2018-01-29	2018-02-04	2018-01-21	Beviljat	
ST_20171215_836 □	📎	Hökvägen 25, 52, 54, 56	2018-01-22	2018-01-31	2018-01-21	Komplettera	56 / 56
ST_20171130_835 ⚠		Hålvägen 2	2017-12-04	2017-12-08	2018-01-21	Beviljat	

Figure 1. The ISY Case web-based interface (Jernberg, 2019, "ISY Case Schakt & TA" [PowerPoint presentation]).

2.1.1 ISY Case schakt

Schakt is the ISY Case module for handling earthwork, as described by the example in section 2.1. There are three main roles connected to a schakt case and their relation is seen in Figure 2. At the highest level is the municipality, which owns the land where the work is being done. It is their responsibility to review incoming cases, examine them and see to it that all the necessary requirements are fulfilled along the process. The responsible administrator is the company who sets the work-process in motion by signing an agreement with the municipality, towards which they are legally responsible. As for the actual work, the responsible administrator assigns different entrepreneurs to the case. While the municipality and the responsible administrator define the requirements of the work and decide what should be done, the entrepreneurs are the main users of ISY Case. Whenever information about a case is altered or a change of status occurs, it is registered in ISY Case by the entrepreneurs.

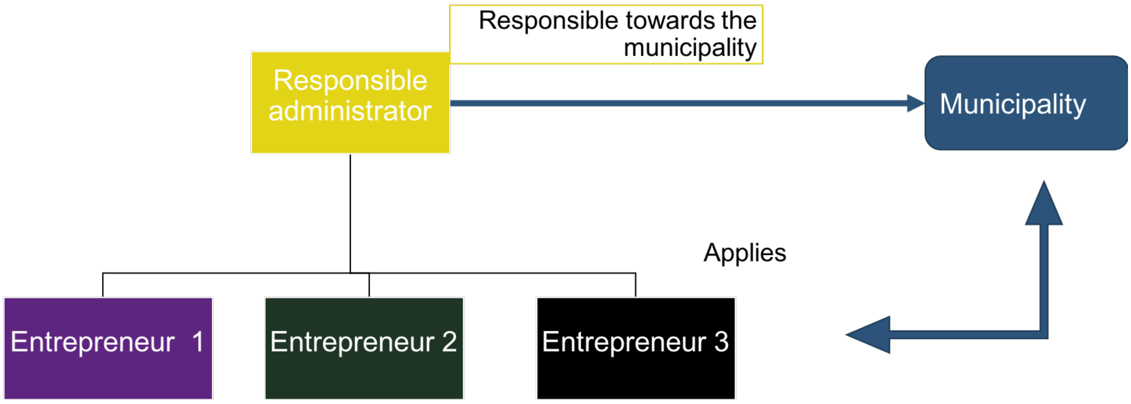

Responsible towards the municipality
Responsible administrator
Municipality
Applies
Entrepreneur 1
Entrepreneur 2
Entrepreneur 3

Figure 2. Roles of an ISY Case schakt case (Adapted from Jernberg, 2019, "ISY Case Schakt & TA", [PowerPoint presentation]).

2.1.2 ISY Case schakt case flow

A schakt case's flow is divided into three phases and twelve statuses, which can be seen in Figure 3. In the following paragraphs, an example is described that will illustrate the flow of a case as it goes through different phases and statuses.

Suppose that a responsible administrator has received a job to install an optical fiber cable under an asphalt road. Before the work starts, the responsible administrator signs an agreement with the municipality owning the road, stating that they are allowed to dig in the area. When the agreement has been signed, the responsible administrator assigns the job to an entrepreneur, which will do the digging of the hole for the cable.

The entrepreneur creates a new case in ISY Case by filling out all the necessary details about the case, such as the type of work, the location, a short description, when the digging will commence and so on. Once the case has been submitted in the system, it receives the status New Case and the Administrating Processing Phase is initiated. Upon noticing that a new case has entered in ISY Case, the municipality starts to examine the application. When this happens, the case receives the status Applied. From here, the case can proceed to two different statuses. Either the municipality deems that the application is up to standard and fulfills all the requirements necessary to install the optical fiber cable, in which the case is assigned the status Grant. However, if the application is faulty in some way, e.g. a start-date for the work is missing or a more detailed description is required, the status is set to Complement. The entrepreneur has to correct the faulty information in the application. When the entrepreneur has addressed the issue, the case is set to status Applied a second time and examined by the municipality again, thus restarting the examination process. It can happen that a case goes back and forth between Applied and Complement several times before receiving a Grant status. When the case finally is set to status Grant, the Administrative Processing

Phase ends. At any time during the Administrative Processing Phase, a case can be canceled.

When the case receives the Grant status, the entrepreneur can begin the actual digging of the hole where the optical fiber cable is to be installed. The case is set to the status Ongoing and the second phase, the Execution Phase, is initiated. When the entrepreneur is done digging and the cable has been installed, the site needs to be restored by filling in the hole with asphalt again. Upon completing this task, the status is set to Restored. In some situations, the case can receive the status Remark, meaning that some additional work has to be done.

The final phase, the Validation Phase begins when the municipality does a Final Inspection at the excavation site. The municipality checks if the hole is proper and that the instructions have been followed. If the municipality is satisfied with the work, the status is set to Approved, otherwise, it is set to Unapproved Inspection. Similar to the flow in the Administrative Processing Phase, the case can alternate between Unapproved Inspection and Final Inspection before it gets status Approved. Finally, a warranty inspection is carried out a few years later in order to validate that the excavation-work appears as expected from the elapsed time or if some of the work has to be redone.

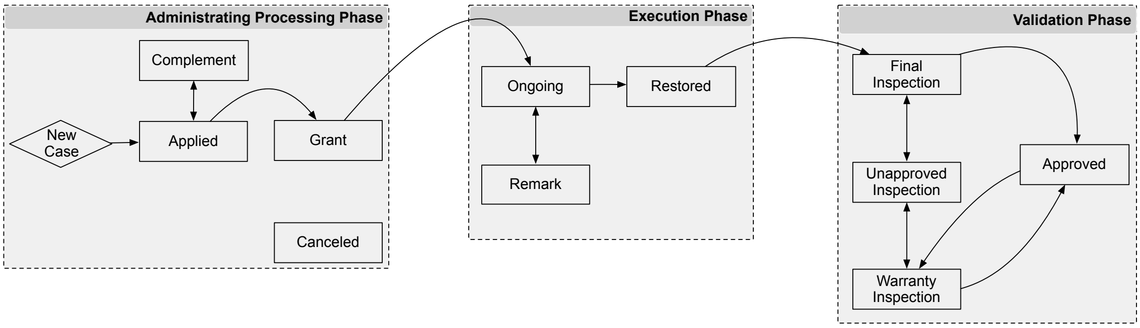


Figure 3. ISY Case schakt case flow (Adapted from Jernberg, 2019, "ISY Case Schakt & TA" [PowerPoint presentation]).

2.2 Analyzing Cases

With the example from above, it is clear that the flow of a case in ISY Case is complex and that the process from New Case to Approved can look very different from one case to another. For some cases, the complete process is done in a matter of days while for others it can take several months or even years. The reasons for this are nearly endless. For example, take the optical fiber cable case from before. In that scenario, it might be that the digging of the hole was impossible to perform due to the weather or that the application was missing some information, hence hindering it from being granted by the municipality. To find out what the process looks like for a case and identify where it is being delayed, it is necessary to view how much time is spent in each status. This is defined as the *lead-time* of a status. For instance, the lead-time for status Applied describes the time it takes for a case to be examined by the municipality. Another example is the lead-time of New Status, which tells how long it takes for the municipality to take on the case from the time it enters the system. Having access to the lead-times of a case opens up the possibility to examine the cases and their flow. Further, the lead-times can be used to answer analytical questions about cases such as the duration of a phase for a case, how time and work is divided between the municipality and the entrepreneurs, how many cases are applied and approved, and so on.

These types of analytical questions are of great interest to the municipality since they can provide reports and statistics that can be used as a support in different decision-making processes and municipal procurements. For instance, a municipality can analyze the number of incoming cases to decide how many cases they can grant during different time-periods. Another way the information can support decision-making is when deciding how much resources are allocated to different cases. It can also be of help when evaluating how the entrepreneurs involved in a task have performed. These are some examples of how the decision-making process can be benefitted by having access to this type of information.

The statistics and reports can provide details on how the municipality organize their own work. Some municipalities, for example, have a limit of 10 days to take on a case when it enters ISY Case (Jernberg, Interview 4, 2019). By having the lead-times of the incoming cases, the municipality would be able to evaluate how they handle their internal work and business. Further, since municipalities are a part of the government, it is important that they utilize their publicly financed resources in the best possible way. Enabling analysis, thereby making it possible to handle cases cost-effectively, can be a way to achieve this.

In its current state, the ISY Case system offers no clear-cut way for the user to view the lead-times for cases, thus hindering the analytical possibilities. If a municipality wants to view statistics or develop reports on the case data, Norconsult Astando has to produce these documents for them by aggregating the data. Not only is this a time-consuming task, but it requires clear communication between Norconsult Astando and the municipality. In order to enable analysis of cases for municipalities, a new type of system must be introduced.

2.3 Gatuarbete Webb

Gatuarbete Webb is the equivalent to ISY Case for Trafikkontoret, the public administration responsible for traffic- and roadwork in Stockholm. Trafikkontoret uses Gatuarbete Webb to handle and process excavation applications, similar to the way municipalities use ISY Case for schakt cases. Unlike ISY Case though, the system stores the lead-times of cases. This allows for users of Gatuarbete Web to view and study the lead-times of separate cases. However, in order to generate reports or statistics, the user is required to extract the data from Gatuarbete to external software such as Microsoft Excel where different calculations and operations can be done (Brundin, Interview 6, 2019).

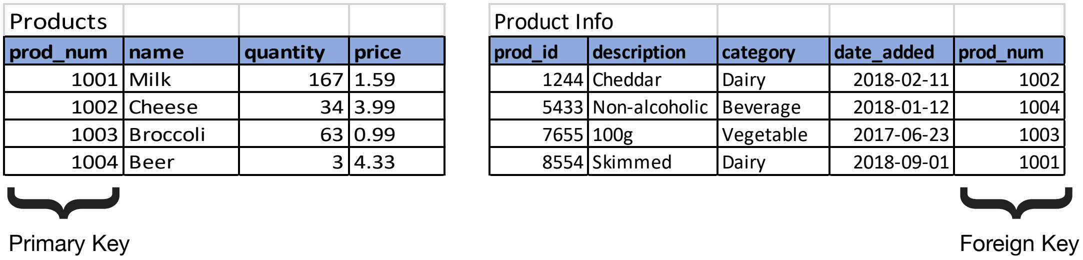
3. Theory

This chapter describes the data warehouse theory. Section 3.1 gives a general overview of relational database concepts that the data warehouse theory can be explained from. Section 3.2 describes the definition of a data warehouse, section 3.3 its architecture and the final section, 3.4, goes through the alternatives for design of a data warehouse.

3.1 Relational Database Overview

A *database* is a collection of related data stored in a way so that information can be retrieved from it. *Data* is defined as known facts that have an implicit meaning. In this context, information such as names, addresses or phone numbers is regarded as data. In a way, databases are the foundation of many content-driven websites and applications today. Whenever information regarding a product, some orders, a person or anything else needs to be stored, a database is used. A user commonly interacts with the database by sending requests to it. This is called sending *queries* (Elmasri and Navathe, 2016).

A *relational database* presents the data in a *table,* which is a collection of related information that consists of *rows* and *columns.* Each row represents a record and each column corresponds to an attribute. The Products table on the left-hand side in Figure 4 shows a grocery store's different products, one for each row, and their attributes (prod_num, name, quantity, price) in different columns. In relational databases, there can be relationships between different tables. This is made possible with the use of *primary keys.* A primary key is a column or a set of columns in a table that has a unique value for that row. The primary key in the Products table would generally be "prod_num" since it is unique for each row. Propose that a second table, Product Info, is introduced and one wishes to combine it with the Products table to examine all information about a specific product. In order to link these tables together, there must exist a column that appears in both tables. This column is called the *foreign key* in the Product Info table and references to the primary key in the Products table. For instance, linking the tables on prod_number 1002 would result in name, quantity, price, prod_id, description, category and date_added data about the product (Oracle, 2017). System generated keys that serve no meaning outside the database, like prod_num in the Products table, are called *surrogate keys*. In contrast, a *natural key* is a unique key that has some meaning outside the database. The visual representation of different tables and their relationship is called a *logical schema* (Elmasri and Navathe, 2016).



Products

prod_num	name	quantity	price
1001	Milk	167	1.59
1002	Cheese	34	3.99
1003	Broccoli	63	0.99
1004	Beer	3	4.33

Product Info

prod_id	description	category	date_added	prod_num
1244	Cheddar	Dairy	2018-02-11	1002
5433	Non-alcoholic	Beverage	2018-01-12	1004
7655	100g	Vegetable	2017-06-23	1003
8554	Skimmed	Dairy	2018-09-01	1001

Primary Key
Foreign Key

Figure 4. The Products table and Product Info table with its indicated primary and foreign key.

3.2 Data Warehouse Definition

Databases are designed to support the daily operations of an organization and provide fast, concurrent access to data (Vaisman and Zimányi, 2014). In the grocery store example from the previous section, the Product and Product Info tables can be edited whenever a new product is added or information about an existing product is updated. A *data warehouse*, on the other hand, is a type of database that is used solely for data analysis. Rather than routine data processing and modification, a data warehouse is targeted towards data retrieval. One way of visualizing it is to view it as a repository that gathers and transforms data from various sources in order to use it for analyzing data (Elmasri and Navathe, 2016). The main purpose of a data warehouse is data analysis for support in decision-making (Jukic, 2006).

Given the idea of a data warehouse, what characterizes it? A commonly accepted definition is the one from William Inmon, which states that a data warehouse is a subject-oriented, integrated, nonvolatile and time-variant collection of data (Inmon, 2002).

- The first property, *subject-orientation*, is that data warehouses can be used to analyze a desired subject area. For instance, a retail store might be interested in analyzing a certain product, the store's stock or the overall sales. Each subject is then regarded as the store's subject area. Another example is that of a manufacturer, whose subject area can include product, order, and vendor.
- The second property of a data warehouse is that it is *integrated*, meaning all inconsistencies has to be removed. An example of this is gender data: Presume that data is gathered to a data warehouse from two sources: one that labels gender as m/f and the other label it as 0/1. The property of integration ensures that the data in the data warehouse is encoded as one of these formats.
- *Non-volatile* is the notion that once data has been loaded into a data warehouse, it is static and cannot be updated, created or deleted. The data is read-only.
- The final property, *time-variant*, states that historical data can be stored in a data warehouse. In contrast to traditional databases, which commonly stores the most recent value, a data warehouse allows users to obtain older data.

3.2.1 Comparing data warehouse to a traditional database

Why go through the process of constructing a data warehouse when analysis can be done by simply sending queries to the database? The answer lies in the fundamental differences between the two concepts. A traditional database supports *online transaction processing* (OLTP) which includes insert, update and delete operations. They are optimized to process queries that are carried out frequently and repeatedly and modifies a small part of the database. An example of an OLTP system is that of a bank, which allows insertion of new transactions, modification of balances and deletion of old accounts. *Online analytical processing* (OLAP) systems, on the other hand, are those systems that support efficient extraction, processing, and presentation of data, such as a data warehouse. OLAP systems are less frequently accessed in comparison to OLTP systems, but the queries are more complex and involve aggregations (Elmasri and Navathe, 2016; Vaisman and Zimányi, 2014). One way of differentiating between the systems is to view OLTP as a customer-oriented system that is used by clerks, clients, and IT-professionals as opposed the market-oriented OLAP system whose users are managers, executives, and analysts (Han and Kamber, 2011).

According to Han and Kamber (2011), one major reason for separating the data warehouse from the database is to promote high performance. A database is tuned to handle tasks like indexing using primary keys or searching for a particular record, while the queries sent to a data warehouse are more complex and involve scanning of summarized data. Due to the systems being optimized for different kinds of operations, processing OLAP queries on an OLPT system would degrade the performance of operational tasks severely. Separating the systems promotes a higher performance of each system since they do not have to compete for the same computing resources (Han and Kamber, 2011).

Another reason for distinguishing between the systems is design. Jukic (2006) argues that it is difficult to structure a database that can be queried for both analytical and operational purposes in a straightforward way. As shown in a later section in this chapter, the data warehouse star schema design is different from that of the entity-relationship schema of a database. Moreover, the contents of an OLTP system are detailed raw data while an OLAP system contains processed data of high quality (Han and Kamber, 2011).

3.3 Data Warehouse Architecture

Figure 5 shows the general architecture of a data warehouse. It depicts the process of collecting data from data sources and loading it into the data warehouse. Starting from the left in Figure 5, the source systems are those systems that contain the information that is to be analyzed. These are commonly databases, files in various formats or other types of data sources (Elmasri and Navathe, 2016). The process of extracting, transforming and loading the source data is handled by the ETL system and is described in the next subsection. The ETL system brings the processed data from the data staging area into the data warehouse. From here, it is possible to present the data using reporting tools, statistical tools, data mining tools or SQL queries (Vaisman and Zimányi, 2014). Note that Figure 5 shows a simplified view of the data warehouse architecture. In many real-life business scenarios, the process can be non-linear and go through other stages such as data validation or be transferred to a multidimensional database (Rainardi, 2008).

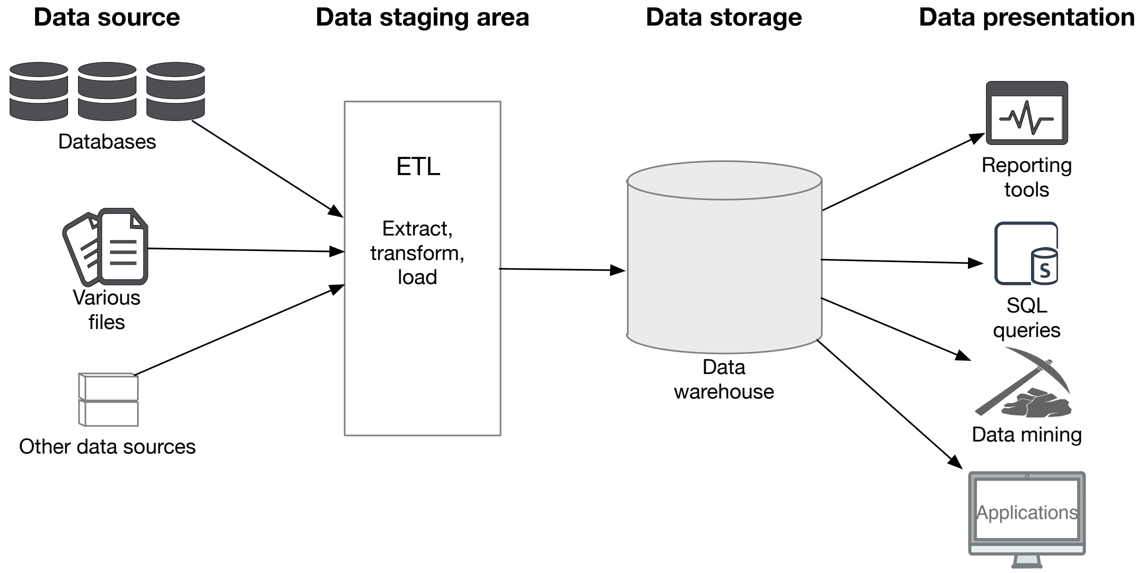


Figure 5. The data warehouse architecture (Adapted from Rainardi, 2008, p. 2).

3.3.1 ETL

ETL (extraction, transformation, loading) is a three-step process that preprocesses the data by retrieving it from the source systems, modifying it and loading it into the target system. The ETL process is widely used whenever data is moved and therefore not exclusive for data warehouses (Rainardi, 2008). *Extraction* is the first step in the process and its purpose is to gather data from multiple heterogeneous data sources. It selects the data that is relevant and excludes the rest. In the *transformation* step, data is modified and prepared for the data warehouse. It includes several aspects such as data cleaning, which removes all errors and inconsistencies and converts the data into one standardized format. Aggregation is performed to summarize the data according to a certain level of detail of the data warehouse. In the final step, *loading*, the transformed data is structured in a data warehouse structure and loaded into the target system. The step also includes refreshing the data to make it up-to-date. Depending on the policy and requirements from the organization, the refresh frequency varies from monthly to daily or in sometimes near to real-time. The ETL process requires a data staging area, which is a temporary database or database tables where the extracted data is modified before it is loaded (Vaisman and Zimányi, 2014).

When designing a data warehouse, the development of the ETL infrastructure is often the most demanding part in terms of time and resources. In some cases, the data warehouse project team can spend up to 70 % of the time on developing ETL functions. However, if a proper data warehouse design is developed, the requirements of the ETL development becomes clear and the success of the process is then a matter of efficient execution (Jukic, 2006). With this in mind, the next section dwells into the different ways of how to approach data warehouse design.

3.4 Data Warehouse Design

The field of data warehouse is considered to have two pioneers: William Inmon and Ralph Kimball. Although their ideas of data warehouse are roughly the same, their methods for designing it differ. They both adopt the concept of a *data mart*, which is a subset of the data warehouse that focuses on a specific department or subject (Jukic, 2006). In the following subsections, Inmon and Kimball's different design approaches are described and compared.

3.4.1 Inmon's top-down design

The top-down design, or the normalized approach, was first proposed by William Inmon and can be seen in Figure 6. Here, the idea is to first design a data warehouse from the entire source data and then build data marts from it. The source data is processed with ETL and once it has been loaded into the data warehouse it is modeled to third normal form (3NF) using database normalization rules. Normalized 3NF structure reduces duplication of data by dividing the data into entities, resulting in a large number of unique tables. (Kimball and Ross, 2013). From the data warehouse, data marts are created based on the desired subject area, such as sales or production. These data marts are restructured from the ER model in the data warehouse to a dimensional model. The principle of the data warehouse being a centralized repository from which data marts are created, along with storing data in third normal form, is what characterizes the top-down design (Jukic, 2006).

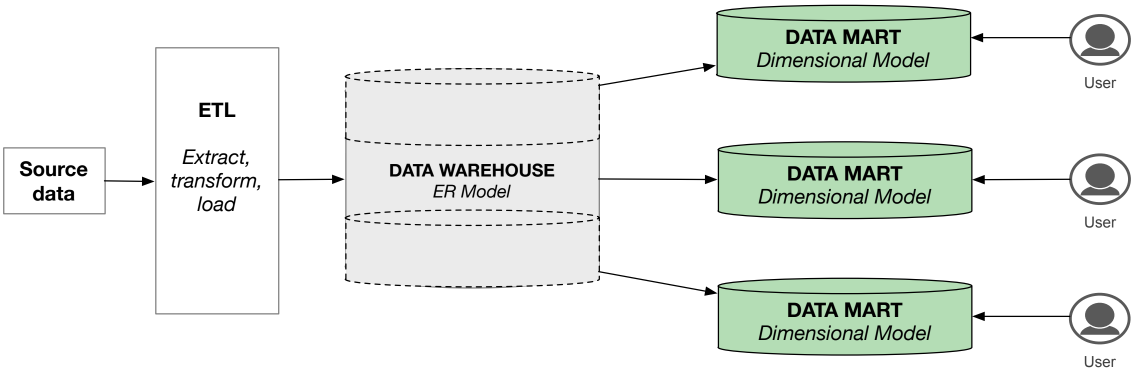


Figure 6. Inmon's top-down design (Adapted from Jukic, 2006, p. 86).

3.4.2 Kimball's bottom-up design

In a sense, Kimball's bottom-up design, also known as the dimensional approach, is the opposite of the top-down design. Instead of building data marts from one central data warehouse, the bottom-up design starts by creating data marts from desired subject-areas and combines them, resulting in a data warehouse. This approach is illustrated in Figure 7. Similar to the top-down design, the source data is first processed through the ETL process. Thereafter, separate data marts are created depending on what data is needed for the data analysis. Multiple data marts are integrated to create a data warehouse. Whereas the top-down design resulted in a ER modeled data warehouse, the bottom-up approach results in a dimensional modeled one (Jukic, 2006).

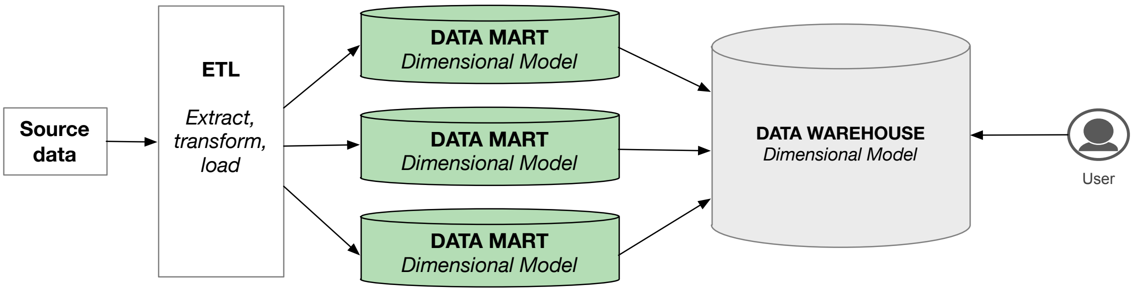


Figure 7. Kimball's bottom-up design (Adapted from Jukic, 2006, p. 86).

3.4.3 Choosing design approach

Given these two methods of design, how does one decide which approach is most suitable? Jukic (2006, p. 87) states that it is a matter of "trade-off between extensiveness and power versus quickness and simplicity". The top-down approach allows for a detailed organizational-wide analysis and the underlying ER model can be used as a basis for other as-yet-unknown analysis in the future. However, the initial task of designing a data warehouse ER model can be demanding in time and recourses. Golfarelli and Rizzi (2009) add to this by stating that the promising results of the top-down approach are hindered by the complexities of bringing together all relevant sources of a business as well as analyzing the need of every relevant department involved. In contrast, the bottom-up design is more suitable to implement when the subject area is limited and takes on a more cautious minded style compared to the top-down approach (Golfarelli and Rizzi, 2009). If an organization requires individual data analysis, the Kimball bottom-up approach is both simpler and quicker to adopt than Inmon's top-down design (Jukic, 2006).

3.4.4 The dimensional model

The data mart used in the top-down and bottom-up designs is modeled with the dimensional model. To get a complete and coherent understanding of the model, the concept will be described through an example by Kimball and Ross (Kimball and Ross, 2013).

Suppose that a retail business collects data at the cash register when a customer makes a purchase. The data stores information that you would normally see on a receipt, such as what items have been purchased, store details, who the clerk is and the time of purchase. The retail business wants to construct a data warehouse to enable analysis of sales-related data to better understand customer purchases. Having this data in a data warehouse enables users to analyze which products are selling in which stores on which days from which clerk.

When adopting dimensional modeling, data is sorted into two types of tables: fact and dimension. A *fact table* contains the quantitative data for analysis. Its columns are divided into either facts or foreign keys. A *fact* is generally some measurements such as the amount of money, number of units or days. It is often easy to identify a fact because they are in most cases continuous numeric values that are additive. The foreign keys are what links the fact table to one or more dimensional tables. In contrast to a fact table, a *dimension table* contains descriptive information that is usually presented in text. Kimball and Ross (2013) express dimensions as describing the *who, what, where, when, how* and *why* associated with the event. Dimensions are used to analyze the facts, often through aggregation operations such as counting, summation, average, etc. (Schneider, 2008). When the facts and dimensions have been identified, they form a schema that includes one or more fact tables referencing to multiple dimensional tables. This is defined as a *star schema*. (Kimball and Ross, 2013).

Returning to the retail business example, the corresponding logical schema of the star schema is shown in Figure 8. Since the business wants to analyze data based on what products are being bought, the dimensional model is structured from individual products that are being purchased. This level of detail is referred to as the *grain* of the model. Rainardi (2008, p. 72) defines grain as "the smallest unit of occurrence of the business event in which the event is measured". The Retail Sales Fact Table contains two numeric measures, SalesQuantity (how many items of a product that are purchased) and SalesDollar. Moreover, it also contains four foreign keys that point to the relevant dimensions. The Product Dimension Table describes what product was purchased, the Store Dimension Table where it was purchased, the Date Dimension Table when the purchase occurred and the Clerk Dimension Table who was working behind the cashier. With this dimensional model, the retail business is able to answer questions such as "find the top five sold products in the Florida region in the month of September the past three years" in a quick fashion with a simple query.

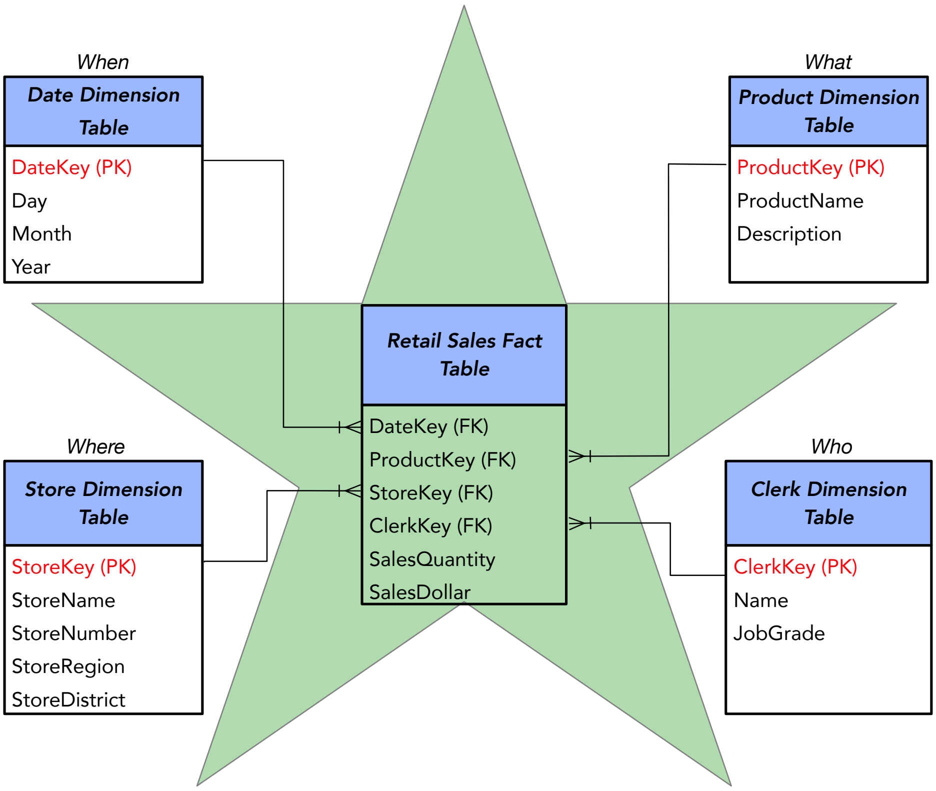


Figure 8. Dimensional modeled data mart in a star schema.

The Retail Sales Fact Table in the star schema in Figure 8 measures the activities of retail sales where each sale corresponds to an event in time and space. This type of fact table is called a *Transaction Fact Table*. Seeing that this type of fact table supports a wide variety of analytical possibilities, it is commonly used in dimensional modeling. However, a business can also be inclined to analyze events with respect to time (Adamson, 2010). Hence, two other types of fact tables are introduced. A *Periodic Snapshot Fact Table* is used when measuring events that occur over a period of time, for example, a day or a week. Here, the grain is the specified period that is being analyzed (Kimball and Ross, 2013). The third and final type that a fact table can belong to is the *Accumulating Snapshot Fact Table*. This type is applicable when one wants to study the elapsed time between events or milestones. Processes that have a defined start- and end point with an intermediate step in between can be modeled with this type of fact table (Adamson, 2010).

4. Methodology

The methodology chapter consists of a review of how the thesis was conducted, seen in section 4.1. The frameworks used and their relevance for this thesis are described in section 4.2 and the data collection methods that were used is given in section 4.3. Section 4.4 lists the development tools used to model and construct the data warehouse.

4.1 How this Thesis was Conducted

The research done in this thesis was initiated by gathering information about Norconsult Astando and ISY Case. This was executed through unstructured interviews with employees at the company. The purpose of the initial research was to get an understanding of the situation and the data available as well as what obstacles existed. Simultaneously to this work, a literature review was performed in the fields of data warehouse and CWA.

After this, the development of the ISY Case data warehouse began. The process followed the waterfall model. Once the design was complete, semi-structured interviews were conducted to collect data to perform WDA. The result from the WDA was used to evaluate the data warehouse design.

4.2 Literature Review

A *literature review* is used by the researcher to increase the understanding of a subject area, test a research question and to examine the methodology that forms parts of the research process (Race, 2008). A large part of the literature used in this thesis were books about data warehouse and the CWA framework obtained from the Uppsala University Library. Journal articles, research papers as well as conference papers, all of which gave academic information about the subjects discussed in this thesis, were provided from publisher databases. For the choices of data warehouse design and guidelines in using the development tools, web-pages and online documentation were used. The use of online sources can in some cases be problematic since they might not be considered academic. However, they are a valuable complement to the other literature on the grounds that they provide the most recent information in a subject area that is evolving. According to Race (2008), it is the job of the researcher to define the boundaries of the literature and underline what aspects a literature review can address.

4.3 Frameworks

4.3.1 Qualitative research

There are many ways to categorize the types of research. Broadly speaking, two categories exist: quantitative research and qualitative research. When conducting a *quantitative research,* the focus is mainly on collecting objective numerical data and the research refers to counts and measures of things. In contrast, a *qualitative research* explores the subjective meanings, concepts, definitions, characteristics, and description of things. (Berg, 2001). The collection of data for qualitative research is commonly done by the researcher in a natural setting. Data is collected from multiple sources through examining documents, observing behavior or interviewing participants. The researcher then makes an interpretation of the collected data (Creswell, 2009). This thesis is a qualitative research on the grounds that its results were based on participants answers to open-ended questions. Rather than statistical evidence, it was the participant's opinions and perceptions that guided the design and evaluation of the data warehouse system.

4.3.2 Case study

One example of qualitative research is the case study. A basic *case study* can be defined as a research approach that studies one or a few instances of a phenomenon in depth (Blatter, 2008). Although the boundaries of the phenomenon in question are rarely clear, it is commonly investigated within its real-life context. Rainer et al. (2012) add to this definition by arguing that the objective of a case study is to increase knowledge and gain an understanding about the phenomenon while simultaneously bringing about change to it. Throughout the years, case studies have received criticism for being of less value when compared to analytical and controlled empirical studies. Advocates of the latter stress that case studies are too difficult to generalize from and is often biased by the researcher. Rainer et al. (2012) suggest that proper research methodologies need to be applied in order for a case study to be successful. This means that the conclusions drawn in this thesis are in the context of Norconsult Astando and that the same results may be inapplicable for another company.

4.3.3 Cognitive work analysis (CWA)

First developed by Rasmussen and further established by Vicente, *cognitive work analysis* (CWA) is a framework used for analyzing complex sociotechnical systems (Jenkins et al., 2008). According to Vicente (1999), a system is considered *complex* if it has a high degree of several pre-defined factors. These factors include large problem space, dynamic, social and heterogenous perspectives. CWA aims to support users in adopting their behavior by systematically identifying the purposes and constraints of the system, rather than the way the work is performed (Stanton et al., 2018). Vicente (1999) states that the analysis should start by finding the *environmental constraints* of the system, meaning the external context in which the users are situated. Thereafter, the focus should be on the *cognitive constraints*, which are those factors that originate with the human cognitive system.

Vicente (1999) argues that CWA differentiates from other frameworks by taking a formative approach to work analysis. By this, he implies that instead of dictating how work *should* be done or how work *is* done, CWA aims to analyze how work *could* be done, given the right tools. This allows CWA to highlight the possible behavior of the system and puts the focus on the requirements instead of the final output (Vicente, 1999). Although the formative approach is one of the strengths of CWA, some see it as its weakness. Stanton et al. (2018) point out that the framework is extremely difficult to be taught and fully understood. Its flexible nature means that the result from one analysis is seldom the same as that from another analyst (Stanton et al., 2018). Fidel and Pejtersen (2004) further critizise CWA in its general approach and lack of instructions on what methods to use or questions to ask. This makes the results of the analysis to be highly dependent on the system's context and it is up to the researcher to select appropiate methods based on the system that is being investigated (Fidel and Pejtersen, 2004).

The CWA framework consists of five phases, starting from the environmental perspective and moving towards the cognitive perspective. Each phase is associated with its own modeling tool. The full description of the entire CWA framework and its five phases is beyond the scope of this thesis. Instead, the thesis will take an environmental approach and focus on the first phase of CWA, *work domain analysis*, which is used to define the environment that the activity is performed in. (Jenkins et al., 2008).

4.3.4 Work domain analysis (WDA)

As mentioned in the previous subsection, work domain analysis (WDA) defines the environment within which the activity is conducted and is done at a functional level, rather than at a behavioral level. WDA identifies a fundamental set of constraints on the actions of any system component (Stanton et al., 2018). The tool used to for WDA is the *abstraction hierarchy* (AH). It contains five levels of abstraction, ranging from the purpose of the system to its material form. The higher levels represent the work domain in its functional properties and the lower levels represent it in its physical properties (Vicente, 1999). Table 1 summarizes and describes the levels of abstraction.

Table 1. Description of the levels of abstraction in the AH (Adapted from Jenkins et al., 2008, p. 20).

Level	**Description**
Functional Purposes	The purposes of the work system and the external constraints on its operation.
Values and Priority Measures	The criteria that the work system uses for measuring its progress towards the functional purposes.
Purpose-Related Functions	The general functions of the work system that are necessary for achieving the functional purposes.
Object-Related Processes	The functional capabilities and limitations of physical objects in the

	work system that enable the purpose-related functions.
Physical Objects	The physical objects in the work system that afford the object related processes.

By identifying different aspects of the system and placing them in their respective levels, the AH creates a link from the purpose of the system to its capabilities (Jenkins et al., 2008). According to Rasmussen et al. (1994, p. 110), it provides a mental-model for reasoning and "maps the field or "territory" in which an actor (decision-maker) has to navigate in order to comply with their work requirements". Rasmussen et al. (1994) goes on stating that the abstraction hierarchy makes it possible to effectively explore different functions and properties of a system, which is highly important for any viable information system in a dynamic environment (Rasmussen et al., 1994). Naikar and Sanderson (2001) highlight the benefits of WDA by comparing it to other technical evaluation processes. Instead of focusing on individual physical devices, they argue that WDA promotes an understanding of how a separate set of components function and interact to fulfill different purposes (Naikar and Sanderson, 2001).

4.3.5 How WDA was used in this thesis

WDA has been used in different areas throughout research and literature. Ahlstrom (2005) performs WDA to find instances in air traffic control systems where weather information is lacking in displays and Effken et al. (2011) identifies design constraints in decision support tools for nurse managers. One shared aspect regarding the usage of WDA is that it is commonly utilized for describing complex sociotechnical systems. In this thesis, however, WDA was used as an evaluation tool rather than as a descriptive tool. This approach to WDA can in some regards be considered unconventional seeing that it does not reside in the works of Rasmussen et al. (1994) or Vicente (1999). Nonetheless, it is not entirely unprecedented. Naikar and Sanderson's (2001) demonstrate that WDA is a feasible approach for evaluating system design and in Carlson's (2018) thesis, it is said that "using a WDA as an evaluation tool is quite unique". While Jenkins el al. (2011) use WDA to evaluate an existing system, they argue that it can also be used to evaluate future changes in the system. Moreover, there is little to none research done in incorporating WDA with data warehouse systems, further establishing the scientific contribution of this thesis.

4.4 Data Collection Methods

There were two types of data used in this thesis. The first type consisted of views and ideas from different participants and was collected through interviews. This data was used to create an understanding of how the data warehouse system was to be designed and to provide the necessary material for performing WDA. The second type of data was the data of the ISY Case database. Without a data source that can be used to extract, transform and load data from, the data warehouse would serve no purpose. These two types of data collection methods are described in detail below.

4.4.1 Interviews

Interviews are a useful method for collecting data when human judgment is of interest and allows the interviewee to provide historical information while the interviewer can control the line of questioning (Creswell, 2009). The two main types of interviews in qualitative research are unstructured interview and semi-structured interview. Although they are both seen as flexible processes, they differ in some respects. In an *unstructured interview*, the interviewer has at the most a range of topics that are to be covered. The entire interview can be based on one single question that the interviewee is allowed to respond freely to, with the interviewer responding with follow-up questions on subjects that are deemed relevant. The character of an unstructured interview is highly informal and share similarities to a conversation.

A *semi-structured interview* is different in the sense that the interviewer has a series of questions or moderately specific topics, referred to as an *interview guide*. The guide serves as an outline of what should be covered in the interview, but the order they appear in can vary. The interviewee is allowed to answer freely and is encouraged to stray from the questions since it gives insight into what he or she regards as important and relevant (Bryman, 2008).

The preparing of an interview guide includes the following basic elements (Bryman, 2008):

- Define an order on the different topic areas so that the flow of questions is reasonable.
- Formulate questions or topics that help answer the research question.
- Use comprehensible and relevant language to the participating people and do not ask leading questions.
- Verify that "facesheet" information is recorded, such as name, age, position in the company, number of years employed, etc.

In this thesis, unstructured interviews were conducted with several employees at Norconsult Astando at an early stage of the research to discover information about the company, ISY Case and the data for designing the data warehouse. Later, two semi-structured interviews were done to perform WDA. The first one of these interviews was with an employee of Norconsult Astando in the purpose of gaining knowledge about the company's vision of the ISY Case data warehouse. The second interview was conducted with an analyst that uses Gatuarbete Webb. It was used to get an understanding of how analysis of lead-time data is performed and how it can be improved.

The interview guides constructed for this thesis followed Bryman's (2008) guidelines. They included one introductory phrase where the purpose of the interview was explained to the interviewee while allowing for an opportunity to ask questions. At the end of the interview, the interviewee got a chance to add additional comments. These measures were taken to guide the interviewee through the process in the best possible way (Gillham, 2005). Appendix A includes the two interview guides.

All the unstructured interviews and the first semi-structured interview lasted from forty minutes to one hour. They were all held at Norconsult Astando's office in Stockholm. The final interview was done at Trafikkontoret in Stockholm and lasted approximately one hour. Notes were taken for the unstructured interviews and the semi-structured interviews were recorded and later transcribed. Table 2 summarizes the interviews done in this thesis.

Table 2. Conducted interviews in this thesis.

Interview reference	**Participants**	**Job Title**	**Type**	**Date**	**Outcome**
Interview 1	Kave Silverklippa	CTO	Unstructured	24th of January 2019	General information about Norconsult Astando and ISY Case.
Interview 2	Göran Löfgren (via Skype)	Administrative Director of ISY Case	Unstructured	30th of January 2019	Information about the ISY Case database, how it works and its structure.
Interview 3	Patrik Backentoft & Magnus Jernberg	Project Leader & Sales Manager	Unstructured	11th of February 2019	The requirements of the data warehouse.
Interview 4	Magnus Jernberg	Sales Manager	Unstructured	11th of March 2019	Detailed information about ISY Case case flow and roles.
Interview 5	Patrik Backentoft	Project Leader	Semi-structured	21st of March 2019	Visions of a data warehouse system and its usage.

Interview 6	Lars-Gunnar Brundin	System Manager of Gatuarbete Webb at Trafikkontoret	Semi-structured	4th April 2019	How analysis is performed and how it can be improved.

4.4.2 Data structure

Each municipality that utilizes ISY Case has its own database where cases, both active and completed, are stored. It provides all the necessary information about a case such as its id, current status, description, location and so on. As was described in section 2.2, the database does not provide a way to easily view the lead-times for cases.

The database is a PostgreSQL relational database that consists of 79 different tables. However, many of these tables contain data about other modules, geographical data or information that is considered redundant for the scope of this thesis. The data used in the data warehouse was gathered from the following five tables and Figure 9 shows their relation:

- **Basecase:** Each case contains some fundamental information, no matter what module they belong to. This information is stored in the table *basecase* and includes an id of the case, its current status, a short description, when the case was created and when it was completed. The responsible administrator for the case is referenced by a foreign key, "ledningsägareid", in the basecase table.
- **Role:** The *role* table contains details about the responsible administrator, such as name, authority and billing information.
- **Schakt:** Information that is unique for schakt cases are stored in the table *schakt* which complements the basecase case data. Here, specific details such as the depth of the excavation, the type of work and entrepreneurs assigned to the case are stored. The private key "id" in the basecase table references to the private key "id" in the schakt table.
- **Platstyp:** The type of location that is assigned to the schakt case is stored in the table *platstyp* and linked by the foreign key "platstypid".
- **Event:** When a case changes status, it is logged in the *event* table. The table can be seen as containing the status history of a case through different events. Each event is described in a JSON string containing what type of change it is and a timestamp for when the change occurred. The event table is needed to calculate the lead-times of cases. "Eventid" links the table to the basecase table.

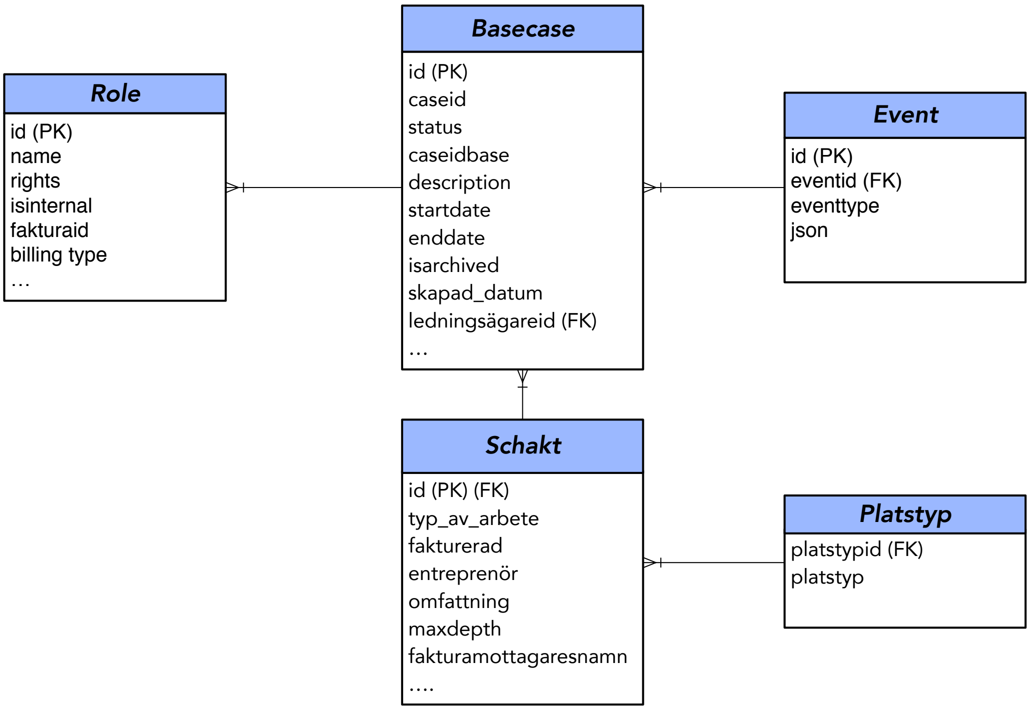


Figure 9. Schema of the ISY Case database tables for the schakt module used in this thesis.

The size of an ISY Case database depends on the municipality and how long the system has been in use. For the design of the data warehouse, data from one municipality in Stockholm was used. It is one of the larger databases available with a total size of 2 GB and has cases stored from 2015. In total, it contains 2004 cases whereas 651 of these are schakt cases. However, the choice of the dataset is of minor significance since the data warehouse design and ETL process are customized to the database structure rather than the quantity of the data.

4.5 Development Tools

4.5.1 Waterfall model

For the development of the data warehouse, the principles of the *waterfall method* were adopted. The method gets its name from the way a waterfall falls from one step to the next. It follows a particular sequence where one step has to be completed before the other. For each step, a systematic analysis is performed before proceeding to the next step (Han and Kamber, 2011).

The first step is to capture and define the requirements of the system to get an understanding of what the system should be able to do (Holcombe, 2008). These can be divided into *functional requirements*, i.e. what the system should do in terms of features and functionalities, and *non-functional requirements,* which guides and limits the operation of the system (Rainardi, 2008). The next step in the waterfall model is analyzing the said requirements and documenting them. Once the requirements of the system are clear, the system- and software design process starts. This step produces models and schemas of the system. Thereafter, the design is implemented and tested. Finally, the completed system is delivered. It is sometimes necessary to backtrack some of the steps, which is illustrated in Figure 10 (Holcombe, 2008).

Although the waterfall method is one of the commonly used approaches in software development and data warehouse design, it is not without criticism. Stober and Hansmann (2010) outline the issue of problems arising late in the development process. For instance, the user might see a prototype of the system during the testing step and realize that additional functionality needs to be added. In that case, the system developer has to traverse back to the requirements stage and add the new requirements while also updating the design and architecture. This lack of agility makes the waterfall approach less suitable for large projects where change is more likely to occur along the way. However, it works better in cases with a clearly defined project plan and requirements and when the scope of the project is limited (Stober and Hansmann, 2010). Acknowledging these criticisms, each step of the waterfall model was verified by the supervisor of this thesis to make sure that the development of the data warehouse was heading in the right direction. By doing so, the risk of problems arising late in the development process was minimized. The last step of the waterfall model, delivery and maintenance, was considered outside the scope of the thesis.

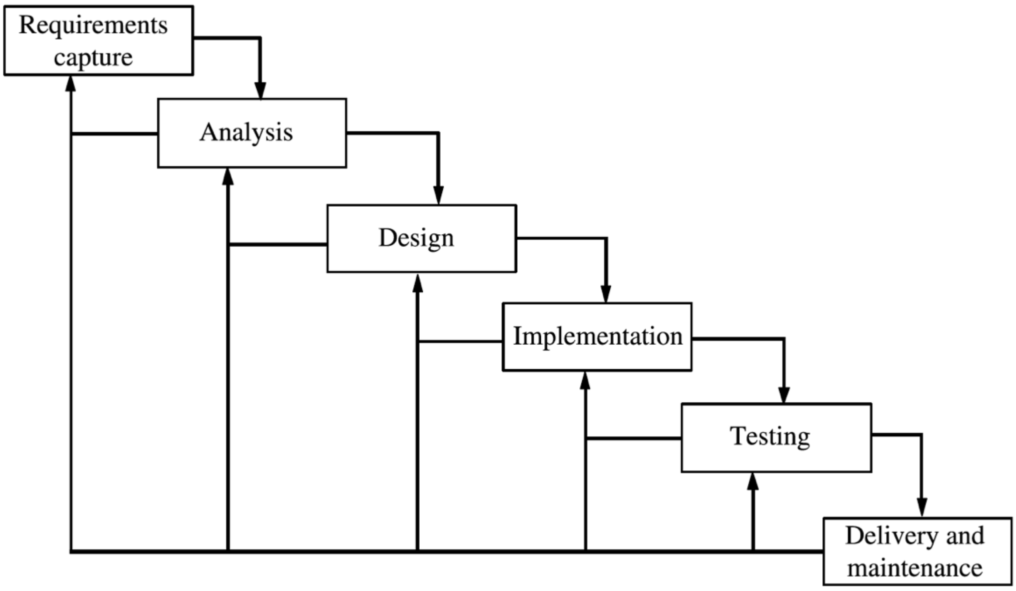


Figure 10. The waterfall model of software development (Holcombe, 2008, p. 4).

4.5.2 pgAdmin

pgAdmin is an open source management tool for PostgreSQL. It is one of the most popular tools used for administration and development (pgadmin, 2019) and Norconsult Astando uses it to manage the ISY Case database. In this thesis, pgAdmin version 4 was used for inspecting and gathering an understanding of the existing database as well as for sending queries to the data warehouse's fact and dimension tables.

4.5.3 Pentaho Data Integration (Kettle)

As the concept of data warehouse has grown more popular over the years, so has the number of available ETL-tools. For the scope of this thesis, the ETL-tool had to be accessible yet powerful and available on macOS (which was the operating system of the computer that the data warehouse was built in). With these aspects in consideration, Pentaho Data Integration, also known as Kettle, was chosen as the ETL-tool. Kettle is an open source Java-based ETL tool with a graphical user interface. It is one part of a larger collection of software applications known as Pentaho Business Intelligence Suite (Casters et al., 2010). The online documentation for Kettle (Pentaho, 2018) along with the Pentaho Kettle guidebook from Casters et al. (2010) was used to provide support for using the tool.

The building-blocks of Kettle are called *jobs* and *transformations*. A job dictates the process flow from a start point to an endpoint and usually contains transformations. A transformation manipulates the data by performing ETL tasks such as reading data from files, data cleaning, sorting rows or loading data into a data warehouse. The transformation consists of one or more *steps* that perform the task. Steps inside a transformation are connected by *hops*, which allows the data to flow one-way between steps (Casters et al., 2010). Figure 11 shows an example of a simple transformation.

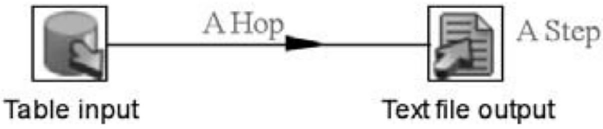


Figure 11. An example of a transformation in Kettle (Casters et al., 2010, p. 26).

5. ISY Case Data Warehouse Design Results

In this chapter, the process of designing the ISY Case data warehouse is described. The layout of the chapter follows that of the waterfall model: section 5.1 describes the requirements, followed by section 5.2 which gives an explanation of the different design choices made for the dimensional model. Thereafter, section 5.3 explains how the design was implemented through the ETL process and section 5.4 shows the testing of the data warehouse.

5.1 Requirements Capture and Analysis

In order to support decision-making, a data warehouse needs to fulfill certain requirements. One of the main requirements is that it must make the business's data accessible and understandable not only to the developers of the system but business users as well. The structure and labels of the data should be logical to the users so that they easily can combine analytic data in different combinations and have a minimal wait time. Moreover, Kimball and Ross (2013) argue that a data warehouse system must serve as a trustworthy and authoritative foundation for decision-making. This means that the data is relevant and that decisions are based on the analytic evidence presented (Kimball and Ross, 2013).

The choice of design of a data warehouse fully relies on its requirements. Through the conducted interviews with Norconsult Astando, a list of functional requirements was captured and analyzed. The functional requirements of the data warehouse were as follows (Jernberg and Backentoft, Interview 3, 2019):

1. Users should be able to analyze the lead-times for status Applied and status Complement of a schakt case.
2. Users should be able to analyze the number of applied schakt cases and the number of granted schakt cases under different time-periods.
3. Users should be able to analyze the total number of days a schakt case is in the Administrating Processing Phase for different responsible administrators.

5.2 Design

The design of the ISY Case data warehouse followed Kimball's bottom-up approach. As previously mentioned, this design principle is best fit for projects that have a well-defined and limited subject area. This approach was deemed suitable for the ISY Case data warehouse because the functional requirements were limited by only including cases of the schakt module. Hence, one data mart of the ISY Case schakt data was created from the ISY Case database and stored in the data warehouse, as seen in Figure 12. In the following subsection, this dimensional model will be described.

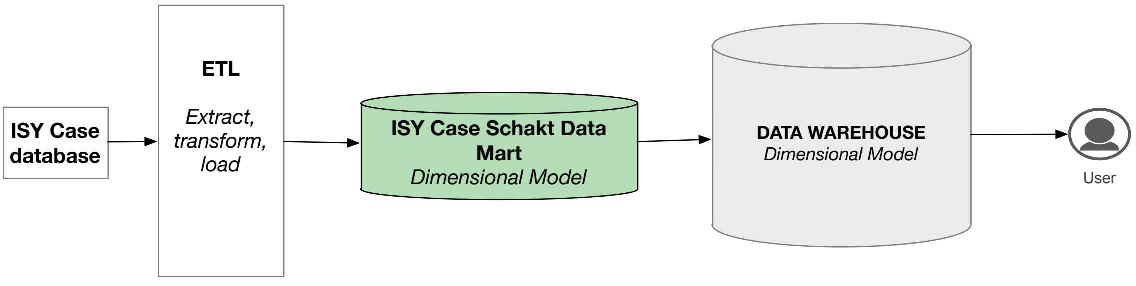


Figure 12. ISY Case data warehouse design.

5.2.1 ISY Case dimensional model

Figure 13 shows the star schema that was designed for the ISY Case schakt data mart. Before going through the fact and dimension tables, some design choices will be clarified. To begin with, the grain of the dimensional model is one case. This means that one row in the fact table corresponds to one case. Second, to maintain consistency between the source database and the data warehouse, the naming convention follows the ISY Case database. Hence, every status and some of the attribute names are in Swedish.

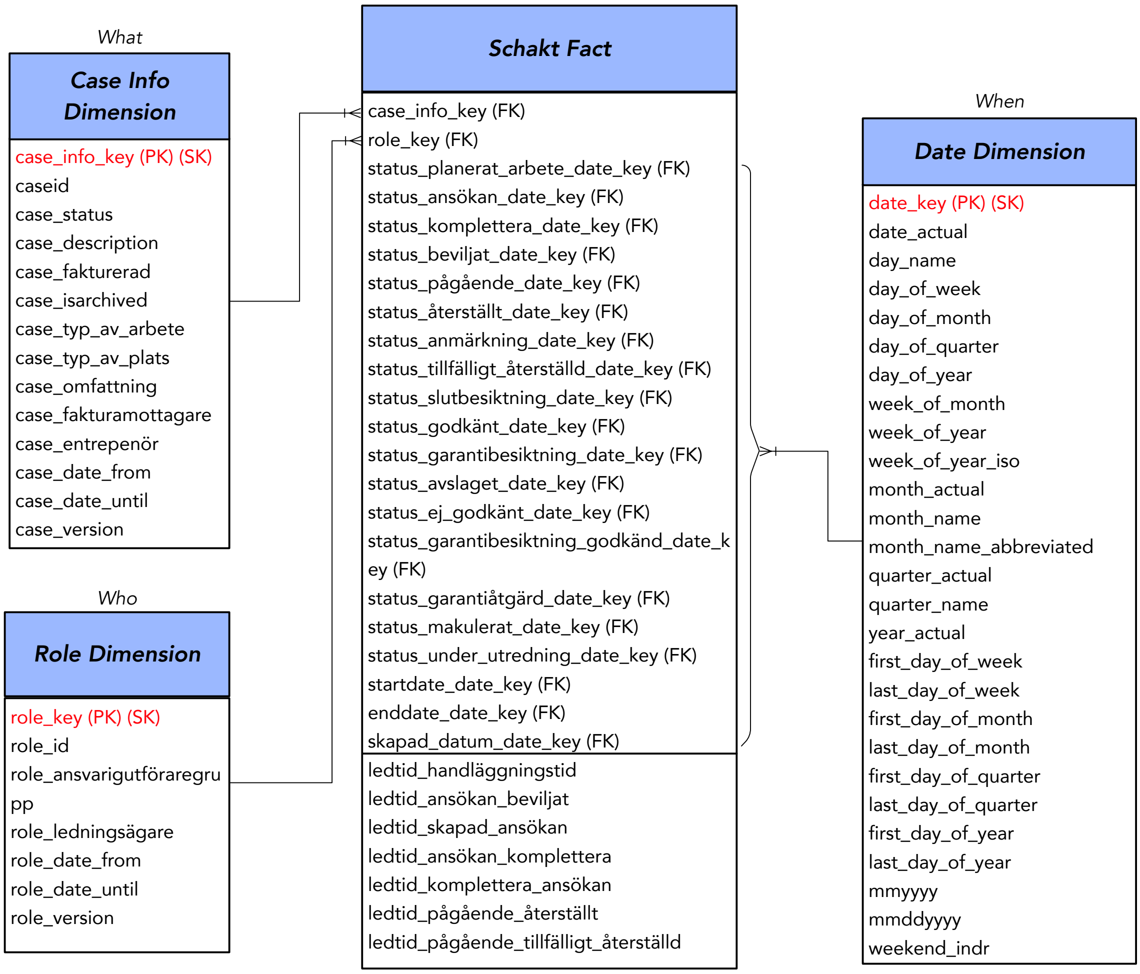
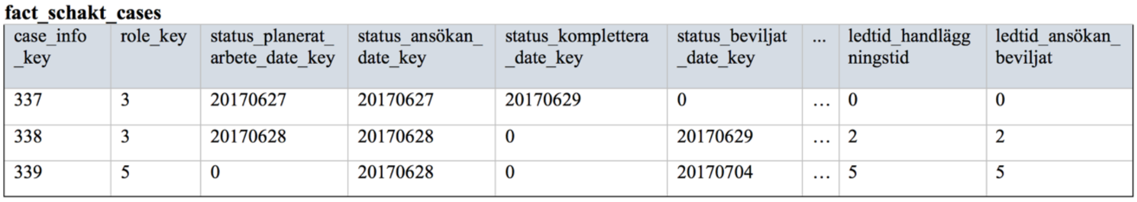


Figure 13. Star schema of ISY Case schakt data mart.

5.2.2 Schakt fact table

The schakt fact table was designed to be of type accumulating snapshot. As Adamson (2010) states, this type of fact table is a good choice of design when tracking time elapsed between one or several steps of a business process. Seeing that the process is the flow of a case as it goes through different statuses (lead-time), the accumulating snapshot was advisable. To successfully track the time in a status, the fact table incorporates a foreign key for each status. This means that the start date for each status is referenced by a foreign key to that date in the Date dimension table. The same principle is used for the date a case starts, ends and is created. The foreign keys “case_info_key” and “role_key” point to the dimension tables that provide information about the details of a case and those assigned to a case, respectively. The facts of the schakt fact table, i.e. the lower part of the fact table in Figure 13, are the lead-times necessary to meet the functional requirements of the data warehouse system as well as the lead-time between status Ongoing and Restored. They are calculated during the ETL process.

Adopting the accumulating snapshot design means that one row in the fact table contains a date for each status of a case, as seen in Figure 14. As a result, the operation of calculating a lead-time that is not stored as a fact can be done by measuring the time between two dates. For instance, finding the lead-time between Applied and Grant of a case can be obtained either directly through the fact “ledtid_ansökan_beviljat” or by calculating the days between the two statuses. If a different fact table design had been used, e.g. a transaction fact table, the same request would likely require combining several rows of a table or multiple tables.



fact_schakt_cases

case_info _key	role_key	status_planerat_ arbete_date_key	status_ansökan_ date_key	status_komplettera _date_key	status_beviljat _date_key	...	ledtid_handlägg ningstid	ledtid_ansökan_ beviljat
337	3	20170627	20170627	20170629	0	...	0	0
338	3	20170628	20170628	0	20170629	...	2	2
339	5	0	20170628	0	20170704	...	5	5

Figure 14. Three cases in the ISY Case fact table.

5.2.3 Handling the non-linear process of a case

An important detail to consider regarding the schakt fact table design is how it handles the non-linear process of a case, specifically for the statuses Applied and Complement. For instance, assume that a user wants to know the lead-time of Applied (requirement 1). Assume further that the Administrative Processing Phase of the case follows the process displayed in Figure 15. As can be seen, the answer to the user’s question is not as straight-forward as it first seems. It may be that the time spent in status Complement lasts several days longer than the time spent in status Applied. Thus, calculating the lead-time from the first status Applied to status Grant does not provide a justifiable answer since it does not convey this information of what happens in between. The design of the schakt fact table overcomes this matter by separating the lead-times of the non-linear processes and calculating the summarized number of days. As illustrated in Figure 15, the days a case spends in status Applied before receiving status Complement is defined in “ledtid_ansökan_komplettera” as the time between the two statuses, visualized by the red arrows, summarized. The same concept is used for calculating the time spent in status Complement, which is the green arrows in Figure 15. In addition, “ledtid_ansökan_beviljat” is the time spent in status Applied before receiving status Grant, visualized by the blue arrow. Thus, the user’s question about how many days are spent in status Applied is answered by adding “leddtid_ansökan_komplettera” and “ledtid_ansökan_beviljat” together.

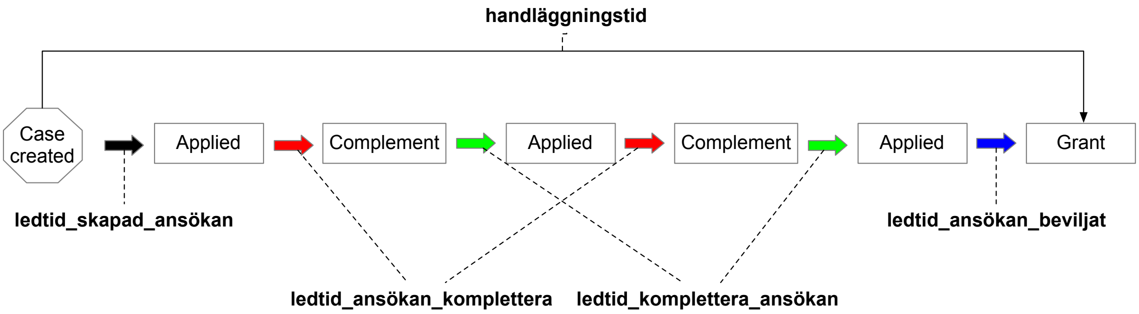


Figure 15. The non-linear Administrative Process Phase and its lead-times.

5.2.4 Dimension tables

Three dimension tables were designed to accompany the schakt fact table and provide details about the who, what and when of a case. Appendix B describes the columns of the Case Info dimension and the Role dimension. Each dimension table contains a primary key that references to a foreign key in the fact table. The primary key is generated by the system in the ETL process and is therefore a surrogate key. While the Case Info and Role dimensions consist of data extracted from the ISY Case database, the Date dimension is essentially a table containing all days from a specified date (in this instance 2015-01-01) up to the current date. The different kinds of attributes of the Date dimension table allow for the possibility to analyze cases based on a range of different date formats. For instance, a user can analyze cases that received status Applied at a certain date, day of the year, month, quarter, and so on.

5.2.5 Slowly changing dimensions

One of Inmon’s defining characteristics of a data warehouse is that it is nonvolatile, meaning that the data is static. In some scenarios, however, it is of analytical interest to be able to reflect changes in data in a data warehouse. For instance, assume that the ISY Case data warehouse loads source data once a week into the data warehouse. In the time between each ETL process, cases that are stored in the data warehouse undergo changes in the ISY Case source database. For example, the location of a case may be changed, or some other information may be updated. The way to handle these type of changes in a dimension table is referred to as *slowly changing dimensions* (Adamson, 2010).

There are a number of different methodologies to adopt when handling slowly changing dimensions. The Case Info dimension and the Role dimension were designed to use type 2 slowly changing dimensions. The idea of this approach is to add a new row in the dimension table every time a change has occurred (Vaisman and Zimányi, 2014). Table 3 shows how this is implemented in the Case Info dimension. In this example, the location of a case has been changed from sidewalk to road from when the data was last loaded into the data warehouse. The first two rows in Table 3 correspond to two different versions of the same case. The value 9999-12-31 in the "case_date_until" attribute, as well as the value 2 in "case_version", indicate that this is the most recent version. When the Case Info dimension is updated, the Schakt Fact table is also refreshed so that "case_info_key" references to the new value.

As a new record is inserted every time an attribute changes a value, the dimensions can grow considerably large. Not only does this increase the size of the data warehouse but it can also entail performance issues when querying join operations with the fact table (Vaisman and Zimányi, 2014). Slowly changing dimensions of type 2 is not used for the status attribute in the Case Info dimension, since the change of statuses happens frequently. Moreover, these changes of statuses are already recorded in the event table from the ISY Case database (as described in the previous chapter).

Table 3. Slowly changing dimensions type 2 in Case Info dimension.

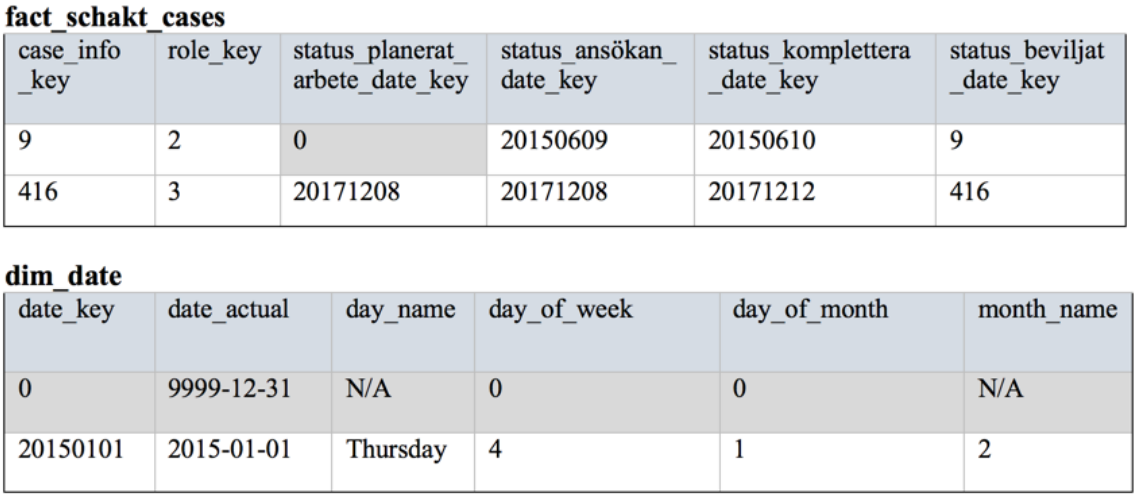
case_info_key	caseid	case_typ_av_plats	case_date_from	case_date_until	case_version
22	ST_20150709_45	Sidewalk	2005-01-01	2019-03-27	1
652	ST_20150709_45	Road	2019-03-27	9999-12-31	2
23	ST_20150715_51	Park	2005-01-01	9999-12-31	1
24	ST_20150715_52	Road	2005-01-01	9999-12-31	1

5.2.6 Handling null

A case will not always contain a value for every attribute in the ISY Case dimensional model. For example, a case might never obtain status Complement, some case info might be missing, or data might be invalid. These values are assigned a null value and their occurrence in a dimensional model can cause several problems when it comes to analyzing data. Although null values look like a blank or zero to the user, the database does not treat it as such (Thornthwaite, 2003). Kimball and Ross (2013) stress that having null in a fact table's foreign key column makes its reference to primary keys invalid while Adamson (2010) adds that null in dimension tables restrict the writing of queries to the data warehouse. Both authors state that null values in a data warehouse should be avoided altogether.

The handling of null values in the ISY Case dimensional model followed the guidelines from Becker (2010). For the foreign keys of the schakt fact table, the missing or invalid values are set to 0. This value references a special row in the corresponding dimension table that carries a non-null value for each column. Null values in the dimension tables are replaced by "N/A" if it is a string format, 0 if it is an integer format and a default date (9999-12-31) if it is a date format.

Figure 16 provides an example of how null is handled when analyzing cases. Suppose that a case has not yet obtained status Planned Work. "status_planerat_arbete_date_key" in the fact table will therefore be 0 and it will reference to a row in the date dimension table where all the attributes have some default data. When analyzing which day the case received status Planned Work, the user is met by "N/A" instead of null or blank values. Moreover, this design allows the user to issue queries that find all cases that have not received a specific status. This would not be possible if the foreign key of the status had a null value.



fact_schakt_cases

case_info_key	role_key	status_planerat_arbete_date_key	status_ansökan_date_key	status_komplettera_date_key	status_beviljat_date_key
9	2	0	20150609	20150610	9
416	3	20171208	20171208	20171212	416

dim_date

date_key	date_actual	day_name	day_of_week	day_of_month	month_name
0	9999-12-31	N/A	0	0	N/A
20150101	2015-01-01	Thursday	4	1	2

Figure 16. Handling of null in the ISY Case data warehouse.

5.3 Implementation

5.3.1 The ETL process

With these design choices established, the ETL process was configured in Kettle. Figure 17 shows the ETL process flow and consists of three jobs and four transformations. Starting from the top, it is segmented into three levels that handle the extraction, transformation, and loading. Each one of these phases includes an abort action that is called if an error occurs during the execution. An overview of what is done at each level is given in the subsections below.

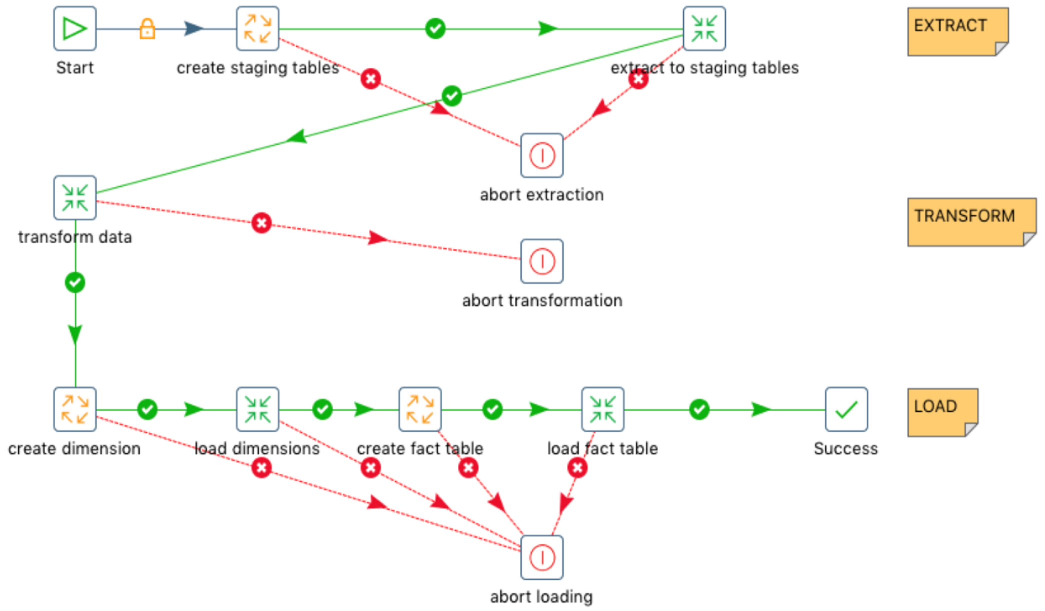


Figure 17. The ETL process for ISY Case data warehouse.

5.3.2 Extract

The extraction phase includes one job and one transformation. The purpose of the job is to create staging tables, i.e. temporary tables where the extracted data can be modified, and the transformation extracts data from the ISY Case database to the staging tables.

- The job, seen in Figure 18, first checks if any of the staging tables exist. If not, the staging tables “schakt_cases_with_status_hist”, “role_dw” and “denormalized_cases” are created. The table “roles” from the ISY Case database is extracted to “role_dw”.
- In the transformation, Figure 19, the ISY Case database tables “basecase”, “schakt” and “platstyp” are joined. The “event” table is thereafter joined using a left outer join operation. This results in a table where one row corresponds to one instance of a status of a schakt case. Thus, if a case has had multiple statuses, it has multiple rows in the staging table. This table is stored in “schakt_cases_with_status_hist”.

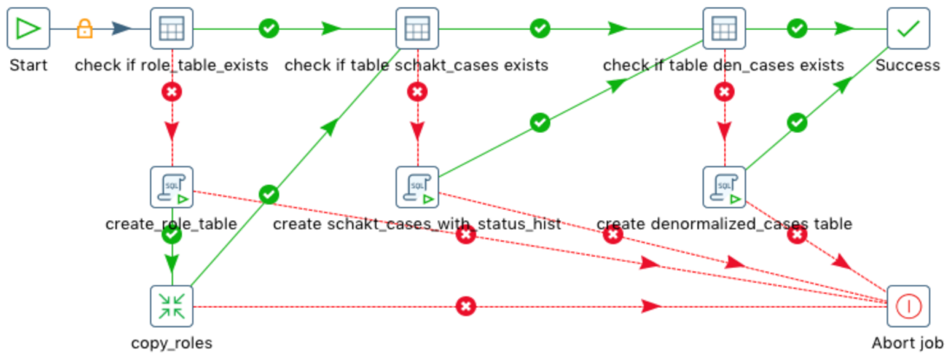

Start
check if role_table_exists
check if table schakt_cases exists
check if table den_cases exists
Success
create_role_table
create schakt_cases_with_status_hist
create denormalized_cases table
copy_roles
Abort job

Figure 18. The job for creating the staging tables.

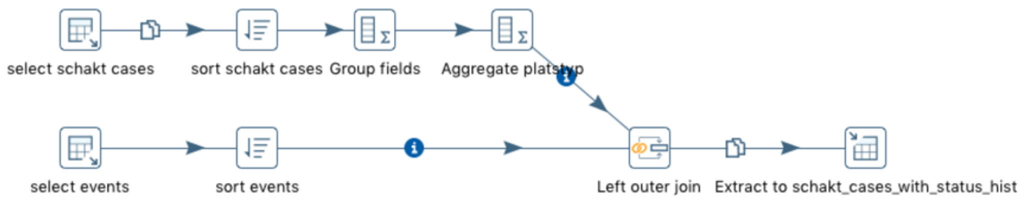


Figure 19. The transformation for extracting data to the staging tables.

5.3.3 Transform

The transformation phase consists of one transformation, shown in Figure 20, where data is cleaned, denormalized and modified.

- Data is cleaned to obtain the integration property**:**
  - String attributes are cut to remove redundancy, e.g. “status.godkänt” is transformed to “godkänt”.
  - String attributes are edited to be consistent uppercase/lowercase.
  - Quotation marks and other special characters are removed.
  - Null is replaced with “N/A”.
- The staging table is denormalized so that one row corresponds to one case with timestamps for each status. If a case has multiple Applied or Complement statuses, one column is created for each instance of the status.
- Dates that are in string format are formatted to date format.
- Lead-times are calculated.
- The transformed data is extracted to “denormalized_cases”.

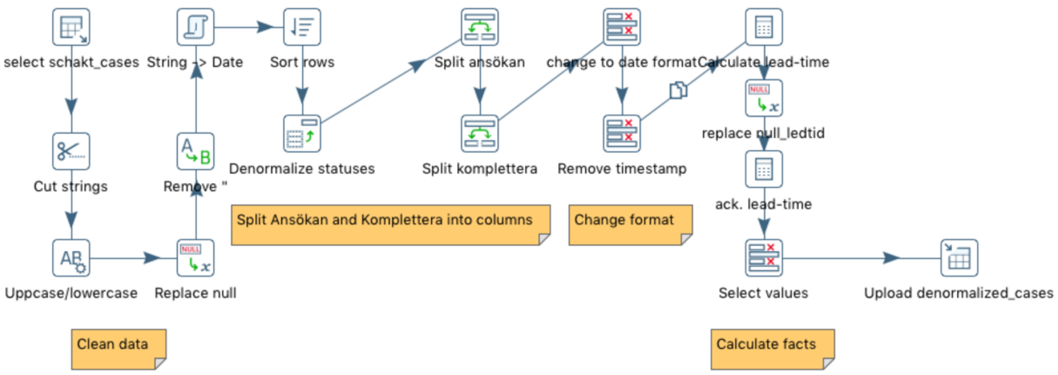


Figure 20. The transformation for transforming the data.

5.3.4 Load

The loading phase consists of two jobs and two transformations. The jobs create the dimension tables and the fact table while the transformations loads the data into the tables and structures it in the dimensional model.

- Dimension tables and fact table are created using the same procedure from the extraction phase, as seen in Figure 21 and Figure 22.
- Dimension tables are loaded with the transformed data by adopting slowly changing dimensions of type 2, shown in Figure 23
- The fact table is loaded with facts and foreign keys. If a foreign key is null, it is replaced with 0. If “lead_time” is null, it is replaced with 0. If a fact table already exists, it compares the two versions and updates the values where a change has been recorded. The transformation in Figure 24 shows this process.

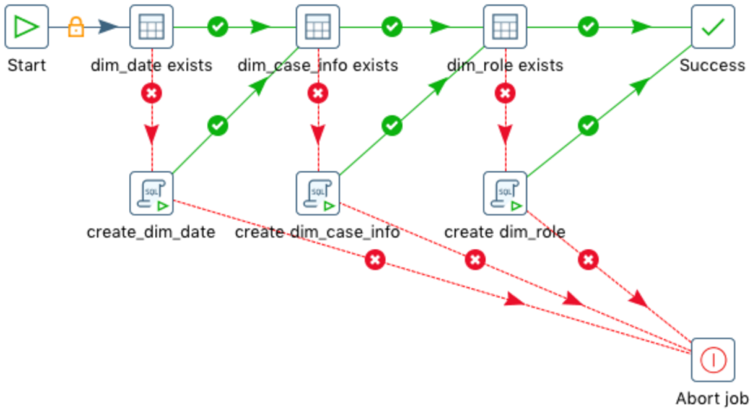


Figure 21. The job for creating the dimensions.

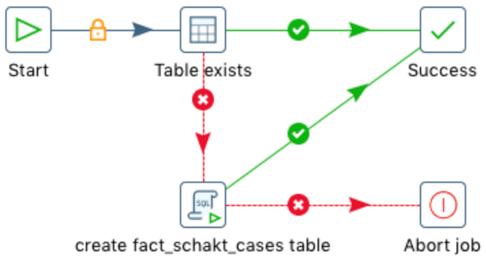


Figure 22. The job for creating the fact table.

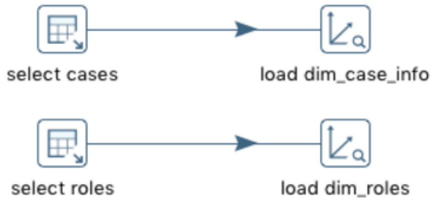


Figure 23. The transformation for loading the dimension tables.

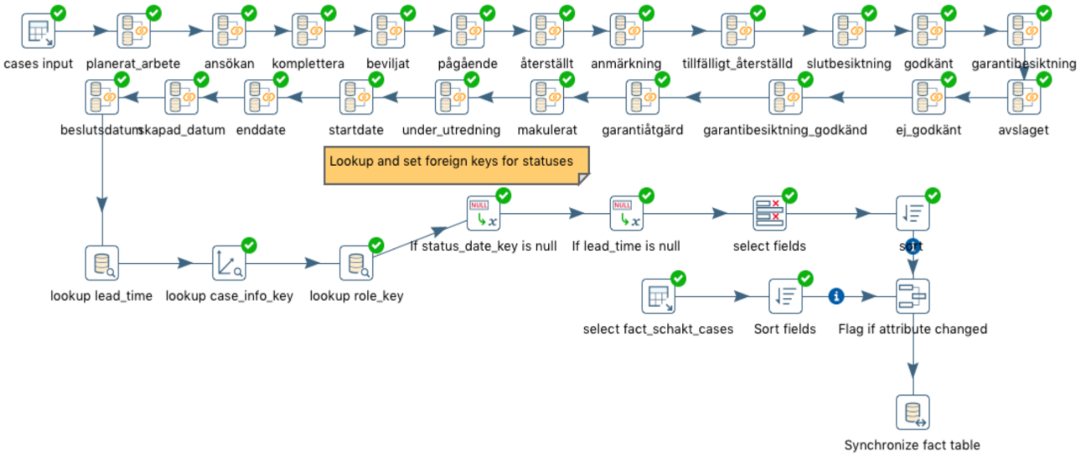


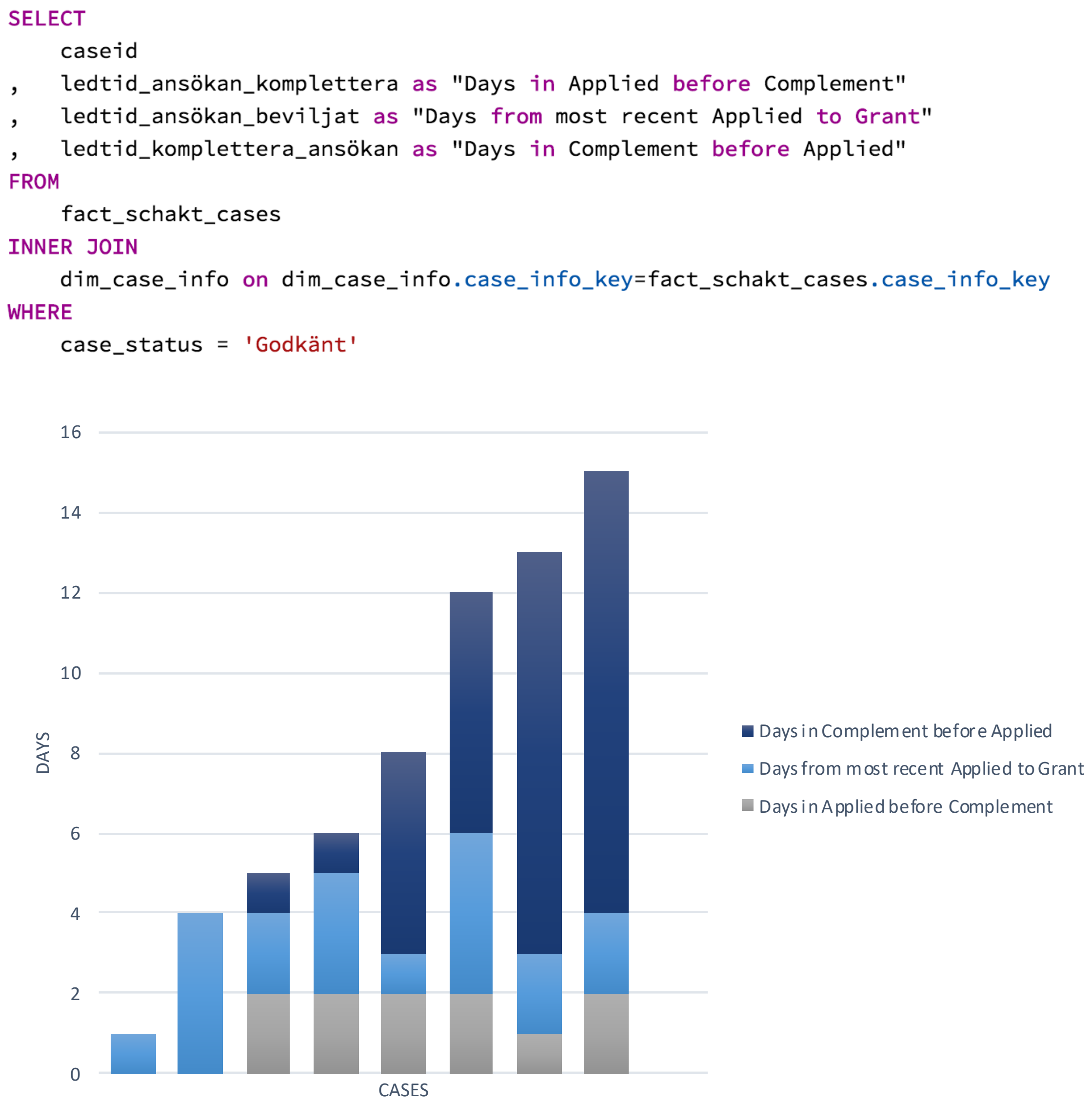
Figure 24. The transformation for loading the fact table.

5.4 Testing

There are several ways for a user to interact with a data warehouse system (as seen in Figure 5). In the purpose of testing the data warehouse after it had been constructed, SQL queries were used to produce a result that was later visualized in Microsoft Excel. The following subsections show how each of the three requirements of the data warehouse system was met.

5.4.1 Requirement 1: Time in status Applied and Complement

Requirement 1 of the ISY Case data warehouse was to analyze the time a case spends in status Applied and status Complement. This requirement was met by sending the SQL statements that is seen in the upper part of Figure 25. The query selects the relevant facts from the Schakt Fact table that have status Approved (i.e. completed cases) and joins it with the Case Info dimension to retrieve the id of cases. The lower part of Figure 25 shows a visualization of the result.



```
SELECT
    caseid
,   ledtid_ansökan_komplettera as "Days in Applied before Complement"
,   ledtid_ansökan_beviljat as "Days from most recent Applied to Grant"
,   ledtid_komplettera_ansökan as "Days in Complement before Applied"
FROM
    fact_schakt_cases
INNER JOIN
    dim_case_info on dim_case_info.case_info_key=fact_schakt_cases.case_info_key
WHERE
    case_status = 'Godkänt'
```

Figure 25. Testing and visualizing requirement 1.

5.4.2 Requirement 2: Applied and granted cases over time

In Figure 26, an approach to meet the demands of requirement 2 is shown. To view the number of applied cases over a period of time, the Date dimension table was used. The SQL statement selects the month (both by name name and number) as well as the cumulative number of cases with status Applied. Considering that municipalities can grant cases from past months, cumulative values were used. The result was joined with the Date dimension on the date key for status Applied to filter all cases that were applied for in the year 2017. The visualization of the result in Figure 26 shows applied cases and granted cases.

```
SELECT
    dim_date.month_actual,
    dim_date.month_name as "Month",
    sum(count(status_ansökan_date_key)) OVER (ORDER BY dim_date.month_actual) as "Applied"
FROM
    fact_schakt_cases
INNER JOIN
    dim_date on fact_schakt_cases.status_ansökan_date_key=dim_date.date_key
WHERE
    dim_date.year_actual = '2017'
GROUP BY
    dim_date.month_actual,
    dim_date.month_name
```

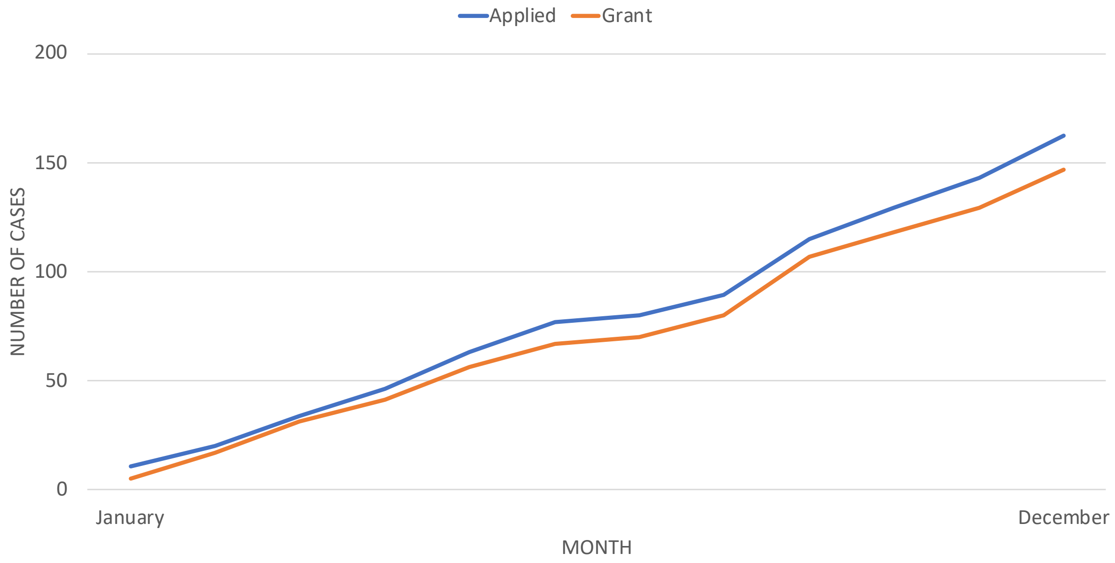


Figure 26. Testing and visualizing requirement 2.

5.4.3 Requirement 3: Time in the Administrative Processing Phase

Requirement 3 of the ISY Case data warehouse was to analyze how many days a case was in the Administrating Processing Phase for different responsible administrators. The chart in Figure 27 shows the result for one anonymous responsible administrator. For this SQL statement, all three dimension tables were used to filter approved cases (Case Info dimension) for one responsible administrator (Role dimension) during the fourth quarter of 2016 (Date dimension).

```
SELECT
    ledtid_handläggningstid
,   caseid
FROM
    fact_schakt_cases
INNER JOIN
    dim_role on fact_schakt_cases.role_key = dim_role.role_key
INNER JOIN
    dim_case_info on fact_schakt_cases.case_info_key = dim_case_info.case_info_key
INNER JOIN
    dim_date on fact_schakt_cases.status_ansökan_date_key = dim_date.date_key
WHERE
    ledningsägare = '[redacted]'
and dim_date.year_actual = '2016'
and dim_date.quarter_actual = 4
and dim_case_info.case_status = 'Godkänt'
```

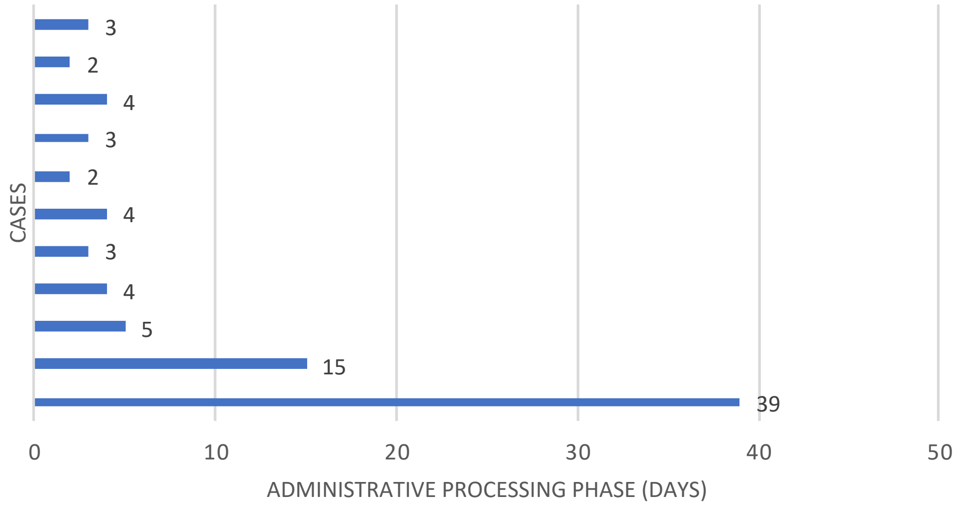


Figure 27. Testing and visualizing requirement 3.

6. Interview Results

In this chapter, the data captured to perform WDA is presented. It describes Norconsult Astando's vision of the data warehouse system and compares it to how analysis is done by an analyst. It is based on the information gathered from interviews with Lars-Gunnar Brundin, System Manager of Gatuarbete Webb at Trafikkontoret and Patrik Backentoft, Project Leader at Norconsult Astando. Section 6.1 shows how analysis is usually performed and section 6.2 describes why analysis is needed. The types of questions that the analysis should answer are explained in section 6.3 and section 6.4 lists the needs and requirements for analyzing cases.

6.1 The Work of an Analyst

6.1.1 Brundin, interview 6

As the System Manager of Gatuarbete Webb, one of Lars-Gunnar Brundin's tasks is to perform analysis by generating documents such as reports and statistics. Currently, this is done by selecting the relevant cases in Gatuarbete Webb and thereafter extracting them to Excel (or a similar program). In Excel, Brundin is able to perform calculations and operations on the data in order to aggregate it and compile reports. The actual requests for the documents are made by those in managerial positions, mostly managers and supervisors, who use it for decision-making and running their businesses. Generally, the people requesting analysis do not have knowledge in handling the data presentation tools. Thus, they are required to work together with an analyst, in this case Brundin, who knows the data. The process of generating reports usually starts by finding the relevant data and making a basic analysis that meets the requirements. Thereafter, together with the manager or supervisor that made the request, the analyst explores how to present it in a clear and pleasing way in graphs, tables, charts or other types of visualization.

6.1.2 Backentoft, interview 5

Patrik Backentoft, Project Leader at Norconsult Astando, envisions that the user of the ISY Case data warehouse system is someone who has both business knowledge in formulating an analytical request while also possessing skills in some data presentation tool. Backentoft is aware that this is a visionary and perhaps unrealistic scenario and that the knowledge and skills required are often split between different people, like in Brundin's case. Still, it is crucial to have the ability to formulate an analytical question. Backentoft draws parallels to how Hans Rosling, the co-founder of the Gapminder Foundation, used statistics and data to view data in a new and exciting way. Although Rosling had powerful tools to his disposal, his way of formulating the questions was just as important.

6.2 Motive for Performing Analysis

6.2.1 Brundin, interview 6

As explained throughout this thesis, the managers and supervisors who make the analytical requests do so in order to receive support in their decision-making process. By having reports and statistics, they get an overview of the processes and see where changes can be made in order to become more effective in their administrative work. This is something that is apparent from Brundin's point of view when he performs analysis.

6.2.2 Backentoft, interview 5

Although Norconsult Astando perceives the ISY Case data warehouse as a system that provides support in decision-making, it should also have other areas of usage. Backentoft highlights how the system could be utilized to provide information demanded by the media, the public or the government that reflects some current event, situation or condition of society. For instance, a municipality may be requested by the media to provide information about the work of a responsible administrator after a scandal about the company has been discovered. These situations are often time-sensitive, and it is therefore important for the municipality to meet the demands as quickly as possible.

Norconsult Astando's idea is that the data warehouse system will be offered to ISY Case users as a separate system, allowing them to provide one system for handling cases and another system for analyzing its data. Not only will this approach generate a greater profit from each customer, but it will also establish a longer-lasting relationship with them. This form of business organization, i.e. that one company controls several stages of a supply chain, is defined as vertical integration ("Vertical integration," 2011). Further, the enabling of analysis can be perceived as a way to strengthen the ISY Case brand and gain competitiveness against Norconsult Astando competitors. Backentoft ultimately hopes that by offering a system that satisfies the needs and wishes of the customers, Norconsult Astando will distinguish from its competitors and get a stronger positioning.

6.3 Types of Requests

6.3.1 Brundin, interview 6

Although the requests that the analyst receives often include analysis of the lead-times and statuses, they are rarely identical to each other. One request may be to find the total of approved applications while another is to find how many applications that were made by a certain entrepreneur during a specified time period in a small area. Regardless of the complexity of the question, it often happens that it gets adjusted along the analytical process. Moreover, the person who makes the request does not always know what to expect from the output. Thus, when the first result of the analysis is shown, the request can be altered by adding or removing some criteria in order to see how the result changes. This is by no means out of the ordinary. Brundin describes that there have been several scenarios where he has delivered an answer to an analytical question, but once he has presented it, he is requested to do the same analysis on a different area or another time period in the interest of comparing the different results.

6.3.2 Backentoft, interview 5

The idea that each request is unique is something that Norconsult Astando is aware of when envisioning how their data warehouse system should function. In order to assist its user in every possible way, the data warehouse system should not be limited to answering a set of requests. Rather, it should have the ability to answer a broad variety of analytical questions. It is therefore favorable to include a high verity of data in the data warehouse and it should include data rather than exclude it, Backentoft argues. In doing so, the data warehouse will be able to support a broad scope of analytical capabilities and satisfy requirements and demands without having to adjust the design. It is then up to the user to decide which data should be used.

6.4 Improvements and Requirements for Analysis

6.4.1 Brundin, interview 6

When asked how the process of analyzing cases could be improved, Brundin mentions four aspects. The first is about the software tool and its flexibility. As described previously, the current method includes selecting data from Gatuarbete Webb and then transferring it to Excel where calculations are done to compile reports. Having one external tool that handles both of these processes would make the analysis less complex to perform. It would eliminate the need for transferring the data to Excel and allow the analyst to use one single tool for both reading the data and performing analytical operations.

The second aspect of improvement is how the data is visualized. Brundin describes that he sometimes can feel overwhelmed by the massive amount of data that is shown in Excel when doing analysis. The rows upon rows of information in numerous columns divided between different sheets make it difficult to get a general overview of the data. Not only does this entail difficulties when operating on the data, but it also makes it hard to present it in a pleasing way when generating reports and statistics. If the data would be visualized in a more comprehensible way, it would be easier to interpret the data when performing analysis and be of better support when making decisions.

A consequence of the high amount of data is a reduction in performance, which is the third area of improvement. Brundin recalls an occasion when he was making an exhaustive statistical report with several advanced aggregations and suddenly had to change one minor calculation, which took Excel fifteen minutes to recalculate. Higher performance of the system when performing calculations would make the analysis less time-consuming for the analyst.

Finally, there is a possible improvement to be made in assuring that the data has a certain quality to it. Part of the responsibility of the analyst is to know the data and make sure that it is correct. Brundin exemplifies this by describing that a few years ago, the standard was to set the start date of a case to the same date as when the case entered the system. However, when entrepreneurs began to issue applications via e-mail, a decision was made to set the start date of a case as the date it was granted. This caused a lot of confusion when analyzing data, as some cases had a missing start date and others had an incorrect one. As an analyst, it is vital to know these details about the data since it constitutes if the analysis is correct or not. If the data is guaranteed to be of high quality and correct, it would make the analysis more accurate.

6.4.2 Backentoft, interview 5

Norconsult Astando also considers the performance of the data warehouse as one of its most important aspects. Without having the ability to do advanced operations that give immediate results, the data warehouse would be trivial. Backentoft expresses it as having the ability to perform analysis ad-hoc and that it is especially important in situations where there are requests that are difficult to foresee and prepare for in advance. The data warehouse system should then aid the user in generating this information in a quick manner.

Similar to Brundin, Backentoft also mentions data quality as one of the most vital requirements of a data warehouse. He argues that the decision-makers must have accurate and correct data to their disposal since it affects what decisions are made. Backentoft also promotes data hygiene. By this, he implies that the data should be consistent and applicable. For instance, the lead-times of cases should be in a useful format, i.e. days instead of minutes, and the formats should be the same for all cases. It is the job of the data warehouse rather than the analyst to tweak and convert the data so that it is easy to comprehend, hence making the analysis quicker to perform.

7. Analysis

Section 7.1 in the analysis chapter uses the results from the interviews to perform WDA. Each subsection describes one level of the AH in the order they were developed in. It starts from the bottom two levels (7.1.1 and 7.1.2), then skipping to the upper two levels (7.1.3 and 7.1.4), and finally ending at the middle-level (7.1.5).

7.1 Work Domain Analysis

WDA was conducted to define the ISY Case data warehouse system environment, and an AH was used to map the system's functions and properties. The analysis is based on an analyst's experience and desired requirements and Norconsult Astando's vision of the data warehouse, described in the previous chapter. Each level of the AH includes a function that describes *what* should be done. Between the levels, the functions are connected through mean-ends links. The function's link to the lower level describes *how* the function should be done and the link to the upper level describes *why* it should be done (Rasmussen et al., 1994). Figure 28 shows the AH.

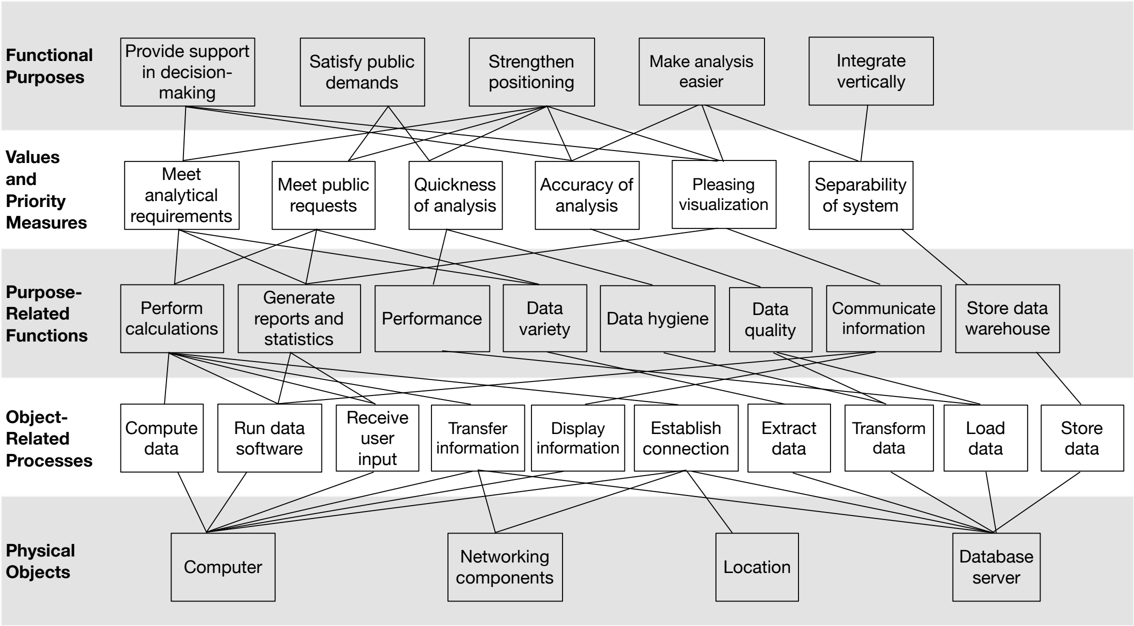


Figure 28. Abstraction hierarchy of data warehouse system.

7.1.1 Physical objects

The base of the AH shows the physical objects that make up the data warehouse system. This level represents all the external and internal components that are required to analyze cases. For instance, the computer, which is a trivial component used by the analyst, consists of internal hardware (including processor, memory, storage) as well as external hardware (keyboard, mouse, screen). The database server is assumed to store both the ISY Case database as well as the ISY Case data warehouse. Network components such as a router allow the computer to connect to the database server, which may only be accessible in certain areas, as symbolized by the location component in the AH.

7.1.2 Object-related processes

How the physical objects can or should be used is described in the second level of the AH, object-related processes. The processes are explained in generic terms in order to avoid constraining their possibilities (Stanton et al., 2018). For instance, explaining the process as "compute data" instead of "calculate case data" indicates that the computer can have other purposes and functions unrelated to case data.

Following the mean-ends links downwards shows which physical objects are required to perform the processes. For example, running data software only requires a computer, but establishing a connection involves having a computer that can connect to a database server while being in a certain location and connected to networking components. In the AH, the database server is assumed to handle the ETL process. Thus, the processes of extracting, transforming and loading data are linked to the database server.

7.1.3 Functional purposes

At the highest level of the abstraction hierarchy are the functional purposes, which represent the purpose of the system and ultimately the reason it exists (Stanton et al., 2018). The results from the interviews showed that the purpose of the ISY Case data warehouse system has different meanings depending on how it is perceived. From an analysis perspective, the purpose is to provide support in decision-making processes and to generate statistics and figures based on the demands from media, the public, and the government. These purposes can be captured as "Provide support in decision-making" and "Satisfy public demands". Further, the interview results showed that improvements could be made to make analysis easier to perform. In the AH this purpose captured as "Make analysis easier". From a business perspective, Norconsult Astando's perceives the data warehouse system as way to gain competitiveness and to broaden the scope of available systems to their customers. The AH captures these purposes as "Strengthen positioning" and "Integrate vertically".

7.1.4 Values and priority measures

The level underneath the purpose-related functions consists of values and priority measures that signify progress towards achieving the functional purposes. They are all in some way expressed in subjective or objective measurements. For example, the measurement "Quickness of analysis" is a measurement of the time it takes to perform an analysis while "Pleasing visualization" depends on the user's own perception of how the data is presented.

Studying the upwards links in the AH shows that "Provide support in decision-making" is linked to three measurements: how well the analytical requirements are met, how well the results are visualized to the decision-maker and the accuracy of the analysis, i.e. how correct the analysis is. No connection between the accuracy of the analysis and "Satisfy public demands" were possible to make from the results. Instead, the results showed that "Satisfy public demands" values how quick the analysis can be performed as well as how well the public requests are met. The results further showed that accuracy was regarded as a way to make analysis easier for inexperienced analysts, thus the link to "Make analysis easier". How the data is presented, captured as "Pleasing visualization" and having a separable tool for performing analysis, captured as "Separability of system", were also values that were considered to simplify the analysis. Further, since all the measures mentioned make the data warehouse system more powerful in some way, they are all ways to strengthen Norconsult Astando's positioning. The same conclusion cannot be drawn for having a separable system, however, because the results from the interviews indicated that it was done to integrate vertically and make analysis easier.

7.1.5 Purpose-related functions

The middle-level of the hierarchy consists of the purpose-related functions. It describes the functions required by the ISY Case data warehouse to achieve its functional purposes. By viewing the relation to the object-related processes, some functions require several processes. For instance, if a user wants to calculate lead-times of cases, the user must first input the request via the data presentation software. The data is transferred through the established connection and computed.

The function captured as "Communicate information" in the AH can perhaps be trivial, but necessary it is to include this function in the AH to study which processes are required for the user to view information. The desired functions of "Performance", "Data variety", "Data hygiene" and "Data quality" can be seen as characteristics of a general data warehouse. Hence, they are enabled by the different stages of the ETL process. High performance of operations is an effect of structuring the data in a suitable data warehouse design. For example, if the requirement is to analyze sales, the transaction fact table structure is advisable because it promotes high performance for these requirements. The structuring of data is done during the loading stage of the ETL process, thus the means-end link to "Loading". The second function that is enabled by ETL, "Data variety", is linked to the extraction process since it determines what data is transferred from the source systems to the data warehouse. "Data hygiene" is made possible in the transformation process when data is cleaned and made non-volatile. Finally, "Data quality" was defined in the results as having authentic data in the data warehouse. This can be linked to both the transformation process, where data errors are removed, as well as the loading process where actions are made to refresh the data and make it up to date.

Studying the links upwards shows that both "Perform calculations" and "Generate reports and statistics" are necessary to meet the analytical requirements and public demands. "Generating reports and statistics" is also done to produce a visual presentation of the result. “Performance” is promoted to speed up the analysis. The same conclusion can be drawn from “Data hygiene” on the basis that good data hygiene eliminates the need for the analyst to alter and correct the data, which can be time-consuming. Having data of good quality provides a more accurate analysis. Finally, if the data is highly varied, it allows for the system to meet a broader scope of analytical requests.

8. Discussion

This chapter discusses the design choices made for the ISY Case data warehouse, section 8.1, and evaluates its design through the AH model that was presented in the preceding chapter, section 8.2

8.1 Design Choices for the ISY Case Data Warehouse

The ISY Case data warehouse constructed in this thesis fulfills Inmon's four data warehouse characteristics. First, it has a subject-orientation of schakt case's lead-times. Second, the transformation process makes the data integrated by replacing null values and removing inconstancies in status names and date formats. Third, it contains data that is static. Finally, slowly changing dimensions allow the system to store historical data. Whereas studying lead-times was considered to be a difficult task to perform in the operational database of ISY, the star schema with the accumulating snapshot approach provides a convenient way to both capture and analyze them to meet Norconsult Astando's requirements.

Kimball's bottom-up design approach suits the ISY Case data warehouse in the purpose of analyzing cases of the schakt module. The design also allows for future development of the data warehouse through the construction of additional data marts of other ISY Case modules. For example, suppose that one wishes to enable analysis of the traffic arrangement module. This could be achieved by designing a secondary star schema, similar to the one designed in this thesis. As a result of having a different case flow, the fact table would have to contain different attributes that correspond to the statuses of the module and the Case Info dimension would have to be altered. However, the general star schema design would be the same for the two modules.

By adding additional data marts for each module, the ISY Case data warehouse would gradually gain a higher variety of analytical possibilities and become a more powerful system. However, it could be argued that the simplicity of the data warehouse would eventually be lost if the number of data marts and supported subject-areas becomes too great. This would notably be the case if the data warehouse is to include data from several of Norconsult Astando's systems other than ISY Case. A more suitable approach would be to follow Inmon's top-down approach. This would entail modeling the ISY Case source data to third normal form in a data warehouse, and from this extract data marts based on what subject-area is to be analyzed.

8.2 Evaluating the ISY Case Data Warehouse Through WDA

By following the means-end links from the functional purposes down the hierarchy in the AH, it is possible to see what functions are required from the ISY Case data warehouse design to achieve all of its purposes. In the following subsections, the ISY Case data warehouse system design is evaluated through each functional purpose.

8.2.1 Provide support in decision-making

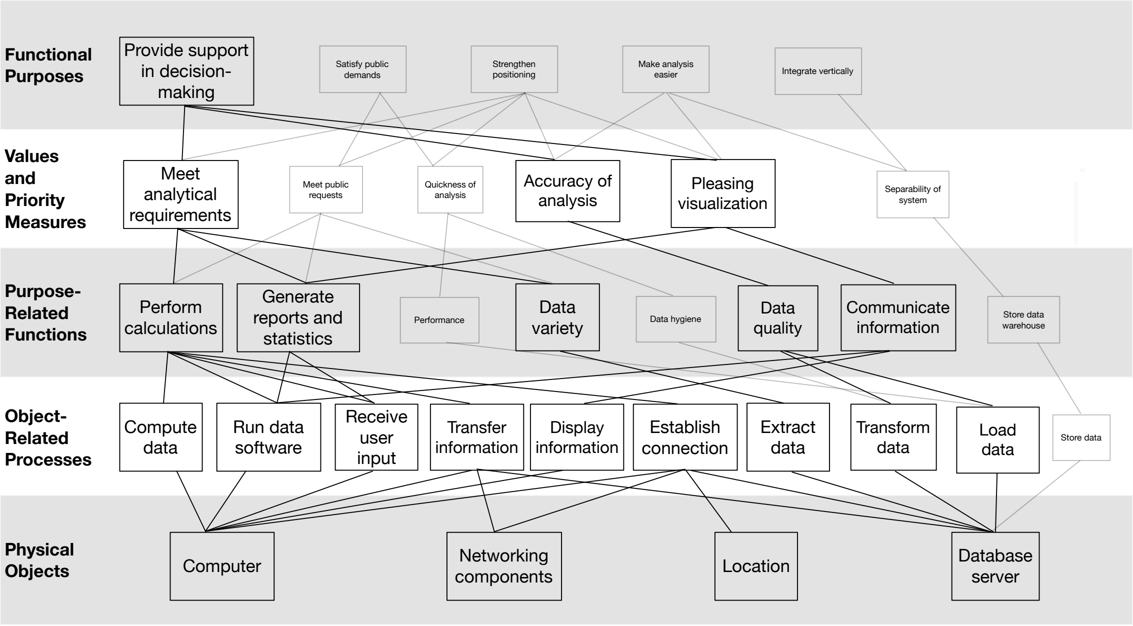


Figure 29. AH highlighting means-end links from "Provide support in decision-making".

In order for the data warehouse design to provide support in decision-making, it needs to fulfill five purpose-related functions: “Perform calculation”, “Generate report and statistics, “Data variety”, “Data quality” and “Communicate information”. As seen in Chapter 5.4, where the requirements were tested, the data warehouse design supports a variety of different aggregation operations which can be presented in graphs, charts or other types of visualizations. The purpose-related functions of performing calculations, generating reports and communicating information are thus achieved by the design. Further, how pleasing the data is visualized depends on what data presentation tool is used to interact with the data warehouse. It can be assumed, however, that the data warehouse assists in providing pleasing visualization.

Whether or not the purpose-related function of “Data variety” is fulfilled by the design is not as clear. On one hand, the ISY Case data mart contains all the necessary data to enable lead-time analysis. By capturing the date of each status of case in the fact table, it is possible to find the elapsed time between any two statuses and the associated dimension tables allows for analysis based on case details, authorities, or dates. On the other hand, the design limits the analytical possibilities since the user only has access to the data of the fact- and dimension tables. If an analytical request requires data that is stored in the ISY Case database and not in the data warehouse, for instance detailed geographical information, a higher data variety would be required. Figure 29 shows that “Data variety” is achieved during the object-related process “Extraction”. Thus, by altering what data is extracted from the ISY Case source database, the function of data variety would be obtained. This could be done by either adding additional dimension tables that contain more detailed data or adopting Inmon’s top-down approach to model, which captures the entire source data in the data warehouse. Both solutions would enable the function of data variety and aid in the progress towards providing support in decision-making by meeting the analytical requirements.

Figure 29 shows that “Data quality” improves the accuracy of the analysis and is gained during the transformation and loading processes. There are several ways that the design helps in making the data authentic. For one, having the missing foreign keys referencing a special row in the dimension tables conveys that every key has a reference. This means that the analyst is never met by an empty value and the likelihood for confusion is reduced. The design choice of using type 2 changing dimensions can also be viewed as a way of achieving data quality. Whenever errors are corrected or data is updated in the ISY Case database, it is recorded in the data warehouse. However, the design does not provide a way of indicating the grade of authenticity of the data. There is nothing hindering the entrepreneurs using ISY Case to enter faulty data that is later loaded into the data warehouse. Thus, the scenario explained by Brundin (Interview 6, 2019) where start dates of cases were missing can still transpire when analyzing cases in the ISY Case data warehouse.

8.2.2 Satisfy public demands

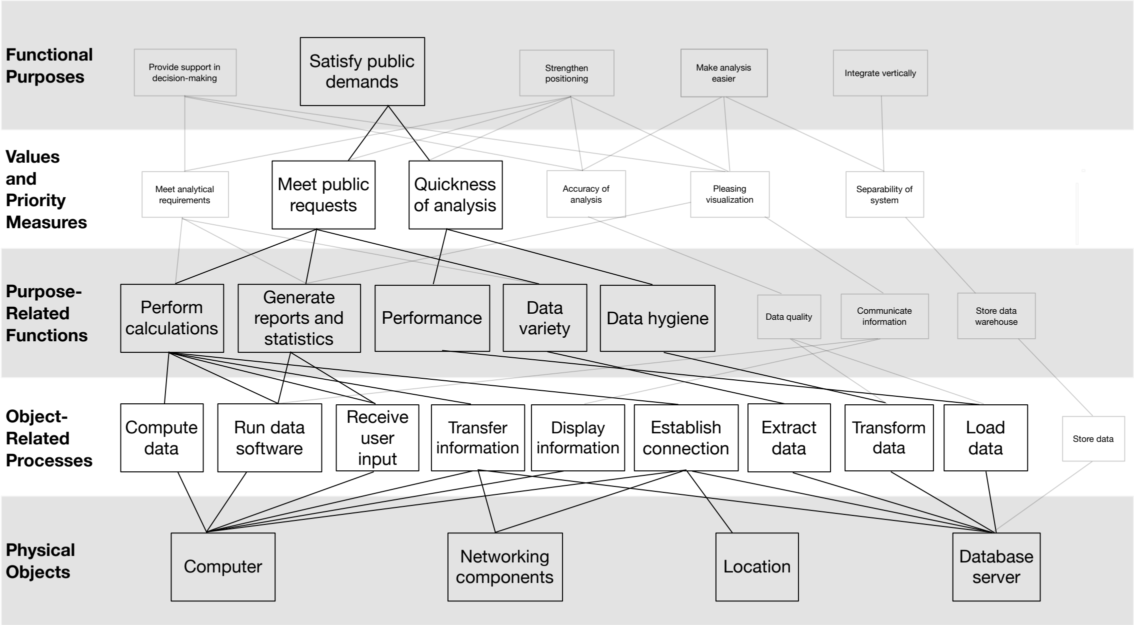


Figure 30. AH highlighting means-end links from "Satisfy public demands".

In addition to the requirements mentioned in the preceding subsection, the progress towards “Satisfying public demands” is measured by how fast the analysis is performed, “Quickness of analysis”. Following the links downward from this measurement shows that the functions required are “Performance” and “Data hygiene”. As seen in Figure 30, “Performance” is obtained when data is loaded into the warehouse and structured in a data warehouse design. For the ISY Case data warehouse, the accumulating snapshot fact table design is advantageous since the operations needed to calculate lead-times are rather simple, making them likely to be performed quickly. However, as no comparisons were made between different star schema design and fact table structures, it cannot be concluded that the proposed design is the most favorable in terms of performance. As for the functionality of data hygiene, it is achieved during the transformation process. The cleaning of ISY Case data that occurs during this process, as well as replacing null values, are actions that improve data hygiene. Thereby, the analysis can be performed quicker and the progress towards satisfying public demands is improved.

8.2.3 Strengthen positioning

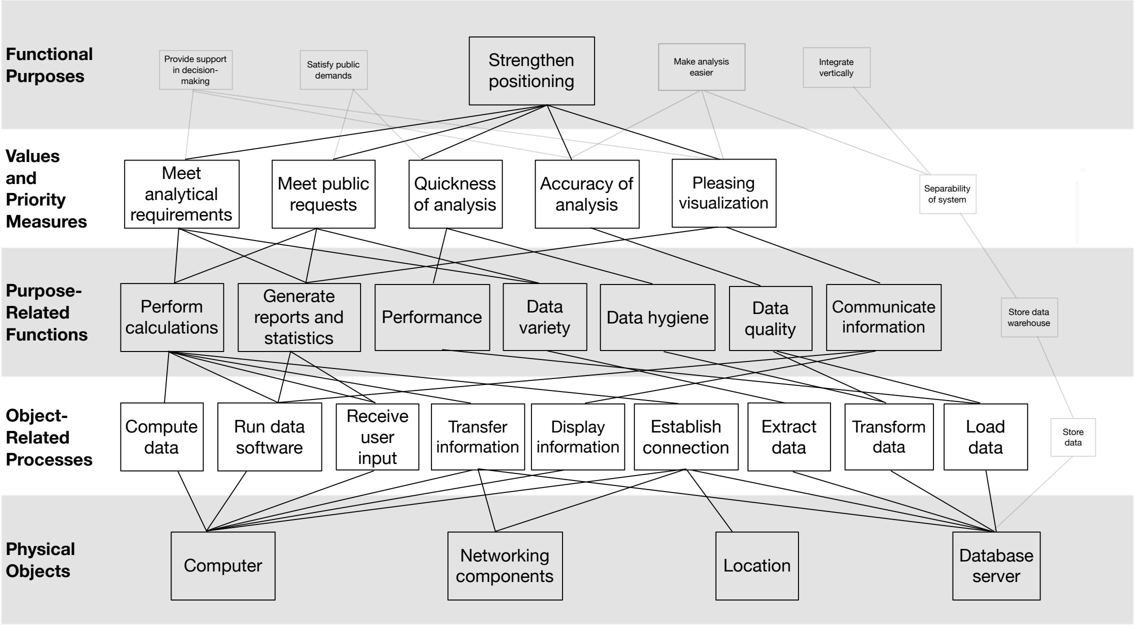


Figure 31. AH highlighting means-end links from "Strengthen positioning".

In Figure 31 it is shown that the means-end links to “Strengthening positioning” includes all the functions explained in the preceding subsections. The ISY Case data warehouse strengthen Norconsult Astando’s positioning by being able to perform quick analysis, present the results in pleasing visualization and having a certain accuracy to the analysis. However, how well the system meets the analytical requirements and public demands are aspects that can be improved by including data that is highly variated.

8.2.4 Make analysis easier

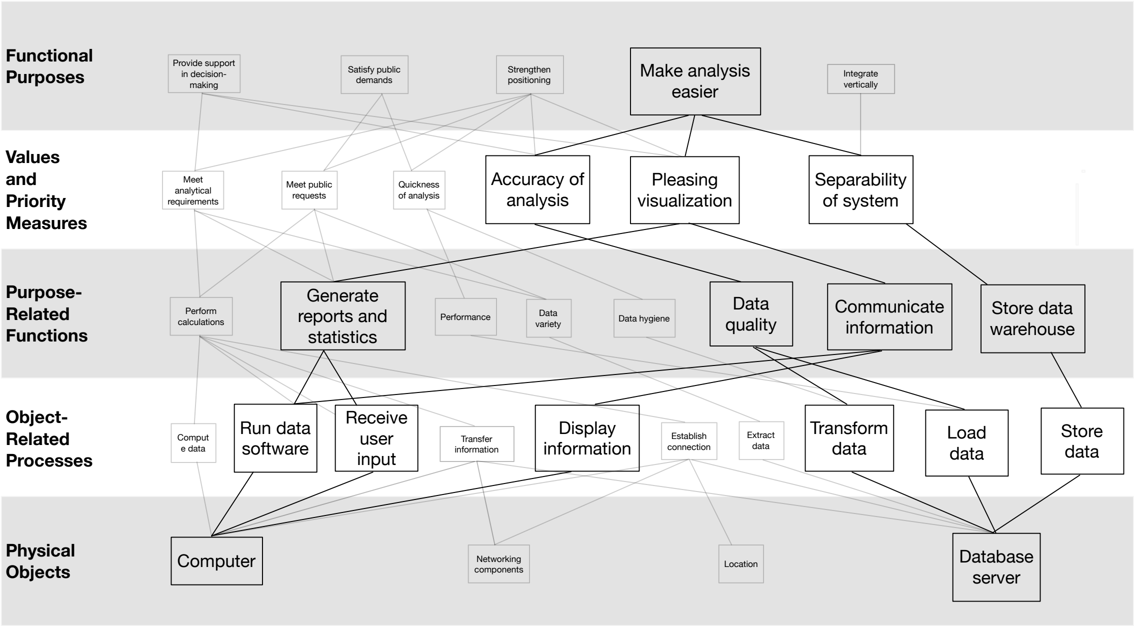


Figure 32. AH highlighting means-end links from "Make analysis easier".

For reducing the difficulty in performing analysis, the data warehouse design should provide high accuracy of analysis, have a pleasing visualization and provide a separable system, see Figure 32. As mentioned in the previous subsections, the design provides data quality in some aspects, which in turn improves the accuracy. Second, the data can be visualized in different pleasant ways depending on what data presentation tool is used. Finally, on account of that any data warehouse system works as a separate data store that is detached from the operational database, the functionality of storing the data warehouse is obtained. By having the ISY Case data warehouse system on a database server that is accessible to the analyst, the analysis is separated from the every-day handling of cases. Altogether, the design choices for the ISY Case data warehouse make analysis easier to perform.

8.2.5 Integrate vertically

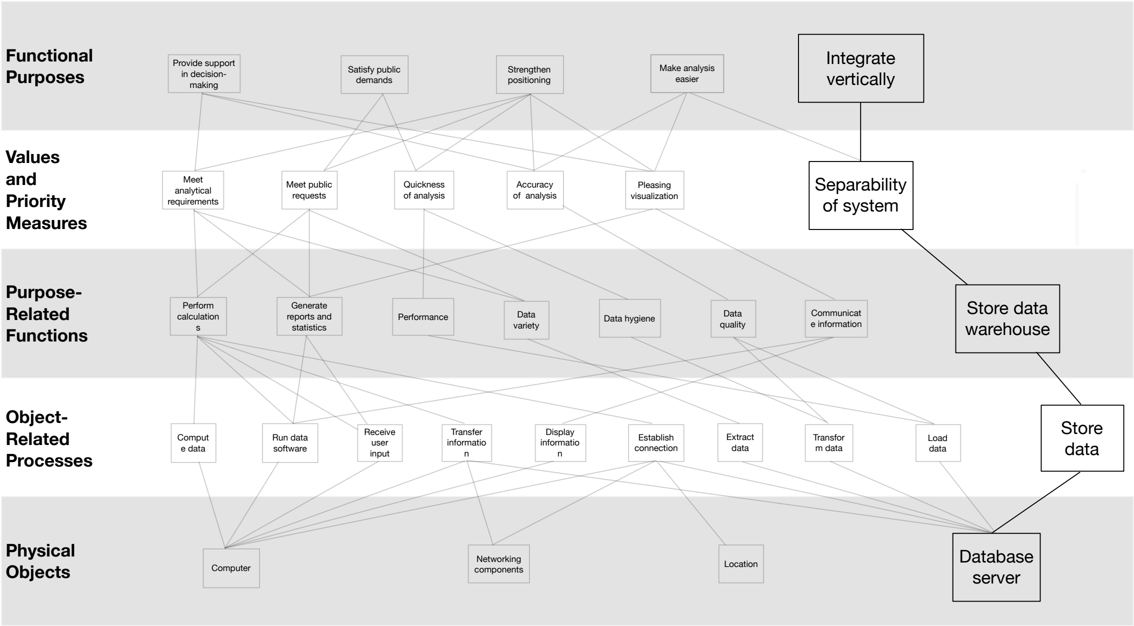


Figure 33. AH highlighting means-end links from "Integrate vertically".

The means-end link in Figure 33 shows that the progress towards integrating vertically is measured by the separability of the ISY Case data warehouse system. As has been stated previously, the data warehouse is a fully separable system. Hence, the ISY Case data warehouse allows for Norconsult Astando to integrate vertically.

9. Conclusion

Section 9.1 and 9.2 in this chapter outline the conclusions drawn from this thesis by answering the research questions presented in the introduction. The final section, 9.3, provides suggestions of how the research can be furthered explored.

9.1 What Data Warehouse Design Can Meet Norconsult Astando's Requirements for Enabling Analysis of ISY Case?

In enabling analysis of ISY Case, a data warehouse design adopting Kimball's bottom-up approach and a dimensional modeled data mart in a star schema was shown to meet the desired requirements from Norconsult Astando. Although the scope of the thesis included cases of only one module, the same design principles can be used for other ISY Case modules to create multiple star schemas. This would enable analysis of the entire ISY Case system. The simplicity of the design allows for a data warehouse that is quick to construct and easy to use. However, for broadening the analytical possibilities to several of Norconsult Astando's systems, and given enough time and resources, Inmon's top-down design would be better suited.

9.2 To What Extent Can WDA Be Used to Evaluate the Design of the ISY Case Data Warehouse System?

Most of the design choices for ISY Case data warehouse were captured in the AH. The measures of speed of analysis, accuracy of analysis, presentation of data, and separability of system were evaluated as being aided by the design to some degree and provided progress towards each of the system's functional purposes. On the other hand, how well the demands from the public and the analytical requirements are met were measurements that had an unfavorable effect on some of the functional purposes and could be improved by altering the design.

This thesis showed that WDA is a powerful evaluation tool for measuring how well a data warehouse design achieves its needs and purposes. However, the success of using WDA for evaluation is limited by what is captured in the AH model. It is therefore necessary to have sufficient data collection methods to perform WDA properly and to fully capture the functions of each level in the AH. With this, however, WDA can be used for evaluating a data warehouse system to a large extent.

9.3 Suggestions for Future Work

The research done in this thesis showed how a data warehouse system can be designed to enable lead-time analysis and how WDA can be utilized to evaluate its design. There are several opportunities for how the research can continue. For instance, it would be interesting to compare different alternatives for design of the ISY Case data warehouse to study how the functions presented in the AH differ. Another opportunity could explore how the WDA can be employed on similar data warehouse systems to evaluate their performance. This would entail further development of the AH and lead to a model suitable for evaluating a wider range of data warehouse systems.

10. References

10.1 Printed References

Adamson, Christopher (2010), Star Schema: The Complete Reference. Mc-Graw Hill Osborne Media.

Ahlstrom, Ulf (2005), Work Domain Analysis for Air Traffic Controller Weather Displays. Journal of Safety Research, 36(2): 159–169.

Berg, Bruce L (2001), Qualitative Research Methods for the Social Sciences. Boston: Allyn and Bacon.

Blatter, J (2008), Case Study, 69–71, in: The SAGE Encyclopedia of Qualitative Research Methods. Thousand Oaks, CA: SAGE Publications, Inc.

Bryman, Alan (2008), Social Research Methods. Oxford: Oxford University Press.

Carlson, Sara (2018), Using Work Domain Analysis to Model the Impact of Digitalization in Intensive Care. Uppsala universitet, Institutionen för informationsteknologi.

Casters, Matt, Bouman, Roland and Dongen, Jos van (2010), Pentaho® Kettle Solutions: Building Open Source ETL Solutions with Pentaho Data Integration. Sybex: John Wiley & Sons.

Creswell, John W (2009), Research Design: Qualitative, Quantitative, and Mixed Methods Approaches. Sage.

Effken, Judith A., Brewer, Barbara B., Logue, Melanie D., Gephart, Sheila M. and Verran, Joyce A. (2011), Using Cognitive Work Analysis to Fit Decision Support Tools to Nurse Managers' Work Flow. International Journal of Medical Informatics, 80(10): 698–707.

Elmasri, Ramez and Navathe, Sham (2016), Fundamentals of Database Systems. Hoboken, NJ: Pearson.

Fidel, Raya and Pejtersen, Annelise Mark (2004), From Information Behavior Research to the Design of Information Systems: The Cognitive Work Analysis Framework. Information Research: An International Electronic Journal, 10(1): 210.

Gillham, B. (2005), Forskningsintervjun: Tekniker Och Genomförande. Malmö: Holmbergs.

Golfarelli, Matteo and Rizzi, Stefano (2009), Data Warehouse Design: Modern Principles and Methodologies. New Delhi: McGraw-Hill Education.

Han, Jiawei and Kamber, Micheline (2011), Data Mining: Concepts and Techniques. Burlington, MA.: Elsevier.

Holcombe, Mike (2008), Running an Agile Software Development Project. Hoboken, N.J.: John Wiley & Sons, Incorporated.

Inmon, W H (2002), Building the Data Warehouse. New York, NY: Wiley.

Jenkins, Daniel P., Stanton, Neville A., Salmon, Paul M. and Walker, Guy H. (2011), Using Work Domain Analysis to Evaluate the Impact of Technological Change on the Performance of Complex Socio-Technical Systems. Theoretical Issues in Ergonomics Science, 12(1): 1–14.

Jenkins, Daniel P, Walker, Guy H. and Stanton, Neville A (2008), Cognitive Work Analysis: Coping with Complexity. England; Burlington, VT: Ashgate Publishing

Jukic, Nenad (2006), Modeling Strategies and Alternatives for Data Warehousing Projects. Communications of the ACM, 49(4): 83–88.

Jernberg, Magnus (2019), ISY Case Schakt & TA [PowerPoint presentation]. Norconsult Astando. Stockholm

Kimball, Ralph and Ross, Margy (2013), The Data Warehouse Toolkit: The Definitive Guide to Dimensional Modeling. Indianapolis, Indiana: John Wiley & Sons.

Naikar, Neelam and Sanderson, Penelope M. (2001), Evaluating Design Proposals for Complex Systems with Work Domain Analysis. Human Factors: The Journal of the Human Factors and Ergonomics Society, 43(4): 529–542.

Norconsult Astando (2017), ISY Case - Underhåll Med Koll På Läget. Stockholm

Rainardi, Vincent (2008), Building a Data Warehouse with Examples in SQL Server. Apress.

Rainer, A, Runeson, P, Host, M and Regnell, B (2012), Case Study Research in Software Engineering : Guidelines and Examples. Hoboken: John Wiley & Sons, Incorporated.

Rasmussen, Jens, Mark Pejtersen, Annelise and Goodstein, L. P. (1994), Cognitive Systems Engineering. New York: Wiley.

Schneider, M (2008), A General Model for the Design of Data Warehouses. International Journal of Production Economics, 112(1): 309–325.

Stanton, Neville A., Salmon, Paul M., Walker, Guy and Jenkins, Daniel P. (2018), Cognitive Work Analysis: Applications, Extensions and Future Directions. Boca Raton: CRC Press, Taylor & Francis Group.

Stober, Thomas and Hansmann, Uwe (2010), Agile Software Development: Best Practices for Large Software Development Projects. Heidelberg [Germany] ; New York: Springer.

Vicente, Kim J. (1999), Cognitive Work Analysis: Toward Safe, Productive, and Healthy Computer-Based Work. Mahwah, N.J: Lawrence Erlbaum Associates.

10.2 Webb References

Barone, Jennifer (2007), Scientist of the Year Notable: Hans Rosling. Discover Magazine. December 6. available at http://discovermagazine.com/2007/dec/hans-rosling [26 April 2019].

Becker, Bob (2010), Design Tip #128 Selecting Default Values for Nulls. Kimball Group. available at https://www.kimballgroup.com/2010/10/design-tip-128-selecting-default-values-for-nulls/ [1 April 2019].

Cambridge Dictionary (2019), Data Analysis. Cambridge Dictionary. available at https://dictionary.cambridge.org/dictionary/english/data-analysis [29 April 2019].

Norconsult Astando (2019), Om Oss - Om Norconsult Astando - Norconsult Astando. Norconsult Astando. available at https://www.norconsultastando.se/om-oss/om-norconsult-astando/ [3 April 2019].

Oracle (2017), A Relational Database Overview. Oracle. available at https://docs.oracle.com/javase/tutorial/jdbc/overview/database.html [15 February 2019].

Pentaho (2018), Pentaho Data Integration. Pentaho Documentation. available at https://help.pentaho.com/Documentation/8.2/Products/Data_Integration [2 April 2019].

pgadmin (2019), PgAdmin 4 — PgAdmin 4 4.3 Documentation. pgAdmin 4. available at https://www.pgadmin.org/docs/pgadmin4/4.x/ [26 March 2019].

Race, R (2008), Literature Review, 488–489, in: The SAGE Encyclopedia of Qualitative Research Methods. 2455 Teller Road, Thousand Oaks California 91320 United States: SAGE Publications, Inc. available at http://sk.sagepub.com/reference/research/n249.xml [2 April 2019].

Thornthwaite, Warren (2003), Design Tip #43: Dealing With Nulls In The Dimensional Model. Kimball Group. available at https://www.kimballgroup.com/2003/02/design-tip-43-dealing-with-nulls-in-the-dimensional-model/ [1 April 2019].

Vaisman, Alejandro and Zimányi, Esteban (2014), Data Warehouse Systems. Berlin, Heidelberg: Springer Berlin Heidelberg. available at http://link.springer.com/10.1007/978-3-642-54655-6 [22 January 2019].

Vertical Integration. (2011) Encyclopædia Britannica. available at https://www.britannica.com/topic/vertical-integration [17 April 2019].

Appendix A

Interview guide for interview 5

Bakgrund

1) Berätta lite om din roll på Norconsult Astando.
    a. På vilket sätt är du drivande i detta projekt?

Vision

2) Vad skulle du säga är Norconsult Astandos tanke med systemet?
3) När väcktes intresset för att skapa ett sådant här system?
    a. Vem är initiativtagare?
4) Varför ska användaren använda detta system?
    a. Vad tror ni användaren hoppas att få ut?
    b. Vad hoppas ni att informationen kan användas till?

Användaren

5) Vilka ser ni som användarna av systemet?
6) Har du upplevt att det finns ett behov hos användaren av ett sådant här system?
7) Hur tror du att det kommer att tas emot?
8) Tror du att det finns en tröskel för att använda system?

Användning

9) I vilka situationer ska systemet användas?
    a. Speciella tillfällen?
    b. Regelbundet?
10) Vad hoppas Norconsult Astando få ut av att detta system används?
11) Existerar det något liknande system idag?
    a. Om ja, vad?
    b. Om nej, hur går man tillväga för att få ut samma information?
12) Ser ni några svårigheter eller nackdelar med att ha ett sådant system?
13) Hur kan ni garantera att det användas på rätt sätt?
14) Tror du att liknande system kan utvecklas för era andra tjänster?

Avslut

15) Har du något som du vill tillägga?

Interview guide for interview 6

Bakgrund

1) Berätta lite om dig själv och din roll på Trafikkontoret.
   a. Vad är dina uppgifter?
   b. Hur länge har du jobbat?

Om arbetet inom ärendehantering

2) Hur ser handläggningsprocessen ut?
   a. Vad är era uppgifter?
3) Hur upplever du ledtiderna/handläggningsprocessen?
   a. Upplever du att det finns någon problematik?

Gatuarbete Webb

4) Berätta om Gatudrift
   a. Vad är syftet?
   b. Vilka använder det?
   c. Hur länge har det funnits?
5) Hur ser ditt arbete med Gatudrift ut?
6) Hur skiljer det sig från ISY Case schakt?

Nuvarande analysering

7) Vilka typer av analyser gör ni på data från Gatudrift?
   a. Från vem kommer kravet?
   b. Er själva?
8) Hur går du tillväga för att göra analyser?
   a. Vem gör det?
   b. Hur lång tid tar processen?
   c. Hur ofta behöver ni göra det?
9) Finns det några svårigheter i att göra analyser?
   a. Tekniska?
   b. Kompetens?
   c. Tidskrävande?
10) Kan det uppstå tillfällen då du måste göra det snabbt?
11) Hur hanterar ni situationer då felärenden uppkommer?
    a. Hur kan du identifiera dessa?

Behovsbilden

12) Hur ser du på behovet av ett system som är utvecklat för att analysera ledtider och handläggningsprocessen?
    a. Finns det ett användningsområde?
13) Vad ser du för möjligheter med ett sådant system?
14) Vilka typer av frågor skulle du vilja kunna besvara med ett sånt system?
15) Vid vilka situationer skulle ett sånt system skulle användas hos er?
16) Vem skulle använda ett sådant system?
17) Ser du några uppenbara krav på ett sådant system?
18) Problematik med ett sådant system?
    a. Ekonomiskt?
    b. Tidskrävande?
19) Har du sett något liknande system användas?

Avslut

20) Har du något som du vill tillägga?

Appendix B

Case Info dimension

Column name	Data type	Description
case_info_key	int	Primary key of case info
caseid	varchar	ID of case
case_status	varchar	The current status of the case
case_description	varchar	Description of the case
case_fakturerad	boolean	True/false if invoice has been sent
case_isarchived	boolean	True/false if the case has been archived
case_typ_av_arbete	varchar	Describes the type of work associated to the case
case_typ_av_plats	varchar	Describes the type of location associated to the case
case_omfattning	varchar	Describes the scope of the case
case_fakturamottagare	varchar	Name of the invoice receiver
case_entrepenör	varchar	Name of the entrepreneur associated to the case
case_date_from	date	Validity starting date (SCD type 2)
case_date_until	date	Validity ending date (SCD type 2)
case_version	int	Indicates the current version of the case (SCD type 2)

Role dimension

Column name	Data type	Description
role_key	int	Primary key of role
role_id	int	Id of role
role_ansvarigutföraregrupp	varchar	Name of head of responsible administrator group
role_ledningsägare	varchar	Name of the responsible administrator
role_date_from	date	Validity starting date (SCD type 2)
role_date_until	date	Validity ending date (SCD type 2)
role_version	int	Indicates the current version of the case (SCD type 2)