Master Thesis

Data Transformation Portal

by

Sara Andersson

LIU-IDA/LITH-EX-A--10/021--SE

2010-05-17

Supervisor and examiner:
Henrik Eriksson
Abstract

The purpose with this report is to present the findings from the thesis work performed at Ipendo Systems. The goal was to develop a methodological support for the migration process and implement a web portal for migration of data.

When a company acquires a new application to perhaps replace a legacy system, or to improve their efficiency and thereby their competitiveness, the company’s data need to be transferred into the new application. The process of transferring the data from a source to a target is called data migration. Because the source and target systems probably have a somewhat different architecture, some transformation to the data has to be made.

The thesis is divided into two parts, a theoretical part where I learned about data migration and developed a methodological support for data migration projects. The second part of the thesis work was practical and I developed a data transformation portal.

Data migrations are often a somewhat forgotten activity in a project. It is sometimes carried out without a proper plan or structure. To bring some structure to this important process I developed a methodological support. The methodological support is made like a guide for how to conduct data migration projects. The purpose of the methodological support is to make data migrations more visible as an own project and add more structure to it. The methodological support is divided into five phases. The five phases are planning, analysis, design, implementation and validation. Every step has its own milestones and deliverables so the support could be used as a sort of checklist during the project.

I have also developed a web portal in SharePoint. The purpose with a data transformation portal is to gather all data migration to one common area without a third-party migration tool and minimize the technical complexity associated with data migration projects. I have developed two modules for the portal. The first module concerns migration from an Excel document to a SharePoint list. The second module handles upload of documents to a SharePoint document library. The portal has functionality like data mapping, validation and setting of metadata. Migration of data is a specific process, depending of the type of data that should be migrated it requires a somewhat different approach. A data transformation portal which can visually monitor, filter, transform and import various types of data to and from various data sources would facilitate the migration process.
Sammanfattning

Syftet med denna rapport är att presentera mina slutsatser från examensarbetet jag utfört på Ipendo Systems. Målet var att ta fram ett metodstöd för datamigreringsprojekt och att utveckla en webb portal för migrering av data.

När ett företag anskaffar en ny applikation som kanske ersätter ett äldre system, eller som ökar deras effektivitet och ger dem konkurrensförde lar, behövs företagets data förflyttas till den nya applikationen. Processen när data flyttas från en källa till ett mål kallas datamigrering. På grund av att källan och målsystemet troligtvis har skillnader i arkitekturen måste någon form av transformering också göras.

Examensarbetet är indelat i två huvudsakliga delar, en teoretisk del där jag studerat datamigreringsprojekt och tagit fram ett metodstöd. Den andra delen var mer praktiskt utformat där jag utvecklat en portal för datamigrering (data transformation portal).


1 Introduction
In this chapter the reader is introduced to the background and the purpose of the thesis.

1.1 Background
Ipendo Systems is an IT company that delivers business oriented IT solutions. Their main market is Northern Europe. The company is divided in two business areas; Ipendo Solutions and Solutions Experts. Ipendo Systems work with Microsoft products and are Gold Certified Partner. Implementation projects are a part of being a provider of IT solutions. And with implementation comes often migration of data as an effect of the implementation.

Data migration is a complex part of a software implementation project. It is often a task that is somewhat overlooked and carried out without a proper plan or structure. Data migration is a time consuming process and the need for efficiency improvements is great.

1.2 Purpose
The purpose with this thesis is to present a methodological support for the migration process and develop a web portal for migration of data.

1.3 Problem Formulation
- How would a methodological support for data migration be designed to structure the migration process?
- How should a Data Transformation Portal (DTP) be created and designed to support and structure the migration process?
- Can the DTP facilitate the migration process?
- How should the DTP be designed to be as intuitive as possible to minimize the need of technical knowledge of the user?

1.4 Target Group
The intended readers of this thesis report are mainly students and teachers at Linköping University and employees at Ipendo Systems. This thesis report is written so that the target group can understand the content without having any specific knowledge in the topic. People not included in the specific target group may of course also read the report.

1.5 Delimitations
This thesis only addresses the issues and strategies of data migration.

For the implementation of the data transformation portal I had to make some limitations due to lack of time. I chose to develop two modules for the portal: migration of an Excel document to a SharePoint list and migration of documents into a SharePoint document library.
1.6 Outline of the Report
First the reader is introduced to theories around data migration. In chapter three the methodology of the thesis work is explained thoroughly. Chapter four is devoted to the methodological support for data migration, the purpose and the need for it. In chapter five the reader can read about the development of the Data Transformation Portal. Chapter six is dedicated for alternative solutions regarding the development of the portal. In chapter seven is the conclusion of the thesis work presented. The report finishes with future work in chapter eight.

1.7 Reference Literature
For the theoretical background of this thesis I have done a literature study containing mostly of academic articles from various journals. Also different data integration companies, like Informatica, have been a great help getting a better understanding of data migration and the market as a whole.

For the practical part of the thesis work I have also done some literature studies. I have been using books about C# programming and SharePoint. Also various online forums and blogs have been helpful in my work.

1.8 Abbreviations
The following abbreviations are used in the report. The first time they are used they are spelled out.

- DTP – Data Transformation Portal
2 Theoretical Background

This section aims to give the reader a basic understanding of data migration, some common strategies and best practices. But also show upon the complexity and the common issues with data migration.

2.1 Data Migration

When a company acquires a new application to perhaps replace a legacy system, or to improve their efficiency and thereby their competitiveness, the company’s data needs to be transferred into the new application. The process of transferring the data from a source to a target is called data migration. Because the source and target systems probably have a somewhat different architecture some transformation to the data has to be made (Cheong et al., 1992).

Today’s business environment with standardization and increased numbers of mergers and acquisitions has lead to a need for optimized IT infrastructures. Optimization can be crucial for business competitiveness but you also have to maintain the consistency of information needed to support the business. The information can contain data from many generations of technology. According to Dey et al. data migrations are simply afterthoughts. They are viewed apart from the mainline development efforts associated with the systems they are used to populate, and often, the migration task is not even considered until the target application is nearing completion (Dey et al., 2007).

Migrations have taken on a strategic importance. The wanted result of an IT project is that service and functionality should be as good as or better than before, while free of costly redundancies. Poorly executed migrations reduce these returns and fail to meet the “as good as or better than before” criteria. Second, reengineering and consolidation programs are complex, and should be implemented with the correct priority and only when needed. Unless migration methods and tools can support the project, it will tend to be overly costly and time consuming (Dey et al., 2007).

Data migration is often spoken about in three steps, extraction, transformation and load (ETL). Data extraction is the process of extract data from the source system and storing the data into an extracted file. The data loading is the process of transferring the extracted file to the target system. If the source and target system have a similar interpretation and architecture the mapping can be relatively simple. But if the two systems are different from each other transformation of data is necessary (Cheong et al., 1992). Many of the existing migration tools today, like some of Informatica’s products, use ETL.

2.2 Migration Strategies

According to Shepherd is roughly two-thirds of the Fortune 1000/Global 2000 companies engaged in some form of data conversion project at any given time. The data conversion projects include migration from legacy systems to packaged applications, data consolidations, data quality improvements and creation of data warehouses and data marts. Unfortunately many of these projects are not successful. According to Shepherd as many as 74% of all IT projects carried out during 1998 either overran budget and time or failed. And according to PublicTechnology.net the number is approximately the same today with 80% failure rate on
data migration projects (PublicTechnology.net, 2009). One of the primary reasons for this huge failure rate is the lack of a thorough understanding of the source data early in the project. Shepherd suggests a data migration strategy divided in six steps as a possible solution to this problem. The six steps consist of data profiling and mapping. Each step is building on the information produced in the previous step. The result of the six steps will be transformation maps which could be used together with a third-party data migration tool. The six steps are: Column profiling, dependency profiling, redundancy profiling, normalization, model enhancement and transformation mapping. Data profiling and mapping, if done correctly, can dramatically lower project risk and lead to that companies can complete their data migration projects with success (Shepherd, 1999).

Figure 1 - Six steps of data profiling and mapping. Source: Shepherd, 1999.

Data migration is an important part of legacy system migrations. There are two common migration strategies when it comes to migration of legacy systems: The Chicken Little methodology and The Butterfly methodology.

The Chicken Little migration strategy consists of 11 steps. In this method the legacy and the new target system run in parallel throughout the migration process. The target system is built during the migration, until it has the functionality of the old legacy system. During the migration process data is stored in both the legacy and the target system. In order to maintain data consistency gateway coordinators is often used (Wu et al., 1997).

The Butterfly methodology divides the migration process into six phases. Unlike Chicken Little the Butterfly method only stores data in the legacy system during the migration. The target system will not be used until the whole migration process is finished (Wu et al., 1997).

According to Wu et al the existing migration methodologies are either so general so that they omit many of the specifics or too complex that they are almost impossible to use in practice. The Chicken Little offers the most mature approach (Wu et al., 1997).

According to NetApp there is one migration methodology, see Figure 2 - Migration Methodology, no matter on the vendor of the application or storage device or if the migration is carried out by internal IT or by a third party (NetApp Global Services, 2006).
2.3 Best Practice

How should you carry out a data migration project? Is there a best practice?

According to a survey paper by Bloor Research there is an increasing market for data migration. Companies spend a lot of money, time and resources on these projects. The authors of the paper think it is surprising that a market of this size has no recognition as a market in its own right. They would advocate that such a market, with standards, best practices and so on, needs to be established in order to ensure that appropriate disciplines are in place to reduce or eliminate the time and budget overruns of these projects (Howard & Potter, 2007).

In companies today there are many reasons to make data migration. Technology update, server and storage consolidation, data center relocation, data classification, and mergers/acquisitions just to mention a few. Despite companies do data migration regularly the process do not become an easy routine project. Migrating, or moving data from one storage device to another is a complex process. Companies mission-critical data, data availability demands, downtime acceptance and also the risk of performance impact to production environments, technical incompatibilities and data corruption/loss makes migration one of IT’s biggest challenges (NetApp Global Services, 2006). According to the survey paper by Bloor Research the interviewed companies shows a great interest in adopting a best practice for their data migration projects (Howard & Potter, 2007).
According to PublicTechnology.net it exist best practices. But even with best practice models, too many projects rely on inventing their own approach, with not enough thought given to aligning the migration with the actual business need (PublicTechnology.net, 2009).

2.4 Migration Issues
According to publicTechnology.net most data migration projects technology and business remains largely separate from each other until problems begin to manifest on the business side. Three of the biggest issues in data migration projects are, according to publicTechnology.net, inadequate data preparation, poor business engagement and underestimating project complexity (PublicTechnology.net, 2009).

Many of the issues brought up later in this section can be found in a research paper written by Bloor Research. The paper is based on interviews with 43 of the Forbes Global 2000 companies. All the interviewed companies had projects where the overall budget for the project was $1m or more (Howard & Potter, 2007).

2.4.1 Data Cleansing
One great issue with data migration is data quality. The answer to bad data quality is data cleansing and transformation. Data transformation is needed to make changes in structure or representation. The primary issue for data cleansing to solve is errors and inconsistency in data content, for example misspellings, duplicates and contradictory values. To perform data cleaning you need to go through several phases. The first phase is data analysis where you detect what kind of errors are to be removed. Next phase is defining rules for transformation and mapping. To complete the cleansing operation you have to do verification and the actual transformation (Rahm et al., 2000).

What is good data quality? According to Müller and Freytag data quality can be divided into data quality criteria. The data quality criteria form a hierarchy. The two main criteria are accuracy and uniqueness. An accurate data collection does not contain any data anomalies, but can contain duplicates. A unique data collection does not contain any duplicates. A data collection that is both accurate and unique does not need data cleansing (Müller & Freytag, 2003).

2.4.2 Strategic or Tactical
A data migration project can either be carried out as a strategic or tactical project. Because data migration projects are part of a bigger project it is often carried out only as a tactical challenge. This tactical perspective leads organizations to seek “quick-fix” solutions to data migration, but these do not work because data migration is a much more complex project than organizations initially expect. The strategic view of a data migration project means that migration should be viewed as an ongoing process of making data work, no matter what changes occur in the organizations systems. To make a data migration project a success you have to carry out the project with strategic intention and not tactical (Informatica, 2004).
2.4.3 Time and Budget
A problem with data migration projects is time and budget overrun (Howard & Potter, 2007). What implications does this have on the data migration project itself, on the broader application project it belongs to and in the end for the whole business?

According to a survey paper by Bloor Research 51% of all data migration project went over both time and budget. The research paper brings up a few reasons to why these projects go over time and budget. Almost 50% of the respondents blame the time overrun on budget issues. Also scoping issues and improper use of tools is mentioned (Howard & Potter, 2007). According to publicTechnology.net more than 80% of data migrations activities fail to hit budget and time targets (PublicTechnology.net, 2009). And according to Shepherd 74% of the IT projects carried out during 1998 failed (Shepherd, 1999).

If the data migration project fails, or technically succeeds but is unaccepted by the users, then the application project itself will have failed with direct costs to the business (Howard & Potter, 2007).

2.4.4 Scoping
Project scoping and estimating is a serious issue according to the survey paper by Bloor Research. Scoping issues was one of the reasons for time and budget overruns. Despite that the majority of the respondents thought that their scoping activities were successful (Howard & Potter, 2007).

The authors of the Bloor research paper believe that a proper use of tools is one of the key elements in ensuring realistic scoping that would help prevent time and budget overruns (Howard & Potter, 2007).

2.4.5 Methodology
According to a survey paper by Bloor Research the majority of respondents had a formal methodology that they used in their data migration project. Most of the used methodologies had been developed in-house. The research paper refers to the projects time and budget overruns and draws the conclusion that either these methodologies were not so good or were not used properly (Howard & Potter, 2007).

A majority of the respondents in the Bloor research were interested in adopting a best practice methodology (Howard & Potter, 2007).

2.5 Microsoft Office SharePoint Server 2007
SharePoint, or Microsoft Office SharePoint Server (MOSS), is Microsoft’s content management system. SharePoint can help improve organizational efficiency and facilitate information-sharing across boundaries for better business insight (Microsoft Office SharePoint Server, [www]).

Microsoft Office SharePoint Server 2007 provides a single, integrated location where employees for an example can collaborate with team members, find organizational resources, search for experts and corporate information and manage content. Some capabilities of SharePoint are collaboration and social computing, enterprise content management, portals,
business process and forms, enterprise search and business intelligence (Microsoft Office SharePoint Server, [www]).

Of the SharePoint capabilities I have used Portals. Portal sites connect people to business-critical information, expertise, and applications. SharePoint is an enterprise portal platform that makes it easy to build and maintain portal sites thanks to a comprehensive framework (Microsoft Office SharePoint Server, [www]).
3 Methodology

The thesis work is divided into two parts: The development of the methodological support for data migration and the implementation of the data transformation portal. In this section, the methodology for the two parts of the thesis work is described.

3.1 Development of the Methodological Support for Data Migration

To be able to create a methodological support for data migration, a literature study was made. First, I read a lot about data migration in general and as I became more familiar with the subject, I was able to study the most complex parts deeper. In my literature study, I used academic articles and technical journals I found on the internet.

3.2 Implementation of the Data Transformation Portal

During the development of the DTP, I worked according to Ipendo Systems' development methodology. In the initiation phase, you define the scope and the goals with the project. In the design and planning phase, you set up the requirements of the project and make the functional anatomy and the use cases. A prototype can be useful to describe the interface. Last in the methodology is the implementation, test, go live, closing, and evaluation. In the following sections, you can read more about the steps of the implementation of the DTP.

3.2.1 Functional Anatomy

The functional anatomy gives an overview of the system. The functional requirements and their priorities form a map that makes it easy to overview. The map makes it easy to see the dependencies and functions of the requirements. The functional anatomy consists of groups. Each group in turn consists of use cases, which describe the functional requirements. The use cases can be associated with one or several functions.

3.2.2 Use Cases

The use cases from the functional anatomy are described thoroughly. In each use case, actors, precondition, success guaranteed, main scenario, and extensions, if any, are presented. The actors are those who are involved in the scenario; it could be a person or a system. The precondition explains to the reader what tasks or events that must have been carried out or occurred before the use case can start. The success guaranteed tells the reader what condition must be reached, or object be created for the use case to be considered finished and successful. The main scenario describes the actor’s interaction with the system. The main scenario is explained step by step. The main scenario can have an extension or an alternative flow which is described step by step in the extension section.

3.2.3 Prototyping

To simulate the use cases, a prototype was made. The prototype also enriched the use cases with views to give a better understanding for the reader. The prototype was built in PowerPoint. In PowerPoint, it is easy to make buttons, text, and various fields. You can create hyperlinks between the slides, making it easy to make the prototype interactive in slide show mode. PowerPoint also has the advantage that it is familiar, which means no time on learning a completely new program. Another option for prototyping is paper, but it has the disadvantage
of not being interactive in the same way as a computer prototype. A paper prototype also makes it harder to do small changes to the design.

3.2.4 Implementation

When the functional anatomy, the use cases and the prototype had been approved by my supervisor the implementation of the data transformation portal could start.

The Data Transformation Portal was supposed to run in SharePoint. In order for me to be able to start experimenting, a development environment was set up in Visual Studio 2008. In my virtual environment I also had access to Microsoft Office SharePoint Server 2007. After I had done a lot of reading about SharePoint on the internet I decided to build web parts for my portal. To be able to build web parts I realized I had to learn C# and SharePoint. I borrowed some books from the library about programming in C# and the technologies of SharePoint. Internet with forums and blogs has been a great help in the development process. Some examples of forums I have used are: daniweb.com, dreamincode.net, stackoverflow.com, eggheadcafe.com and social.msdn.microsoft.com.

During the implementation I used the Microsoft .NET Platform, Visual Studio 2008, C#, Microsoft Office SharePoint Server 2007, SQL Server 2008 and ASP.NET.

I developed the modules for the portal as web parts in Visual Studio. Then I deployed them to SharePoint and added the web parts to web part pages. You can read more about the modules in chapter 5, Data Transformation Portal.
4 Methodological Support for Data Migration

Data migrations are often a somewhat forgotten activity in a project. It sometimes is carried out without a proper plan or structure. To bring some structure to this important process I have developed a methodological support. The purpose with the methodological support is to have a general guideline for how to conduct a data migration project. The guide makes the data migration project more visible as a part of the bigger project. It also makes the migration process to an own project, makes it more important and reduces the risk for it to be forgotten in the initial planning of the bigger project. The methodological support that I made is a plan for how to conduct a migration project in a structured way. It includes all the important steps from planning to validation. You can read more about the steps in the following section.

4.1 The Five Phases

The data migration process consists of, according to me, five main phases: planning, analysis, design, implementation and validation. Every phase in the methodological support have their own deliverables and milestones. The phases are described in more detail in the following sections.

4.1.1 Planning

In this phase a plan for the migration project is being made. Some important aspects to consider in the planning phase are: Scope, which data is to be migrated, where, when and how. Requirements, how much downtime is acceptable? Should both the source and target system be up and running simultaneously? How many migrations should be made? Goal, define the overall goal for the project. Organization, how many people is needed in the project group? Define the group members’ roles and responsibilities. You also need a test group for the validation phase.

4.1.2 Analysis

In the analysis phase the data to be migrated is analyzed and a migration plan is formulated. The migration plan should contain the aspects brought up in the planning phase and the results from the data analysis. Data set, what size are the data? Does this bring up some difficulties or technical concerns? Sources, how many sources are involved in the migration? Analysis of data structures, analyze the structure of both the source and target data. This gives an indication of what scale of mapping and transformation that is needed. Migration tools; decide if the migration requires a migration tool and acquire it.

4.1.3 Design

In the design phase the mapping and transformation is planned and prepared. The phase should result in a transformation plan and a mapping schedule. A deeper analysis of the data is done to be able to make the mapping schedule. The mapping schedule should describe the source and destination of specific data. The transformation plan describes which changes and quality improvements are to be made to the data during the transfer. The mapping schedule is a part of the transformation plan.
4.1.4 Implementation
In the implementation phase a test migration or proof of concept is being carried out. The test is done with a small data set to validate that the mapping and transformation is correct. The migration can be made with various approaches; automatic, manual or semi automatic. Another thing to consider is if the migration is going to be big bang or phased.

4.1.5 Validation
After the migration is carried out you have to validate the migrated data. This is where the test group comes in. A sample of data is picked out and tests are being made according to a test plan or checklist. Mapping and data quality are important factors to validate. It is also important to test the new system’s functionality with the migrated data.
5 Data Transformation Portal

The purpose with the data transformation portal is to gather all data migration to one common area without a third-party migration tool. The portal is supposed to handle various types of data migration.

Migration of data is a specific process, depending of the type of data that should be migrated it requires a somewhat different approach. A data transformation portal which can visually monitor, filter, transform and import various types of data to and from various data sources would facilitate the migration process (Kvist & Nijm, 2009).

The portal is supposed to support export to various types of file formats, for example Excel and Word documents and text files. It should also support export to database and various kinds of containers in SharePoint, for example list and document library. Import from various types of file formats, for example Excel and Word documents and text files. Support for import of databases. The portal should also support validation, transformation and setting of metadata.

To make the Data Transformation Portal work for various kinds of migrations the user creates templates depending on the type of migration. During the migration process you create two templates, one import template and one export template, see Figure 3. The import template makes it possible to get the imported data in the format you want in the staging area. The export template maps the data from the staging area to the target.

![Migration process overview](image)

When I started to implement the portal I decided to first implement one module or migration flow of the portal. I chose migration and transformation of an Excel document into a SharePoint list. This module became some sort of proof of concept. I chose Excel and
Data Transformation Portal

SharePoint list because they have a somewhat similar architecture of rows and columns. Because I chose to import data to a SharePoint container i.e. a list I had to learn a lot about how SharePoint is built and what you are able to do with lists. And since I think SharePoint is the central and most interesting thing in my thesis the choice was easy to make.

When the first module was working I started developing the next one. The second module was a document upload module. The user was supposed to select a folder and upload it and its content of files and subfolders to a specific document library in SharePoint. To the uploaded material the user was supposed to be able to write metadata. Metadata is an important feature in SharePoint and therefore it felt important to research and learn more about this area.

5.1 Functional Anatomy
The functional anatomy is divided into groups, use cases and functions. The functional anatomy for the DTP is divided into five groups: Migration workflow, import, export, error handling and validation. Each group contains a number of use cases. Each use case contains functions. The functions describe the systems functionality connected to that specific use case. The functional anatomy consists of a list of all the requirements, described in use cases and functions but also a map to show the use cases and the functions dependencies. The functional anatomy worked as a kind of requirement specification during the whole development process. See Appendix B – Functional Anatomy.

5.2 Use cases
Out of the use cases in the functional anatomy you can chose to write complete use cases on some of them. I chose to do complete use cases on all of my 22 listed in the functional anatomy. When writing the use cases it really made me think of how the portal should work. The use cases describe how the user interacts with the system. When writing the use cases I created a work flow of how the portal should be used. I knew that the more detailed the use cases were, the easier the implementation part was going to be. The use cases first states the actors involved in the use case, such as the user and the system. All use cases start with the preconditions of the case and the success guarantee. When the start and goal is made clear the main scenario in described. Last you declare all of the alternative flows that may occur. It is common to have pictures to illustrate the main scenario. That is why I in my next step chose to make a prototype.

5.3 Prototype
I made a prototype of the portal in PowerPoint. The prototype was made with an easy layout but supported interaction, such as clickable buttons. When I made the prototype it really made me think about the design and how difficult it is to make an easy-to-use application with an intuitive flow. I made the prototype based on the use cases, but as I made it I did some changes to the functionality and the flow. You get a better view of the whole flow when you see it in pictures, rather than plain text. I used the prototype as illustrations in my use cases. I think it lifted the use cases and made them easier to understand. The use cases in combination with the prototype helped me during the implementation. When I became uncertain of the functionality or the flow I just could go back and look at a specific use case.
5.4 Implementation
The implementation is divided into two modules. In the following sections I describe both the modules I have implemented.

5.4.1 Module 1 – Excel Document to SharePoint List
The idea with this first module was to transform an Excel document into a SharePoint list. The whole transformation process is made as a flow or a kind of wizard to guide the user through the process. In the import template the mapping between the Excel document and staging is done, see Figure 4. The user can also choose to include various kinds of validation to be carried out during the migration. In the export template the mapping between the staging and target is specified. In the export template is also where the SharePoint list is being created. The wizard starts with the user choosing a name for the project. The user also has to choose type of project, Excel document to SharePoint list or migration of documents to document library, see Appendix A – Screenshots of Data Transformation Portal, Figure 5. Similar to most wizard styled applications the user then clicks next.

![Figure 4 - Excel document to SharePoint list](image)

Import Template
Next step in the wizard is the selection or creation of an import template. All the available import templates are shown to the user, see Appendix A – Screenshots of Data Transformation Portal, Figure 6. The user can browse among the import templates and the basic information such as name, mapping and validation will be shown to the user. Now the user can either select one of the available import templates or chose to create a new one. If the user chooses to go with an available import template the wizard will continue to the export template. If the user chooses to create a new import template the user first has to give the import template a name, see Appendix A – Screenshots of Data Transformation Portal, Figure 7. In the next step the mapping is selected, see Appendix A – Screenshots of Data Transformation Portal, Figure 8. The user maps the columns in the Excel document, here named A, B, C and so on, to the
staging area. This can seem like an unnecessary step but the import template lets you organize the data in any document so that it always will look the same in the staging area. So if you have same kind of information but in several different kinds of document types you can create a custom import template to get the information organized in staging. Then you can use one export template to get the data from staging to target, independent on the original document type. The next step in creating the import template is validation, see Appendix A – Screenshots of Data Transformation Portal, Figure 9. The chosen validation is carried out in the migration step. The import template, the mapping and validation rules are saved in a database.

**Export Template**
In the next step in the wizard the user creates an export template. The user can chose to use a saved export template to the project or create a new one. The selection process of an available template follows the same pattern as in selection of import template, see Appendix A – Screenshots of Data Transformation Portal, Figure 10. If the user chooses to create a new export template he or she gives the export template and the SharePoint list names, see Appendix A – Screenshots of Data Transformation Portal, Figure 11. It is also possible to give the SharePoint list a description. The next step in creating the export template is mapping, see Appendix A – Screenshots of Data Transformation Portal, Figure 12. In the export template the data from staging is mapped to the target. The columns in staging are listed on the page and the user can give the columns in the target, here the SharePoint list, names. After the mapping is done the user can chose validation, see Appendix A – Screenshots of Data Transformation Portal, Figure 13. At this moment the only available validation rule to choose is validation of rows. The template, mapping and validation rules are saved in a database.

**Migration**
The next and last step in the wizard is the selection of file and then the actual migration, see Appendix A – Screenshots of Data Transformation Portal, Figure 14. The user selects an Excel file and clicks on upload. If the uploading of the file is successful the start migration button becomes visible. The user clicks on the start migration button and the migration is carried out with all the selected mapping and validation, see Appendix A – Screenshots of Data Transformation Portal, Figure 15. When the migration is finished a log report of the validation is shown on the page. A log file is also saved on the server in the projects name. After the migration is finished the user can open the SharePoint list, see Appendix A – Screenshots of Data Transformation Portal, Figure 16.

### 5.4.2 Module 2 – Import Documents to SharePoint Document Library
After I had implemented module 1 I could start with the next one. I choose, together with my supervisor at Ipendo Systems, to implement a module for document upload. The document upload module was supposed to handle uploading of whole folder hierarchies and the setting of metadata. The folder was supposed to be uploaded from a client computer to a chosen or created document library in SharePoint. At the same time the user was supposed to have the opportunity to set metadata on the folder and documents. During the implementation the module focus changed from folder upload to document upload depending on a number of reasons which you can read about in chapter 6, Solution Alternatives. This module follows the same concept with the wizard flow and the templates as the first module. The user first
creates a new project by giving it a name and selects the type of the project, see Appendix A – Screenshots of Data Transformation Portal, Figure 17.

**Import Template**

At the moment the import template does not contain anything except for the selection of template name. If further development comes in question the import template can contain for example validation rules. I wanted the wizard to follow the same flow as the first module to keep the portal as user friendly as possible. You can still choose among the available templates as in module 1, or create a new one.

**Export Template**

The next step in the wizard is the selection of an available export template, or the creation of a new one, see Appendix A – Screenshots of Data Transformation Portal, Figure 10. If the user chooses to create a new export template he or she first gives the template a name, see Appendix A – Screenshots of Data Transformation Portal, Figure 18. The user can then choose to use an existing document library. If the user chooses an existing library the metadata fields which were created with that library will be used also in this project. If the user chooses to create a new document library he or she has the opportunity to create metadata fields, see Appendix A – Screenshots of Data Transformation Portal, Figure 19. The user first gives the new library a name and then clicks on add metadata fields. The metadata fields have to be given a name and allocated a data type (text, number or date). The user can add one to five metadata fields for each document library. The export template and the metadata settings are saved in a database. The document library with the metadata fields is created in SharePoint.

**Migration**

Next step is the migration. All the metadata fields from the selected or created document library are now listed on the page, see Appendix A – Screenshots of Data Transformation Portal, Figure 20. Next to the name of the metadata field the user can choose if the metadata should be extracted from the file’s parent folder, from the actual file or the user can fill the field with any text. If the user chose that the metadata should be extracted from the file three new options appear on the page. The user can choose to extract the author, last modified date or creation date from the file. When the user has filled all the metadata fields all there is left to do before the migration can start is to choose a file. The user selects a file and clicks on upload. The file is now uploaded and the metadata is set. The user can now choose to upload another file to the same document library or close the project. After uploading the file it will directly appear in the selected document library, see Appendix A – Screenshots of Data Transformation Portal, Figure 21.
6 Solution Alternatives

In this section I motivate and explain certain choices and solutions I have used in the portal.

6.1 Web Parts
I choose to build the entire portal as web parts. An alternative had been to build it more as a site in ASP.NET. But as I never worked with SharePoint and web part I thought it was a great opportunity to get to know it. I also found a lot of information and various tutorials on how to build web parts, which was very helpful in the startup phase.

6.2 Two Templates
I use two templates in my solution, one import template and one export template. The alternative would have been to use only one. In which case, you then have to map the data directly from the source file to target, and there had not been a need for the staging area. The drawback with only using one template is that you most likely have to create a new template for every project. Consider the case where you gather information from various customers once a week. The information is the same every week: articles, amount and price. You get the information from your customers in various kinds of formats: Excel documents, text files or database files. Then it would be nice if you could reuse the import template that fits that special document you receive regularly and migrate the data into your system.

6.3 Staging Area
The staging area makes it possible to organize the data in a standardized way. With the help of the import template you can organize the data so that data can work with a specific export template. So no matter if you have the same information in an Excel document or a text file, you can make it look the same on staging and therefore use one and the same export template.

6.4 Document Upload
The second module I developed was meant to be a folder upload. The user was supposed to be able to select a folder on his or her computer and upload it and all its content to a document library in SharePoint. At the same time the user was supposed to be able to set metadata on the uploaded content. Sadly I had to reconsider my solution for the document upload. When I started to search for information on how to upload a folder and folder hierarchies I soon realized that this task was not going to be easy. Because the code runs on the server the upload from a client computer to the server is limited to one file. There are a few possible solutions to this problem. Develop a JavaScript or ActiveX plug-in that runs on the client computer and not on the server. You can then through the plug-in gain access to the client computer. Another possibility could be to write some client based code that generates a zip file of the selected folder. The zip file can then be uploaded to the server where it will be unpacked. In the SharePoint document libraries it exist an explorer view. The view opens the document library as a Windows explorer window. Then you can drag and drop files and folder to the document library from your client computer. Unfortunately I did not find a way to use the explorer view programmatically. Due to lack of time I did not have time to test these possible solutions in practice.
What I have done is to develop a function which iterates a folder structure and uploads it to SharePoint, both the files and folders. But this solution requires that the folder is placed on the server. To extract metadata from the files they have to be physically located on the server, otherwise you will not be able to access the metadata.

One possible modification to my solution as it is now could have been to select the document you wanted to upload and fill in the metadata and then click on add. The document and its metadata appear in a list and you can add as many file to it as you want. When you completed the adding of documents and metadata you can click on migrate and all documents will be uploaded at the same time.

6.5 More Modules
I would have wanted to implement more modules or made the document upload module working with folder hierarchies. But due to time shortage it was not possible. You can read my proposal to further development of the portal in chapter 8 Future Work.
7 Conclusion

In my thesis work I have developed a methodological support for data migrations and developed a data transformations portal. These are my conclusions and findings.

Companies are spending a lot of time, money and effort on data migration projects. Many of these companies fail. Fail to make the projects successful or fail to hit time and budget targets. In my thesis I have tried to share some light upon the somewhat forgotten data migration projects. I think the data migration projects need to be more visible and be given a more strategic importance. I also think that the right tool can make the migration process less technically complex.

The methodological support for the data migration process I developed should work like a guideline for how you should conduct data migration projects. Its purpose is also to make the data migration process more visible in the bigger project it is often a part of. The structure and design of the methodological support is made so it would be easy to follow and its outline is inspired by Ipendo Systems own project management methodology. Every chapter has milestones and deliverables so it can be used almost like a checklist. I would say that the methodological support is general and can be used in all data migration projects.

I have developed a part of the data transformation portal as a proof of concept. The portal could theoretical be used in data migration projects, when fully developed. The idea of the portal is very good and it takes away a lot of the technical complexity associated with data migrations.

The question is if the portal can be made both general and still have all the wanted functionality that is specific for each migration type. I think that at some point you will have to compromise between the general solution and the solution with all the specific and desired functionality. I tried to make it flow as general as possible and it will follow the same pattern independent on which type of migration project the user chooses. The portal is dynamical in that sense that the user can make a lot of various options. Of course it has room for even more options and functionality in future development.

For the portal I used a wizard design and divided the process into steps. This makes the process logical and the intent is that it should be easy for the user to understand the flow. The thought was also to take away a lot of the complexity associated with data migrations. The portal has not been user tested because in was not a part of this master thesis. It would be interesting to see the result of a user test on the portal. The user test would show on the usability and bring the improvement possibilities to the surface.
8 Future Work

According to Howard and Potter (2007) and the Bloor Research survey there is a lot left to be done in the data migration market. A demand for standards and best practices are growing and organizations are spending a lot of money on their data migration projects (Howard & Potter, 2007).

There is also a lot left to do to make the portal complete. Even the modules I have implemented, as a sort of proof of concept, miss some parts of the wanted functionality according to the requirement specification. Below I have listed some of the most vital parts I think the portal should have.

**Workflow**
Possibility to create workflows connected to a created list or a document library. The user should be able to dynamically create a workflow and choose field, condition and action connected to a specific container in SharePoint.

**Update Projects and Templates**
The user should be able to select available templates and modify the rules and properties of it and then use them in new projects. In the present solution it is possible to select a saved template but not to modify it. The user should also be able to select a saved project and run it or modify it.

**Transformation Rules**
The user should be able to add more transformation rules to the project. The only implemented transformation rule at the moment is mapping. It is likely that some sort of more advance data cleansing is wanted in the future.

**Folder Upload**
Possibility to upload a folder and its content from the client computer to a document library in SharePoint is desired.

**Migration of Databases**
Opportunity to migrate databases is missing from the current solution. Both import and export of a database should be possible in the future.

**Export of Files**
Functionality to convert to various kinds of file formats is missing in the current solution.

**Extended Error Handling**
The current solution lacks functionality to handle error. Now warnings are created in validation. These warnings are ignored in the migration process but logged to the log report. If a major error occurs that have a significant impact on the migration process the user also should have the possibility to abort the process and make a rollback.
References


PublicTechnology.net (2009). “Data migration problems are exactly the same as they were 10 years ago” [www] http://www.publictechnology.net/content/21449, 2010-03-25.


Figure 5 - Create migration project (module 1)
Figure 6 - Create or select import template
Figure 7 - Create import template (module 1)
Figure 8 - Create import template - mapping (module 1)
Figure 9 - Create import template - validation (module 1)
Figure 10 - Create or select export template
Figure 11 - Create export template (module 1)
Appendix A – Screenshot of Data Transformation Portal

Figure 12 - Create export template - mapping (module 1)
Figure 13 - Create export template - validation (module 1)
Figure 14 - Migration - select file (module 1)
Figure 15 - Migration (module 1)
Figure 16 - SharePoint list
Figure 17 - Create migration project (module 2)
Figure 18 - Create export template (module 2)
Figure 19 - Create export template - metadata
Appendix A – Screenshot of Data Transformation Portal

Figure 20 - Migration (module 2)
Appendix A – Screenshot of Data Transformation Portal

Figure 21 - SharePoint document library
## Appendix B – Functional Anatomy

<table>
<thead>
<tr>
<th>Use Case</th>
<th>Description of Use Case</th>
<th>Function</th>
<th>Group</th>
<th>Migration Workflow</th>
</tr>
</thead>
<tbody>
<tr>
<td>U1 Design migration project</td>
<td>Start a new migration project</td>
<td>Set migration project information</td>
<td>Migration</td>
<td>U1 Design migration project</td>
</tr>
<tr>
<td>U2 Update migration project</td>
<td>Update an existing migration project</td>
<td>Set properties, create workflow</td>
<td>Migration</td>
<td>U2 Update migration project</td>
</tr>
<tr>
<td>U3 Create import template</td>
<td>Create transformation rule</td>
<td>Set properties, create workflow</td>
<td>Migration</td>
<td>U3 Create import template</td>
</tr>
<tr>
<td>U4 Select import template</td>
<td>Exclude validation</td>
<td>Create workflow</td>
<td>Migration</td>
<td>U4 Select import template</td>
</tr>
<tr>
<td>U5 Create export template</td>
<td>Create mapping rule</td>
<td>Create workflow</td>
<td>Migration</td>
<td>U5 Create export template</td>
</tr>
<tr>
<td>U6 Select export template</td>
<td>Save template</td>
<td>Create workflow</td>
<td>Migration</td>
<td>U6 Select export template</td>
</tr>
<tr>
<td>U7 Update import template</td>
<td>Set name, length, data type, mandatory, based on filename or info from source</td>
<td>Create workflow</td>
<td>Migration</td>
<td>U7 Update import template</td>
</tr>
<tr>
<td>U8 Update export template</td>
<td>Set new rules set</td>
<td>Create workflow</td>
<td>Migration</td>
<td>U8 Update export template</td>
</tr>
<tr>
<td>U9 Select file</td>
<td>Select files to migrate</td>
<td>Create workflow</td>
<td>Migration</td>
<td>U9 Select file</td>
</tr>
<tr>
<td>U10 Select input file</td>
<td>Select multiple files</td>
<td>Create workflow</td>
<td>Migration</td>
<td>U10 Select input file</td>
</tr>
</tbody>
</table>

**Figure 22 - Functional Anatomy 1/3**
<table>
<thead>
<tr>
<th>Use Case</th>
<th>Description of Use Case</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>Import</td>
<td>UC10 Transform</td>
<td>Start transformation, transform input file, implement mapping rules</td>
</tr>
<tr>
<td></td>
<td>UC11 Load to staging</td>
<td>Process transformation rules, upload staging data</td>
</tr>
<tr>
<td>Export</td>
<td>UC12 Export to file</td>
<td>Start export, process rules</td>
</tr>
<tr>
<td></td>
<td>UC13 Export to DB</td>
<td>Start export, process rules</td>
</tr>
<tr>
<td></td>
<td>UC14 Export to SharePoint</td>
<td>Start export, process rules</td>
</tr>
</tbody>
</table>

**Figure 23 - Functional Anatomy 2/3**
<table>
<thead>
<tr>
<th>Group</th>
<th>Use Case</th>
<th>Function</th>
<th>Description of Use Case</th>
<th>Description of Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>Error handling</td>
<td>UC15 Log issue</td>
<td>Create warning</td>
<td>The logs are saved to the log report</td>
<td>Warnings are ignorable.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Create error</td>
<td></td>
<td>Errors need to be taken care of and then the validation must be run again.</td>
</tr>
<tr>
<td></td>
<td>UC16 View log report</td>
<td>Open log</td>
<td>After a validation the user is shown the log report</td>
<td></td>
</tr>
<tr>
<td></td>
<td>UC17 Handle log issues</td>
<td>Correct error</td>
<td>If an error is detected during validation: the user can choose to either ignore or correct it</td>
<td>Correction of errors are not a part of DTP.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Ignore warning</td>
<td></td>
<td>Choose a warning in the log report to ignore.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Ignore all warnings</td>
<td></td>
<td>Choose to ignore all warnings in the log report.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Continue migration process</td>
<td></td>
<td>After all warnings have been ignored, the process can continue.</td>
</tr>
<tr>
<td>Validation</td>
<td>UC18 Stop migration project</td>
<td>Rollback</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Validation</td>
<td>UC19 Validate input file</td>
<td>Validate file format</td>
<td>Validation of input file</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Validate if a file is selected</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Validate content of file</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>UC20 Validate transform</td>
<td>Validate transformation rules</td>
<td></td>
<td>Validate that the input file are being transformed correctly during transformation.</td>
</tr>
<tr>
<td></td>
<td>UC21 Validate upload to staging</td>
<td>Validate number of rows</td>
<td></td>
<td>Validate that all data are being uploaded to staging, during load.</td>
</tr>
<tr>
<td></td>
<td>UC22 Validate export</td>
<td>Validate number of rows</td>
<td></td>
<td>Validate that all data are being exported to target, during export.</td>
</tr>
</tbody>
</table>
Upphovsrätt
Detta dokument hålls tillgängligt på Internet – eller dess framtidare ersättare – under en längre tid från publiceringsdatum under förutsättning att inga extra-ordinära omständigheter uppstår.

Tillgång till dokumentet innebär tillstånd för var och en att läsa, ladda ner, skriva ut enstaka kopior för enskilt bruk och att använda det oförändrat för ickekommersiell forskning och för undervisning. Överföring av upphovsrätten vid en senare tidpunkt kan inte upphäva detta tillstånd. All annan användning av dokumentet kräver upphovsmannens medgivande. För att garantera äktheten, säkerheten och tillgängligheten finns det lösningar av teknisk och administrativ art.

Upphovsmannens ideella rätt innefattar rätt att bli nämnd som upphovsman i den omfattning som god sed kräver vid användning av dokumentet på ovan beskrivna sätt samt skydd mot att dokumentet ändras eller presenteras i sådan form eller i sådant sammanhang som är kränkande för upphovsmannens litterära eller konstnärliga anseende eller egenart.

För ytterligare information om Linköping University Electronic Press se förlagets hemsida http://www.ep.liu.se/

Copyright
The publishers will keep this document online on the Internet - or its possible replacement - for a considerable time from the date of publication barring exceptional circumstances.

The online availability of the document implies a permanent permission for anyone to read, to download, to print out single copies for your own use and to use it unchanged for any non-commercial research and educational purpose. Subsequent transfers of copyright cannot revoke this permission. All other uses of the document are conditional on the consent of the copyright owner. The publisher has taken technical and administrative measures to assure authenticity, security and accessibility.

According to intellectual property law the author has the right to be mentioned when his/her work is accessed as described above and to be protected against infringement.

For additional information about the Linköping University Electronic Press and its procedures for publication and for assurance of document integrity, please refer to its WWW home page: http://www.ep.liu.se/

© Sara Andersson