Merging field feedback data for product design improvement: Enterprise Search tools versus ad hoc solutions
by
R.W.M. Kuijpers

Bachelor of Engineering - 2007
Student identity number 0633293

in partial fulfilment of the requirements for the degree of

Master of Science

in Operations Management and Logistics

Supervisors:
dr. A.J.M.M. Weijters, TU/e, IS
dr. Y. Lu, TU/e, ID
L.A.M. Nooyens, Xerox
TUE. School of Industrial Engineering.
Series Master Theses Operations Management and Logistics

Subject headings: quality and reliability, information flows, field feedback, IOP Data Fusion project, Enterprise Search
I. Abstract
This master thesis describes the start of a new phase in the IOP Data Fusion Project. The applicability of Enterprise Search tools is examined to merge reliability-oriented field feedback data for product design improvement. The focus of this project is on the merging of mainly structured data. To make this merged data also useful for reporting and analysis, transformation and analysis tools are used to process the data. A test environment is developed to examine these tools, and a user-friendly interface is developed to combine the functionalities in a single application.
II. Management Summary

Problem Context
Companies store more and more business content every year. It therefore slowly becomes difficult for companies to keep structure in all this content. For manufacturing companies, this detailed information combined and analyzed could give insights that might not be discovered directly. It could say something about how their products are actually performing in the field for example. Such information could therefore contribute to a better product development.

This master thesis performed a part of the IOP Data Fusion Project, which is founded to provide an answer to the above-mentioned problem. Their goal is to ‘support product development with prioritized information related to mismatches between product specifications and customer requirements’. To do this, the project focuses on five primary sources of feedback information that companies can retrieve from the field:

- Test data (obtained during the product creation process)
- Service data
- Helpdesk data
- Trade data
- Internet data (e.g. forums)

Data from these five different sources should be combined in order to get an understanding about how a product is performing in the field. While most of the data is not primarily collected to support product development, often this data is unstructured and even free text based. It is the goal of the Data Fusion Project to look for answers to these issues.

At the XMDU department of Xerox Manufacturing Nederland BV, where the project is executed, the problem is the compilation of European field feedback data about the Xerographic Replaceable Unit (XRU). This XRU is the cartridge that actually makes the prints and is the heart of a printing system (see Appendix I). European field feedback data is necessary to be able to perform analysis on the data and keep track on how the products are performing in the field. This data is available, but it is stored in many different systems which makes it difficult to compile. At this moment North-American XRU data is therefore used for reports as a benchmark for the European XRUs, even though these XRUs are produced in other production sites.

Method
While Xerox has difficulties in finding and merging field feedback data from different sources, the IOP Data Fusion Project is looking for new technologies to merge field feedback data from different sources. Therefore this is typically a problem for which the Data Fusion Project is trying to find answers. This master thesis project examined if Enterprise Search tools are applicable to merge XRU field feedback data from different sources, and can convert potentially useful field feedback data which is stored in the data systems into factual useful data. These Enterprise Search tools should be able to connect to a company’s data systems, develop an index from the available data, and make users able to search through this index. To make the data actually useful, the Enterprise Search tool is combined with an analysis tool to examine if the data can be analyzed automatically. With an analysis tool, the desired values can be combined to develop reports with graphs. If the tools are applicable, it would prove that the product design department can be supported with most recent XRU reports. This resulted in the following Problem Statement:

To what extent can an Enterprise Search environment convert potentially useful field feedback data into factual useful field feedback data in a flexible way and therefore support product design improvement?
Results
First, the channels through which XRU field feedback data returns to Xerox are identified, together with the sources where this data is stored. After examining these channels and their sources, the potentially useful sources to use in this project are selected. Although all available sources contain data which could support product development, not all sources are used in this case. While the focus was on the reproduction of the North-American reports with European data, only the sources which contain the required data are considered.

To check if it is possible to develop the requested XRU reports with the available data at all, and to deliver a direct solution to the XMDU department, an ad hoc way is defined to develop the reports. The systems which contain data from the potentially useful data sources are used for these reports, and their user interfaces are approached to obtain the data manually.

While this way of reporting is time-consuming and quite diffuse, a more flexible and easy way of reporting is desirable. Therefore it is examined if Enterprise Search tools are applicable to merge data from different sources, after which analysis on this data would become easier. While direct access from the Enterprise Search tool to the selected data sources was not possible due to the security regulations at Xerox, this examination is executed in a test environment. After this, an application is developed in Java to combine the different tools. This application combines Apache Solr and a transformation tool with a user-friendly interface (figure 1). The interface is able to show the different indexed dimensions and values in a graphical way, which makes the user able to select the necessary parameters by just clicking boxes. When all needed parameters are selected in the interface and the query is built, results are immediately exported to Excel. After that, the user is able to develop reports using the PivotTable functionality.

This resulted in a final solution (figure 2), called the Enterprise Search and Analysis Tool (ESAT). Although the case is executed in a test environment, a complete and executable solution is developed. ESAT works according to the following steps:
1. Apache Solr processes data from the test databases through content ingestion and content processing and analysis.
2. Solr indexes the data from the different test sources.
3. When the interface is opened, the application makes a graphical view of the indexed dimensions and values.
4. By selecting the boxes in this graphical view, the query is built.
5. The query gets parsed and matched with the index.
6. Search results are automatically transformed and exported to Excel.
7. The PivotTable functionality of Excel is able to develop reports on the data.
When the application was implemented, its applicability could be examined. Reporting with ESAT proved to be possible, while similar reports could be made as in the ad hoc way of reporting. Several technical issues should however be solved to enhance the usability of the application.

**Conclusion**

For this project, the following general conclusions can be drawn:

- Enterprise Search tools are applicable to merge data and can convert potentially useful field feedback data into factual useful information, which is the main issue of this case.
- The developed application seems to be a more flexible solution than a data warehouse or the ad hoc way of merging, analyzing, and reporting. A number of technical issues will need to be fixed however, before the application can be implemented in a real-life situation.

Except for this particular project, the applicability of an Enterprise Search environment can also be reviewed in a broader perspective. With the applicability of Enterprise Search in data fusion, chances for companies are created:

- Data from different systems can be combined, and companies get an overview of the available data. Employees are often not aware of all available data systems. By creating an overview with the Enterprise Search environment, employees can observe all available data which enhances the chance that they will also use the data.
- Data merging with an Enterprise Search environment connects people and departments within a company. By combining data from different systems and making it also available for different departments, employees can obtain new business insights which would otherwise stay unnoticed.
- It is possible to link data from different sources. When the print volume which is stored one system was divided by the parts usage stored in another system, conclusions about a products drum life could be drawn.
Recommendations

Recommendations Xerox

As an Enterprise Search environment proved to be useful in merging data from different sources, it would be a useful solution for the data difficulties at XMDU. It would provide them at least with an overview of all available data, but also reporting and analysis on their data proved to be possible in combination with Enterprise Search.

However, the developed tools are currently in a premature stadium and the application as it is right now would still require a lot of effort to become a workable application. Another point is that within one or two years, Xerox Europe will probably have changed to only using ESAP instead of using all the in-country systems. All European order and failure call data will then be recorded in one system. This would make implementation of Enterprise Search probably more useful, while it will be much easier to implement. There will also be less difference in parameters and naming because of that. Until then, using the ‘ad hoc’ way of reporting will probably be more feasible for Xerox than invest in an Enterprise Search solution.

If all technical issues would have been solved in future, a data fusion application based on Enterprise Search technology could provide major advantages to Xerox. The combination of data from different sources, and even from different departments, could provide a department with new insights. Also the analysis of free text can give insights that would not have discovered otherwise.

Recommendations IOP Data Fusion Project

An Enterprise Search environment proved to be useful in merging data from different sources. It will therefore be useful for the Data Fusion Project to elaborate the functionality of this software further. The developed application also proved to be useful for building the query and exporting the data to Excel. Issues like stated in section 4.4, which are mostly technical issues, should be examined and resolved. If a decent working data fusion application based on Enterprise Search technology can be developed, it will create serious chances for companies.

The PhD students who are participating in the Data Fusion Project can also gain advantage in their work with an Enterprise Search environment. In case of knowledge sharing, this solution can give advantages while an overview of the available data can be retrieved with the application. This could make people be more willing to use this data. Because data from different sources is merged, also data systems from different departments can give employees insight in useful data which would normally stay unnoticed.

In case of linking and analyzing data that is based on free text, an Enterprise Search environment can also gain advantage. Besides databases based on structured data, the Enterprise Search tool is also able to index sources with unstructured and free text data.
III. Preface

This report represents the final assignment in order to achieve my Master of Science degree in Operations Management and Logistics at the Eindhoven University of Technology. The master thesis is conducted at Xerox Manufacturing Nederland BV.

I would like to thank all people from Xerox who helped me in one way or another, and the people from the XMDU department in particular. They made my stay at Xerox a very pleasant one, and they were always willing to help me. Before I knew it six months had passed and I had to hurry to finish my report. As my company supervisor, I would like to thank Ludwig Nooyens for his daily support during my project and for reviewing my reports. I would also like to state a word of thanks to Karl Kurz, who gave me the opportunity to perform a master thesis project in the XMDU group in the first place.

From the university, I would like to thank my first supervisor Ton Weijters for always having a critical look at my work and come up with new insights. Further a special word of thanks to PhD student Joel Ribeiro, who is also participating in the IOP Data Fusion Project. While my project often got quite IT-oriented with the implementation of tools and the programming in Java, he was always willing to help me. Without his help, I definitely did not achieve the same result as I have now. I would also like to thank Lu Yuan as my second supervisor and project leader of the Data Fusion Project, for reviewing my report and come up with some interesting points which made my report stronger.

Roel Kuijpers

Eindhoven, 2011
# Table of Contents

I. Abstract ................................................................................................................................. V
II. Management Summary .......................................................................................................... VII
III. Preface ................................................................................................................................ XI
IV. Table of Contents ................................................................................................................ XIII

Chapter 1: Introduction .............................................................................................................. 1
  1.1 Data Fusion Project ............................................................................................................ 1
  1.1 Company Structure .......................................................................................................... 2
  1.2 Problem Description ......................................................................................................... 2
  1.3 Research design ............................................................................................................... 3
  1.4 Report outline .................................................................................................................. 5

Chapter 2: Field Feedback Data Sources .................................................................................. 6
  2.1 Field feedback channels .................................................................................................. 6
  2.2 Potentially useful field feedback data sources ................................................................ 11
  2.3 Evaluation results .......................................................................................................... 14

Chapter 3: From potential to factual usefulness in an ad hoc way ............................................ 15
  3.1 Data sources ................................................................................................................... 15
  3.2 Developing reports ......................................................................................................... 18
  3.3 Evaluate results .............................................................................................................. 20

Chapter 4: From potential to factual usefulness in a flexible way ........................................... 21
  4.1 Enterprise Search ............................................................................................................ 21
  4.2 Implementation ESAT .................................................................................................... 24
  4.3 Developing reports ......................................................................................................... 31
  4.4 Evaluate results .............................................................................................................. 34

Chapter 5: Conclusion and recommendations ........................................................................... 37
  5.1 Conclusion ...................................................................................................................... 37
  5.2 Recommendations .......................................................................................................... 39

References ................................................................................................................................... 41
Table of figures .......................................................................................................................... 42
Terms and Abbreviations .......................................................................................................... 43
Appendix I: Xerographic principle ............................................................................................ 44
Appendix II: CRUM Parameters XRU ...................................................................................... 46
Appendix III: XRU Field Feedback Systems ........................................................................... 47
Appendix IV: Ad hoc reporting: Xelus and FDRS ................................................................. 48
Appendix V: Ad hoc reporting example: CRUM ................................................................. 49

XIII
Chapter 1: Introduction

This chapter gives an introduction into the project. The project context is explained together with the problem statement and the research questions. An outline of the complete thesis is given to provide an overview.

Companies store more and more data every year. Until the smallest details, information is recorded in a company’s systems. It therefore slowly becomes more and more difficult for companies to keep structure in all this content.

For manufacturing companies, these small details combined and analyzed could give insights that might not be discovered directly. It could say something about how their products are actually performing in the field for example. The focus of this master thesis is therefore on the merging of this kind of content and on how it can contribute to a better product development.

This master thesis originated from collaboration between Xerox Manufacturing Nederland BV and the Eindhoven University of Technology (TU/e), while both parties participate in the IOP Data Fusion Project. This Data Fusion Project is a multi-disciplinary project, executed by a team of researchers from the Eindhoven University of Technology and Groningen University (RUG), working in close collaboration with a few industrial partners.

1.1 Data Fusion Project

The Data Fusion project is founded in 2007 to ‘support product development with prioritized information related to mismatches between product specifications and customer requirements’ (Lu, 2007). The goal is to make field feedback information be available in detail and earlier than currently. By doing this, a product development department can gain advantage while they can react on a product’s deficiency within a shorter time. However, this fast information retrieval is hard today due to four major trends that are identified in industry (Brombacher, 2005):

- Increasingly complex products, becoming available at lower prices ever faster due to new technology.
- Strong pressure on time-to-market
- Increasingly global economy
- Decreasing tolerance of end-users for quality and reliability problems

To meet customer requirements and deal with these increasing and often conflicting trends, the project focuses on five primary sources of feedback information that companies can retrieve from the field (Lu, 2007):

- Test data (obtained during the product creation process)
- Service data
- Helpdesk data
- Trade data
- Internet data (e.g. forums)

Data from these five different sources should be combined in order to get an understanding about how a product is performing in the field. Most of the data is not primarily collected to support product development, so often this data is unstructured and even free text based. It is the goal of the Data Fusion Project to look for answers to these issues. The project is sponsored by the Dutch Ministry of Economic Affairs under the IOP IPCR program (IOP, 2008), and should therefore also justify their results to this instance.
Given its scientific and industrial context, the project requires cooperation between both industrial academic researchers as well as industrial partners. In this context, the Business Information System research group from the Groningen University and the Information Systems and Business Process Design research groups from the Eindhoven University of Technology submitted to this project (Lu, 2010). The Information Systems department of the TU/e, where the author of this report is graduating, contributes in this project by selecting and applying relatively new techniques in the area of data and process mining. The industrial partners can supply the data that is needed for research.

1.1 Company Structure
The company where the project is executed, Xerox, is a globally operating manufacturer of a range of color and black-and-white printers, multifunction systems, photo copiers, digital production printing presses, and related consulting services and supplies. Xerox is headquartered in Norwalk, Connecticut. They exploit their products in about 160 countries and have about 130,000 employees. To give an insight in the financial situation of the company, 2010’s revenue was about 22 billion dollar.

Xerox Manufacturing Nederland BV, which is located in Venray, was the European manufacturing site until about 2005. From then more and more production lines moved to other production sites or production was being outsourced. Currently only a few parts of the printing systems are being produced in Venray. It became the logistic center of Xerox Europe and is responsible for the distribution of machines and spare parts to several European countries at this moment.

The thesis project is executed at the Xerographic Materials Delivery Unit (XMDU) of Xerox Manufacturing Nederland BV. This department is responsible for the development of the Xerographic Replaceable Unit (XRU), which is the cartridge that actually makes the prints and is the heart of a printing system (see Appendix I).

1.2 Problem Description
As explained earlier, companies store more and more business content every year. According to Forrester Research¹, this content volume is even growing at an annual rate of 200%. Also at Xerox, business content is rapidly increasing. This makes it time-consuming for employees to acquire the data they are looking for. At XMDU the problem is the compilation of European field feedback data about the XRU. Data about this product is necessary to be able to perform analysis on the data and keep track on how the products are performing in the field. This data is available, but it is stored in many different systems. For this reason data is difficult to compile.

XMDU employees currently even use North-American XRU data as a benchmark for European field performance reports, while that data is already compiled and therefore easier to acquire. However, North-American and European XRUs are produced on different production sites, which makes it questionable if North-American data is representative for European products.

The reports that the department needs to clarify the performance of the XRU are actually based on simple data. The most important one, the order life graph (figure 3), is for example based on the number of used XRUs and the number of prints that are made with those XRUs. The average number of prints that is made with a XRU should be increasing to conclude that the durability of the XRU gets better. This is important, while the vast majority of the XRUs are used by

---

¹ Forrester Research Inc. is an independent research company that provides advice to global business and technology companies.
companies which have a lease contract with Xerox, and the use of consumables is included in that contract.

The longer customers can print with a XRU, the lower the costs for replenishments will be for Xerox. Because this is such an expensive part, it is important to keep track on how the product performs in the field. If the life of this XRU can be extended with 10% for example, this will involve a huge decrease in costs on XRU replenishments.

Besides the number of prints made by a XRU, also the amount of cycles the drum in the XRU rotated can be used as a measure.

![Figure 3: Order Life (k) graph](image)

While the XMDU department is experiencing problems in finding and compiling recent field feedback data as described above, it is hard to draw conclusions about the current performance of a European XRU. Their goal is therefore to find a way to merge European field feedback data in such way, that reports can be performed easily. This should make it possible to recognize a product’s deficiency within a shorter time span.

### 1.3 Research design

The problem described is typically a problem for which the Data Fusion Project is trying to find solutions. Xerox has difficulties in finding and compiling field feedback data from different sources, and is therefore not able to support product development with useful European XRU data. The IOP Data Fusion Project is on its turn looking for new technologies to merge field feedback data from different sources to support product development. A difference is that in this case mainly structured data will be considered to develop the reports, while the IOP Data Fusion project also focuses on the merging of unstructured data.

Initially, the Data Fusion Project planned to develop a data warehouse within a company. This data warehouse would combine information from the earlier mentioned five sources, if available, and make it available for reporting. In combination with OLAP tools and data and process mining, it would be possible to gain useful information out of these sources within a short time. However, because data is offloaded from the databases and inserted into the data warehouse periodically, a data warehouse is a very rigid and less current solution. The information that is desired in the data warehouse also has to be predefined and adjusted. For example, one operational system feeding data into the data warehouse uses “NL” and “D” to denote the country where the product is sold, while another operational system may use “Netherlands” and “Germany”. To merge such data, these names should be adjusted. Another point is that a product development department might want to be able to search for all relevant documents which concern specific information. For example all e-mails, reports, or forum threads about a problem with a certain XRU type. Retrieving such data is not possible with a data warehouse.

Because of these disadvantages, after some projects in a real environment this plan changed into the idea of implementing a more flexible system. This system should be able to perform a ‘live’ search through the databases and documents from a company’s systems for the latest versions of product information and field feedback. Another point is that the data should not be completely predefined anymore.
Tools that have this functionality are examined in the Literature Study. These so called Enterprise Search tools are linked to a company’s data systems and provide employees with the ability to perform queries and obtain relevant data from these systems. It would become a sort of ‘Google search’, but then through a company’s data systems instead of through the internet. While a ‘live’ search would be performed, most recent data could be retrieved from the systems and product deficiencies could be discovered faster. Further, data is not stored in a new system in this situation and the requested information is not predefined, so it should become a much more flexible system than a data warehouse.

This master thesis project will therefore examine if Enterprise Search tools are applicable to merge XRU field feedback data, and convert potentially useful field feedback data which is stored in the data systems into factual useful data. To make the data really useful, the search tool will be combined with analysis tools to examine if the data can be analyzed automatically. With these analysis tools, the desired parameters can be combined to perform reports with histograms and graphs. If both tools are applicable, product design improvement can be supported with most recent XRU reports.

To mark out the project, the applicability of the tools will be examined by attempting to develop the requested reports as described in the previous section in a more flexible way. Because examining the applicability of the search tool is the main issue, this results in the following problem statement:

*To what extent can an Enterprise Search environment convert potentially useful field feedback data into factual useful field feedback data in a flexible way and therefore support product design improvement?*

In order to find a solution to the problem, this problem statement is split up into multiple research questions. These research questions are already stated in the Research Proposal, and will give the project a structured approach in order to perceive its solution.

**Which available field feedback data sources contain potentially useful data to support product design development?**

To start the project, the channels through which XRU field feedback data returns to Xerox need to be identified, together with the sources where these channels report their data. When these different channels and sources are identified, the reported parameters in the different channels are examined. The potentially useful sources to use in this project can be selected after this.

*To what extent can an Enterprise Search environment be used to convert this potentially useful data into factual useful information?*

First, an ad hoc way will be defined to obtain the data that is needed to perform the requested reports manually. This will provide Xerox with a direct solution to their current problem and this will examine if it is possible at all to perform the reports. A tutorial to make the reports in this way will be developed and results will be evaluated. After that a more flexible way with the help of an Enterprise Search tool will be examined. The architecture of the new system will be developed, and the search tool will be implemented together with the databases. When it is not possible to perform these tests in a real situation, a test environment with test databases can be constructed to perform the tests. Finally, a decision will be made on the applicability of this tool.
To what extent can analysis tools contribute to the simplification of reporting and analysis of this information?

Only merging structured data about a specific product might not be enough to draw a conclusion on, because combining a pile of data will still conclude in a pile of data. Therefore also the potential of data analysis tools will be examined to test their possible contribution to the reporting and analysis of the data. Such analysis tools are able to process the data by plotting sets of values against each other. Therefore it should be possible to construct histograms and graphs about current performance like the ones shown in the previous section. After that a decision will be made on the usefulness of these tools.

1.4 Report outline

This first chapter served as a general introduction into the problem of this case and showed the proposed solution that is going to be examined. It therefore serves as a basis for the entire project. The subsequent chapters broadly follow the outline of the research questions. Chapter 2 describes all channels through which field feedback data returns to Xerox, together with the sources where these channels report their data. When these different channels and sources are identified, the reported parameters in the different channels are examined. The potentially useful systems are then selected to use in this project. In chapter 3 an ad hoc way is defined to obtain the data that is needed to perform the requested reports. A tutorial to make reports in this ad hoc way is developed to provide Xerox with a direct solution to their current problem. Chapter 4 examines a more flexible way to perform these reports with the help of a search tool and an analysis tool. The architecture of the new system will be developed, and the tools will be implemented together with the databases and a user interface. At last, chapter 5 draws a conclusion on the findings in the project and gives recommendations to both Xerox and the IOP Data Fusion Project.

Due to confidentiality issues, some text and figures are adjusted or deleted in this version of the report.
Chapter 2: Field Feedback Data Sources

Chapter 2 describes the channels through which XRU field feedback data returns to Xerox, together with the sources where these channels report their data. The reported parameters in the different channels are examined to decide if the available sources are potentially useful to support product design development.

In the highly competitive market Xerox is in, it is important to be very efficient in developing your products. At each stage of the product development process, reliability-related information should therefore be generated, obtained, and updated. Reliability assessments can be made periodically, but this requires high quality feedback and control loops in order to achieve efficiency of the total business chain (Magniez, 2007).

For this reason it is important to get information out of all contact there is between the company and the customer, and to use this information also. In this case, information about how a XRU is performing in the field should be obtained. While a customer has to work with the product on a daily basis, he should have useful information either in a direct or in an indirect way.

With the XRU, a direct way could be the reporting of a problem with the print quality by a customer. In this way, the customer points out the problem directly.

An indirect way could be that for a specific type of machine, customers order a new XRU when it has only performed for about 200,000 drum cycles, while the cartridge’s hard stop is at 600,000 cycles. Such factors can be observed when analysis is done on data that is stored in databases.

Therefore the first important issue is to obtain a proper view on the following factors:

- Through which channels field feedback arrives at Xerox
- Which systems store this data
- What data is stored by these feedback channels
- Which of these provide potentially useful field feedback data
- Can we get access to the sources of these field feedback data

These five factors are elaborated below.

2.1 Field feedback channels

Through the field feedback channels, field feedback data gets inserted into the data systems. (Magniez, 2007) describes the field feedback process at Xerox after doing research at that company in 2007. He depicts the complaint handling for a customer in figure 4.

Next to the feedback channels described in this book, some more feedback channels can be defined. Note that only channels which include field feedback information about the XRU are discussed.

There are two different types of XRUs. The Engineer Replaceable Unit (ERU) is a cartridge that has to be replaced by a service engineer and the Customer Replaceable Unit (CRU) is a cartridge that can be replaced by the customer. ERUs are placed in Legacy products and CRUs are placed in Office products.
**Welcome Center**

The Welcome Center is the first contact point for customers. Customers who have a problem with their Xerox machine or have a need for replenishments can reach this Welcome Center to contact a Xerox employee.

**Machine failures**

If a customer experiences problems with a Xerox product, the call center of the Welcome Center can be contacted. There are three types of problems a customer can have:

- Failures which are auto-detected by the machine itself and which are identified by a corresponding machine code. Such failures should lead to a correct treatment by the Welcome Center.

- Failures based on customer description. For such failures, the accuracy of the analysis is more random, which could lead to an excess of replacement.

- Failures which have not been anticipated, and for which the Welcome Center has therefore received no instructions regarding its possible treatment. For such failures, the correct advice cannot be provided. Until the failure is reproduced in the laboratory and the “Fault Tree Chart” is updated, replacement will be suggested. This “Fault Tree Chart” is a step-by-step tutorial for Welcome Center employees with possible solutions to a customer’s problem.

Certain data is recorded in this process. For Office products calls are recorded in ESAP and for Legacy products calls are recorded in Voyageur. ESAP should replace all other European ERP-systems in the next few years, but currently this system is only being used for Office products. The Welcome Center agents ask for the Customer’s serial number, after which the customer’s company name, phone number, and address is retrieved. After that the fault code is recorded. Negative points are that no link is made with older calls from the same customer, so the system has no memory. Also there is a conflict in interests in this process. The call center strives for treating as many customers a day, while a XRU engineer wants to identify the root cause of a problem. Another point is that the failure description is dependent on a customer’s perception. This makes it hard for a Welcome Center employee to interpret the real problem. The use of standard ‘Fault Tree Charts’ makes the complaint handling a bit narrow minded, and makes it harder to solve particular customer issues.
Table 1: Machine failures

<table>
<thead>
<tr>
<th>Feedback Channel</th>
<th>Data recorded in</th>
<th>Recorded data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Welcome Center</td>
<td>ESAP (Office products)</td>
<td>Date</td>
</tr>
<tr>
<td></td>
<td>Voyager (Legacy products)</td>
<td>Customer serial number</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Company name</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Phone number</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Address</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Fault code (if available)</td>
</tr>
</tbody>
</table>

**Replenishments**

With Office products a customer has to replace the replenishments by himself. To order replenishments the Welcome Center can be contacted by phone or through a website. Office order data is recorded in the ESAP system. Company data is again retrieved when the customer gives its serial number. The customer is also asked to give the print count meter readings.

Table 2: Replenishments

<table>
<thead>
<tr>
<th>Feedback Channel</th>
<th>Data recorded in</th>
<th>Recorded data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Welcome Center</td>
<td>ESAP</td>
<td>Date</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Customer serial number</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Company name</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Phone number</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Machine serial</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Print count meter readings</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Ordered products</td>
</tr>
</tbody>
</table>

**Service Engineer**

Another information flow is perceived when a service engineer visits a customer to provide service. This can be either to solve a machine failure or to replace an ERU. Again the data about Office products (CRUs) is recorded in ESAP. A company that uses CRUs is only visited by a service engineer when there is a problem. Therefore the normal use of CRUs will not be noticed through this way.

In case of Legacy products (ERUs), data about machine failures and replenishments is inserted in different in-country systems (ICS).

**In-Country Systems**

Xerox deploys activities in 23 European countries. Because Legacy products are not recorded in one European system like ESAP, every country has its own solution. Some countries collaborate with one system to record Legacy data, while other countries have their own system. At this time 15 out of these 23 databases can be used for further processing. As shown in table 3 below, 9 out of these 15 countries use XSAP. While these systems will always be mentioned together, they will be given the name ‘In-Country Systems’ or ICS.
Table 3: In-country systems

<table>
<thead>
<tr>
<th>System</th>
<th>Country</th>
</tr>
</thead>
<tbody>
<tr>
<td>XSAP</td>
<td>Portugal</td>
</tr>
<tr>
<td></td>
<td>Holland</td>
</tr>
<tr>
<td></td>
<td>Ireland</td>
</tr>
<tr>
<td></td>
<td>Austria</td>
</tr>
<tr>
<td></td>
<td>Switzerland</td>
</tr>
<tr>
<td></td>
<td>Denmark</td>
</tr>
<tr>
<td></td>
<td>Finland</td>
</tr>
<tr>
<td></td>
<td>Norway</td>
</tr>
<tr>
<td></td>
<td>Sweden</td>
</tr>
<tr>
<td>GSAP</td>
<td>Germany</td>
</tr>
<tr>
<td>Coresap</td>
<td>Belgium</td>
</tr>
<tr>
<td>SBS</td>
<td>Great Britain</td>
</tr>
<tr>
<td>Corail</td>
<td>France</td>
</tr>
<tr>
<td>IMS</td>
<td>Spain</td>
</tr>
<tr>
<td>FWSS</td>
<td>Italy</td>
</tr>
</tbody>
</table>

For replenishments with Legacy products, a service engineer replaces the parts and therefore also places the order. This can be done through e-mail (named Kanamail), Fax, or Phone. The engineer has to fill in the code of the type of failure, and the part that has been replaced. Faults can be made when a wrong part number or code is registered. Also the engineer can make a bad diagnosis or replace the wrong part.

Table 4: Engineer visits: ICS

<table>
<thead>
<tr>
<th>Feedback Channel</th>
<th>Data recorded in</th>
<th>Recorded data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Service Engineer</td>
<td>In-Country systems</td>
<td>Date</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Customer serial number</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Company name</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Phone number</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Machine serial</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Print count meter readings</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Fault code</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Ordered products</td>
</tr>
</tbody>
</table>

**Eureka**

To provide service engineers with decent repair manuals, a tool called iDoc is used. This tool is installed on their service laptop and can be consulted when they are performing service at a customer. With the Eureka tool, iDoc is complemented with a few extra functions. It is for example possible for service engineers to add tips about a specific product or service action for other service engineers. These tips can also be seen as field feedback, because they handle real experiences of service engineers with the products.

Table 5: Engineer visits: Eureka

<table>
<thead>
<tr>
<th>Feedback Channel</th>
<th>Data recorded in</th>
<th>Recorded data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Service Engineer</td>
<td>Eureka system</td>
<td>Tips about installation and service tasks in free text</td>
</tr>
</tbody>
</table>
**Returned products**

The product design team performs three types of post mortem analysis:

**Analysis conducted on returned products (warranty reclaim)**
When the laboratory investigates the units, several scenarios are likely to occur. In some cases, the service engineer who sent the unit describes the failure. The laboratory team succeeds to make the same diagnostic, and the root cause is known. Based on statistical analysis, the severity of the failure is estimated and corrective actions are prioritized. In other cases, the service engineer cannot describe the complaint, and the laboratory team is not able to reproduce the failure. The situation then becomes more complex: this inability for the team to reproduce the failure may be due either to a mistake by service engineer (e.g. engineer has replaced the wrong part), or to a mismatch in the perception of failure between the customer and the team, or between the service engineer and the team. The analysis and fault reporting are mainly based on experience from the laboratory team. Each time a failure is diagnosed, a report is brought back to the relevant department (design, manufacturing, service organization).

<table>
<thead>
<tr>
<th>Feedback Channel</th>
<th>Data recorded in</th>
<th>Recorded data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Laboratory Analysis</td>
<td>File servers</td>
<td>Complete post mortem reports</td>
</tr>
</tbody>
</table>

**CRUM**
A more recent technology is the use of a chip in the CRU, called the CRUM (CRU Module). Most cartridges have this chip nowadays, which records a wide variety of machine information during the CRUs lifetime. A small amount of CRUs with various lifetimes returns to the laboratory monthly. After this, the data from the CRUMs of these CRUs is downloaded and inserted in a database. When this is done, analysis on this data can give a good insight about how a product functions in the field. A list with all parameters is stated in Appendix II.

<table>
<thead>
<tr>
<th>Feedback Channel</th>
<th>Data recorded in</th>
<th>Recorded data</th>
</tr>
</thead>
<tbody>
<tr>
<td>CRUM</td>
<td>CRUM Database</td>
<td>CRU type</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Installation dates</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Drum cycle count</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Print count</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Many other parameters (see appendix II)</td>
</tr>
</tbody>
</table>

**Analysis is carried out on non-warranty products**
Such post-mortem analysis is accomplished on products which may have reached different lives. The analysis is usually performed after the CRUMs are read and aims at having a general assessment of the design robustness. This is important for new products, especially if it is about an upgraded version of the products, as it allows a better understanding of the product design.

<table>
<thead>
<tr>
<th>Feedback Channel</th>
<th>Data recorded in</th>
<th>Recorded data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Laboratory Analysis</td>
<td>File servers</td>
<td>Complete post mortem reports</td>
</tr>
</tbody>
</table>
Xerox Community

In the third quarter of 2010, a new system is installed that also generates field feedback data, namely the Xerox Community. This is a forum where problems with Xerox products can be discussed.

Currently there are three employees responsible for the forum, one community manager and two moderators. The moderators do not provide solutions to the customers, because the meaning of the system is a customer to customer interaction. Because the system is still in an early development phase, some Xerox employees who work closely with product development teams are currently answering some questions. This should only be temporary and should slowly change into a customer to customer interaction. The Xerox community functions completely separate from the welcome center described above. There is no system that analyzes the unstructured data, for example to note regularly occurring problems.

Table 9: Xerox Community

<table>
<thead>
<tr>
<th>Feedback Channel</th>
<th>Data recorded in</th>
<th>Recorded data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Xerox Community</td>
<td>XC Database</td>
<td>Date</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Name</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Free text with opinions, questions, and solutions given by Xerox customers</td>
</tr>
</tbody>
</table>

Planning Systems

In an indirect way, Xerox’ planning systems are also field feedback channels. Information about ordered consumables can give useful information about the performance of products. For example when there is an unexpected increase in demand for a specific product. Planning system Xelus contains information on the number of XRUes that are monthly shipped to a particular country or district.

Table 10: XelusParts

<table>
<thead>
<tr>
<th>Feedback Channel</th>
<th>Recorded data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Planning systems</td>
<td>Date</td>
</tr>
<tr>
<td></td>
<td>Product Family</td>
</tr>
<tr>
<td></td>
<td>Product</td>
</tr>
<tr>
<td></td>
<td>Area</td>
</tr>
<tr>
<td></td>
<td>Country</td>
</tr>
<tr>
<td></td>
<td>District</td>
</tr>
<tr>
<td></td>
<td>Ordered consumables</td>
</tr>
</tbody>
</table>

2.2 Potentially useful field feedback data sources

All systems mentioned in the previous section can be useful to support the XRU product development. Some sources contain unstructured and free text based information, which could be useful directly or would need advanced analysis to analyze greater amounts of unstructured data. Other sources contain more structured data and need further processing and analysis to be useful. For this project it is decided to examine if the reports as described in chapter 1 can be developed in a more flexible way. Therefore the potentially useful field feedback data sources are the sources that contain the parameters needed for these reports. The potentially useful sources for this project will be identified, together with their attributes. An entity-relationship model is presented to show relationships among the attributes.
Data Sources

The sources which contain information that is necessary to perform the requested reports are stated below. ESAP and the in-country systems contain data about customer accounts, product orders, failure calls and engineer visits. Therefore data about product orders and print count meter reads can be retrieved. XelusParts includes disbursement data, which gives information about product orders. The CRUM database contains data retrieved from CRUM reads. This database contains information about building sites, drum cycle counts, machine failures etcetera.

The Laboratory Analysis file servers are not included, because they contain complete reports that do not need further processing. The Xerox Community data is not included while there is only free text in the fields. Therefore the choice is made to look at the following sources:

Table 11: Potentially useful data sources

<table>
<thead>
<tr>
<th>Source</th>
<th>Location</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESAP</td>
<td>England</td>
<td>Relational database with Office data</td>
</tr>
<tr>
<td>XSAP (ICS)</td>
<td>India</td>
<td>Relational database with Legacy Consumables</td>
</tr>
<tr>
<td>GSAP (ICS)</td>
<td>Germany</td>
<td>Relational database with Legacy Consumables</td>
</tr>
<tr>
<td>CoreSAP (ICS)</td>
<td>Belgium</td>
<td>Relational database with Legacy Consumables</td>
</tr>
<tr>
<td>SBS (ICS)</td>
<td>Great Britain</td>
<td>Relational database with Legacy Consumables</td>
</tr>
<tr>
<td>Corail (ICS)</td>
<td>France</td>
<td>Relational database with Legacy Consumables</td>
</tr>
<tr>
<td>IMS (ICS)</td>
<td>Spain</td>
<td>Relational database with Legacy Consumables</td>
</tr>
<tr>
<td>FWSS (ICS)</td>
<td>Italy</td>
<td>Relational database with Legacy Consumables</td>
</tr>
<tr>
<td>CRUM</td>
<td>US</td>
<td>Relational database with CRU-information read from its CRUM</td>
</tr>
<tr>
<td>XelusParts</td>
<td>US</td>
<td>Relational database about consumable forecasts and orders</td>
</tr>
</tbody>
</table>

Data Views

The data views describe the different attributes. In the data views the attribute name, type, and description are given. Only relevant data for this case is included. For the CRUM Table data view see Appendix II.

Table 12: Data views

<table>
<thead>
<tr>
<th>Table</th>
<th>Source</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Customer Account</td>
<td>ESAP/ICS</td>
<td>Customer information</td>
</tr>
<tr>
<td>Disbursements</td>
<td>XelusParts</td>
<td>Information XRU disbursements</td>
</tr>
<tr>
<td>Order (Office/Legacy)</td>
<td>ESAP/ICS</td>
<td>Information about product orders</td>
</tr>
<tr>
<td>Office Failure Call</td>
<td>ESAP/ICS</td>
<td>Information about Office product failure calls</td>
</tr>
<tr>
<td>Engineer visit (Office/Legacy)</td>
<td>ESAP/ICS</td>
<td>Information about service engineer visits</td>
</tr>
<tr>
<td>CRUM</td>
<td>CRUM DB</td>
<td>Information downloaded from the CRU Modules</td>
</tr>
</tbody>
</table>

Table 13: Data view: Customer Account

<table>
<thead>
<tr>
<th>Attribute Name</th>
<th>Attribute Type</th>
<th>Attribute Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Customer ID</td>
<td>Number</td>
<td>Unique customer serial number</td>
</tr>
<tr>
<td>Company name</td>
<td>Text</td>
<td>Company name of customer</td>
</tr>
<tr>
<td>Phone number</td>
<td>Number</td>
<td>Phone number of the customer</td>
</tr>
<tr>
<td>Address</td>
<td>Text</td>
<td>Address of the customer</td>
</tr>
</tbody>
</table>
An entity-relationship model (figure 5) is depicted to show the relationships between the different data views. It describes the data model of all identified data views on a high level of abstraction.
Data Access

Direct read-only access to the different sources would be most desirable when examining the applicability of the Enterprise Search tool, because it could then be tested in a real-life situation. However, retrieving data was not easy and access to the data sources was limited due to security regulations. If raw data should be retrieved for testing the applicability of Enterprise Search software, direct links between the data sources and the Enterprise Search tool will also not be possible. Data will then only be available in a flat file format and not with the structure as described above. The databases only contain separate data tables like a spreadsheet page.

To ensure the files will still approach reality, the files will be slightly adjusted. This is done by adding columns with information about parent data tables. Because these flat files need to be used, it is unavoidable to develop a test environment. This is however no problem for the execution of this project.

2.3 Evaluation results

All sources described in section 2.1 could support product development. However, while they cannot all contribute to the development of the requested reports, not all of these sources are suitable for this particular project.

The development of the requested reports out of the selected sources will first be done in a manual and ad hoc way. This is done to deliver a direct solution to Xerox and to check if it is possible to develop the requested XRU reports at all with the available data. This will also ensure a certain result of this project.

When it is possible to perform the requested reports out of the selected data, a more flexible way of reporting is examined. This is done by obtaining similar data with the help of Enterprise Search software. Report results developed with the Enterprise Search environment can then be compared with the reports of the ad hoc way.
Chapter 3: From potential to factual usefulness in an ad hoc way

This chapter focuses on the potentially useful data from the sources selected in chapter 2. It examines the possibility to convert this data into factual useful information in an ad hoc way, and to develop the requested XRU reports manually.

At this moment, the potentially useful data from the sources selected in chapter 2 is not optimally used to support product development. To deliver a direct solution to the XMDU department and to check if it is possible to develop the requested XRU reports at all, the possibility to develop these reports manually is examined. If this is achievable, the possibility to develop the same reports in a more flexible way can be examined.

3.1 Data sources

To develop the requested reports out of the four defined data sources, access to their systems is needed. Direct access to the sources is difficult due to security issues. These systems are therefore complemented with a user interface to make employees able to work with the data. These user interfaces will be used in this chapter to develop the reports in an ad hoc way. Appendix III depicts a website with an overview of the available XRU Field Feedback systems, which is developed by the author of this report to give the XMDU employees an overview of the available XRU Field Feedback systems.

Because ESAP and the in-country systems are merged in a data warehouse especially for reporting and analysis, only three user interfaces need to be considered. These systems are the CSO Focus Suite Product Performance, the XelusParts planning system and the CRUM database.

**CSO Web Portal – Focus Suite Product Performance**

To make data from ESAP and the ICS useful for reporting and analysis, the Xerox Customer Service Organization (CSO) Web Portal (figure 6) is developed. This is a web-based interface which lets employees use data from the Field Data Reporting System (FDRS). FDRS is a data warehouse in which information from ESAP and the in-country systems is combined.

![Figure 6: CSO Web Portal - Reporting and Analysis](image)

Instead of only combining data from these sources in a data warehouse, the data is also partly processed before stored. Consumable order data is for example stored by serial number and order date. After that this data is linked with the service call history.
The sources as shown in figure 7 are inserted in the system. In ESAP, data about Office products is available, while the in-country systems contain Legacy data.

As a solution for this project, it is decided to test the applicability of an Enterprise Search tool instead of developing a data warehouse. In this situation however, data from a data warehouse is used to develop the ‘ad hoc’ reports. In one way this is because it is just more practical, but it is also interesting to see the advantages and disadvantages of a data warehouse.

One point is that data is uploaded to this data warehouse with different intervals. ESAP and XSAP are for example updated on a daily basis, while other systems are updated weekly or monthly. Because of this, in the data warehouse values can change during a month. If odd results are retrieved from the data warehouse, it is not possible to see clearly what data is behind the value and see why the value could be different than expected. An advantage is that data is already partly processed, which makes it easier to develop reports on the data.

**XelusParts Planning System**

The XelusParts system is a Java-based application (figure 8) that is used by the order planning employees. It contains the forecasted XRU disbursements for future months and the real disbursements for past months. If a type of XRU is selected, these disbursements can be obtained for the entire XRU Family of that type. For every type the amount of products that went to a certain country or region is defined.

---

**Figure 7: FDRS and its sources**

**Figure 8: XelusParts planning system**
**CRUM Database**

The CRUM Data application (figure 9) is a web-based application which contains data that is directly downloaded from the CRU Module. The product family can be defined together with the build and read site. The time interval can then be selected on build, read, 1st install or last install date. After that the data can be directly exported to Excel.

![Figure 9: CRUM Database user interface](image-url)
3.2 Developing reports

Developing reports based on data from the sources described in the previous section can be done in different ways. In this situation this means either a more current way, with the Focus Suite and Xelus combined, or a more precise and elaborated way, with the CRUM data.

To be able to compare the reports, the same products and time span should be used in every case. It is therefore chosen to look at the following XRU data:

Table 18: Selected products for reporting

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Area</td>
<td>Xerox Europe (XE)</td>
</tr>
<tr>
<td>Product Family</td>
<td>A</td>
</tr>
<tr>
<td>Machine Family</td>
<td>4110F &amp; 4112F</td>
</tr>
<tr>
<td>XRUs</td>
<td>013R00653, 013R00652, 013R00646, 013R00640, 013R00639, 013R00635, 013R00610, 013R90145, 675R20890</td>
</tr>
<tr>
<td>CRUMs</td>
<td>539E07630_01, 539E08260_00, CD3P58200_00, CDC300496C00</td>
</tr>
<tr>
<td>Time interval</td>
<td>July 2010 - December 2010</td>
</tr>
</tbody>
</table>

Focus suite and Xelus

The more current reports can be made by combining data from the Focus Suite with FDRS data warehouse and Xelus. This is therefore a first remark that can advocate for a more flexible way to merge data from different sources, while at this moment both user interfaces need to be approached. In the FDRS data warehouse all order and service call data for Office and Legacy products is combined. A wide variety of reports can therefore be performed in the Focus Suite Web Portal:

Table 19: Focus Suite reporting options

<table>
<thead>
<tr>
<th>CSO Focus Suite reporting options</th>
</tr>
</thead>
<tbody>
<tr>
<td>Product Performance</td>
</tr>
<tr>
<td>Account Performance Reporting</td>
</tr>
<tr>
<td>Engineer Performance Reporting</td>
</tr>
<tr>
<td>Call Causal &amp; Top Parts Usage</td>
</tr>
<tr>
<td>Active Machines In Field</td>
</tr>
<tr>
<td>Service history by Serial Number Selection</td>
</tr>
</tbody>
</table>

For the requested XRU reports, the *Product Performance* and *Parts and Call Causal Performance* options are most important. These options give an overview of how a machine family performs in the field, for example by showing the total volume (k) prints. This print volume can be obtained, because by linking the consumable order data and the service call history, the orders can be linked with the meter readings.
The *Parts and Call Causal Performance* contains information about the shipped quantity of a specific XRU type to a specific country in a specific month. This can be divided with the total volume of prints that are made with that specific XRU type with the same restrictions to develop reports.

Besides the parts usage from the Focus Suite, also the disbursements from the Xelus system can be used. This system gives the actual values of the disbursements to every European country. There is a difference in values between the parts usage and disbursements data, so if these values are divided by the total print volume also both Drum Life graphs will differ (figure 10). If both ways are depicted in a graph, it can be assumed that the value should be between the two lines.

![Figure 10: FDRS/XelusParts graph in an ad hoc way](image)

**Print Volume reliability**

(Part of this chapter is deleted due to confidentiality reasons)

Another point is the question of what a good average print volume per XRU is. Instead of a number of prints, usually a specific number of charge cycles is defined for a XRU. Most of those charge cycles will make prints, but a part of these cycles will be performed when the machine is starting a print job or ending a print job. Machines that print more long jobs will therefore have less of these ‘dead’ cycles, while those that have mainly small print jobs will have more ‘dead’ cycles. When an average life in prints is stated, this can therefore only be used as a guide to compare between months.

Additionally, a part of the prints is made in A3 format, which means a double size of a normal A4 print. Such a print job is only counted as one print. With the 4110F&4112F family as shown in the graph above for example, it is known that about 20% of the prints are made in A3 format. It should therefore be assumed that the real average life is about 350 kp + 20% = 420 kp.

In Appendix IV a tutorial is shown which describes how to develop these reports using data from the Focus Suite system and XelusParts system.

**CRUM Report**

Reports can also be performed by using data from the CRUM Database. This database is completely based on the information provided by the chip that is built in the CRU, the CRU-Module (CRUM). The module delivers a lot of parameters (see Appendix II) that can be very useful when evaluating a CRU. Parameters like the drum cycle count and print count are important, but also the amount of prints made in A3 and A4 or the fault codes can give very useful information.

Disadvantage is that these CRUMs can only be read when the CRU returned to a Xerox Laboratory after his lifetime. This means that the time between introducing a new product and the first reliable CRUM reads of this product is about 6 months. The first months after introducing a
new CRU, only bad performing CRUs returned to Xerox. Therefore the number of drum cycles becomes realistic after about half a year.

An example report is conducted and shown in Appendix V. Some of the important slides are stated below.

Figure 11: CRUM graphs in an ad hoc way

3.3 Evaluate results

It can be concluded that with the above-mentioned data it is possible to develop the requested reports in two ways, either in a more current and a more precise way.

If more current reports are desirable, a combination of the Focus Suite system and the XelusParts system is best to use. Data from most systems is updated on a daily basis, while from other systems data is updated weekly or monthly. Because the data is also partly processed, developing reports is not a very time-consuming activity. Negative point is the usability of the Total Volume (k) Prints. This volume can therefore only be used as a guide to compare between months, instead of comparing against a real reference. Because the data is already partly processed, errors in the data cannot be noticed directly out of the Focus Suite.

If more precise reports are desirable, the CRUM Database can be used to perform reports. Positive point about this database is that a wide variety of data is recorded very precisely on each CRUM, so realistic reports can be produced. Negative point is that the data is less current. Before CRUM data is in the database, the XRU already reached his end of life, had to be sent back to Xerox, data had to be downloaded from the CRUM and had to be sent to the CRUM Database administrator. Another point is that the data that is retrieved is still raw data. This is positive in a way that errors in the data can be easily noticed and deleted, but developing reports out of this data requires quite some effort.

Although it is possible to develop the reports, this ad hoc way of reporting is a time-consuming task. Every system should be approached separately, after which data should be combined and analysis. This situation is therefore not desirable. A more current and flexible way of reporting would be more adequate and is examined in the next chapter.
Chapter 4: From potential to factual usefulness in a flexible way

Chapter 3 described how to develop the requested reports out of the potentially useful data in an ad hoc way. While a more flexible way is desirable, this chapter examines if an Enterprise Search environment might be useful to convert this data into factual useful information in a more flexible way.

Chapter 3 showed how the requested reports can be developed in an ad hoc way. This proved to be possible, although it is a time-consuming and diffuse task. Different data systems need to be approached manually through their interfaces, after which the data should be exported to Excel. Only then it is possible to perform analysis on the data. A more flexible way should therefore be more desirable.

As described in the research questions, the applicability of an Enterprise Search tool will be examined to test if such tool can convert the potentially useful data into factual useful information. The tool should be able to merge data from different sources, which makes it possible to obtain data from different sources by using only one application. This would make it much less time-consuming to develop reports.

First the term Enterprise Search is discussed, after which the search tool that is going to be used in this case is described. Then the implementation of the test databases, the Enterprise Search tool, and the user interface is described. After that, the results of this implementation are evaluated.

4.1 Enterprise Search

At the beginning of the last decade, tools to search within a company’s data systems emerged. With these so called Enterprise Search tools, it is possible to search through several systems like a company’s databases, CRM systems, and e-mails. Using this software could save a lot of time on searching and could increase productivity. (Feldman, Sherman, 2003) state that companies with poor quality search within their enterprise are facing high costs of not finding information, both in the form of lost opportunities and through lost productivity.

(Hawking, 2004) describes that the term Enterprise Search includes:
- Any organization with text content in electronic form
- Search of the organization’s external website
- Search of the organization’s internal websites (its intranet)
- Search of other electronic text held by the organization in the form of email, database records, documents on file shares etcetera.

While there are not sufficient scientific articles available about Enterprise Search, part of the used articles are retrieved from independent research companies, commercial sites, and Wikipedia. The authors of the Wikipedia articles are checked for their knowledge of Enterprise Search, to ensure that the information is reliable. Daniel Tunkelang for example, who is one of the editors of the Enterprise Search Wiki site, is the co-founder of a well-known Enterprise Search company and did a lot of scientific research in the field of Information Retrieval.

Principle

When discussing Enterprise Search, people often expect the same principle as they are used to with an internet search. In practice, Enterprise Search is different from searching a digital library or the web in many aspects (Broder, Ciccolo, 2004). For example in the perception of what a
good search result is. On the internet, a good result is achieved when you find results with a high relevancy. For companies this is different. A good result is then usually achieved when the exact data that you were looking for is found.

In this project, the Enterprise Search tool is used in a slightly different way than it normally is. The difference in this case is that an employee is not searching for information about a subject or for a specific document, but wants to retrieve structured data. After a query, all related values should be returned to ensure that reports are complete if the data is processed. If some data is missing in the search result, graphs will depict an incorrect view of the current situation.

Compared to a data warehouse, in an Enterprise Search environment data is not copied and stored in an entire new system. The only content that is stored is the index, which is developed out of the connected data sources. The phases in which an Enterprise Search solution functions is stated below.

**Phases in Enterprise Search**

Enterprise Search software usually follows five steps in its functionality. These are content ingestion, content processing and analysis, indexing, query parsing, and matching. Although the exact principle can vary between the different vendors, they are mostly comparable with the phases that are described below.

1. **Content ingestion**
   The first step is the ingestion of content, because data must be accumulated before it can be searched. This requires knowledge of where the information is located, and how to access these repositories. After that, content ingestion is possible in two ways, namely through push or pull. If a push model is used, a source system connects to the search engine and pushes new content directly to its Application Programming Interfaces (API’s). The API’s make sure the different systems can communicate. This model is used when real-time indexing is important. When a pull model is used, a web crawler or a database connector that belongs to the search software gathers the content from the sources. With certain intervals, the connector searches through the connected data sources to look for new, updated or deleted content.

2. **Content processing and analysis**
   The Enterprise Search software must be able to search through a lot of different formats or document types. To make this content searchable, the content processing phase processes the incoming documents to plain text using document filters. It is also often necessary to normalize content in various ways to improve accuracy measures like recall or precision. Tokenization, which is the process of extracting words, phrases, symbols or other elements out of a text, is applied to split the content into tokens. It is also common to normalize tokens to lower case to provide case insensitive search, normalize accents to provide better recall.

3. **Indexing**
   To search through large amounts of text quickly, a text has to be indexed and converted into a format that will let you search it rapidly. This conversion process is called indexing, and its output is called an index. This can be considered as a data structure that allows fast random access to words stored inside it. It can be seen as the index at the end of a book, which lets you quickly locate pages that discuss certain topics. The index data structure stores a mapping from content, such as words or numbers, to its locations in a database file, or in a document or a set of documents. In this way quick lookups are possible without storing the full text of the document.

---

may for example contain the dictionary of all unique words in the text as well as information about ranking and term frequency.

4. **Query parsing**

When a user inserts a query into the software, the query parser performs some of the same processing functions as the index processing but can also include query-side only processing. To receive a high quality search result, it is important to weed out information that is dated, irrelevant or duplicated. The query parser therefore contains automated filtering functions. With functions as filtering, faceting, federated search and query suggesting, a user can further refine a query.

5. **Matching**

The software compares the query with the terms in the index, and returns its results by referencing source documents that match. Some software is also able to differentiate between a current and a ‘cached’ result. This way you can not only view a document as it is right now, but also how it was when it was indexed.

**Apache Solr**

After examining different Enterprise Search solutions in the Literature Study, the choice is made to use Apache Solr as the Enterprise Search tool. This tool is chosen because it is the most commonly used open source Enterprise Search tool, through which comprehensive tutorial and wiki sites are available. This makes the tool very approachable. It is also free to use, while it is an open source tool. The software is not only commonly used by companies that want to implement Enterprise Search by themselves, but also by several commercial Enterprise Search developers as a base for their own Enterprise Search software solution. These software solutions are then usually complemented with an extensive user interface which enhances the workability of the software.

Apache Solr is an open source Enterprise Search platform, which is built on top of text search engine Apache Lucene. It uses the Lucene Java search library for full-text indexing and search. Solr is written in Java and runs as a standalone full-text server. It is able to index large amounts of documents. Search results are given in XML format, so they can be processed further easily, because the structure of the data keeps maintained. Solr gives the following features:

<table>
<thead>
<tr>
<th>Table 20: Solr features</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Apache Solr features</strong></td>
</tr>
<tr>
<td>Full text search</td>
</tr>
<tr>
<td>Faceted search</td>
</tr>
<tr>
<td>Dynamic clustering</td>
</tr>
<tr>
<td>Distributed search</td>
</tr>
<tr>
<td>Search results clustering</td>
</tr>
<tr>
<td>Extensible through plug-ins</td>
</tr>
</tbody>
</table>

In this case, the tool should be able to search through a company’s databases. While Solr’s Data Import Handler (DIH) can be used to gather content from relational database management systems (RDBMS), that functionality is examined.
4.2 Implementation ESAT

To implement the Enterprise Search and Analysis Tool (ESAT), first the test databases will be defined and implemented. After that, Apache Solr is implemented and connected to the databases. A user interface will then be developed to simplify working with the tools. It will make the user able to export the search results to Excel, after which reports can be developed.

As discussed earlier, in a most realistic environment the Enterprise Search tool would be directly linked to a few databases. As described in section 2.2, direct links turned out to be unrealistic for this project due to security issues. For this reason it is decided to build a test environment, and to perform all tests within this environment.

Test Databases

In this test environment the Enterprise Search tool will be linked to three test databases. Data from ESAP and the ICS are just as in the ad hoc solution in chapter 3 combined. The database is called the FDRS Test Database. Combining these systems makes the test environment less realistic, but due to the security regulations and the great amount of different systems it was hardly possible to obtain sample data from all different in-country systems. Another reason is the addition of the ‘print volume’ algorithm in FDRS as described in section 3.2. The negative points of this decision are taken into account in this thesis, because testing the applicability of an Enterprise Search tool is more important than simulating the situation at Xerox in detail.

The other databases, the CRUM test database and the Xelus test database, are filled with data tables and data from the real databases, although still in flat file format as explained in section 2.2.

It must be possible to combine data from the different sources. While it needs to be ensured that the information in the three databases has an overlap, the databases need to be filled with the same product type data from the same period. The data selected is the same as the data used in chapter 3. This makes it possible to compare the results. The figure below gives an overview of how the test databases should be linked to Apache Solr.

![Diagram](image12.png)

Figure 12: Connection test databases with Apache Solr

Flat files from the data sources defined in chapter 2 are obtained to put into the test databases. It is decided to construct all databases on a Xerox server, while other options were not possible due to security issues. *Microsoft SQL Server 2008 R2 Management Studio* is used to construct and manage the databases.

<table>
<thead>
<tr>
<th>Database</th>
<th>Data Table</th>
<th>Server</th>
</tr>
</thead>
<tbody>
<tr>
<td>FDRSTest</td>
<td><em>PrintVolume</em></td>
<td>(deleted due to confidentiality reasons)</td>
</tr>
<tr>
<td></td>
<td><em>PartUsage</em></td>
<td></td>
</tr>
<tr>
<td>XelusTest</td>
<td><em>Disbursements</em></td>
<td></td>
</tr>
<tr>
<td>CRUMTest</td>
<td><em>CRUM</em></td>
<td></td>
</tr>
</tbody>
</table>
The sample data in the tables contains information about the A cartridges. Information about this cartridge family is however denoted in three different ways. In the PrintVolume table data is based on the machine family name in which the A cartridges run. The PartUsage and Disbursements tables are based on the XRU numbers, in this case the 013R00653 Family. In the CRUM table, data is based on the CRUM type numbers. All type numbers are shown in table 22.

Table 22: Product data in test databases

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Area</td>
<td>Xerox Europe (XE)</td>
</tr>
<tr>
<td>Product Family</td>
<td>A</td>
</tr>
<tr>
<td>Machine Family</td>
<td>4110F &amp; 4112F</td>
</tr>
<tr>
<td>XRUs</td>
<td>013R00653, 013R00652, 013R00646, 013R00640, 013R00639, 013R00635, 013R00610, 675R20890</td>
</tr>
<tr>
<td>CRUMs</td>
<td>539E07630_01, 539E08260_00, CD3P58200_00, CDC300496C00</td>
</tr>
<tr>
<td>Time interval</td>
<td>July 2010 - December 2010</td>
</tr>
</tbody>
</table>

Table 23 describes the field names that the different tables contain. Some field names represent the same sort of value, but are not named in the same way in the different tables. Because Apache Solr needs a unique ID for every table row, a column with a unique ID for every row is added to every table. In the CRUM database, more than 30 different dimensions are available. In this test database however, only relevant dimensions are included.

Table 23: Fields in the different data tables

<table>
<thead>
<tr>
<th>Disbursements</th>
<th>PrintVolume</th>
<th>PartUsage</th>
<th>CRUM</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>ID</td>
<td>ID</td>
<td>ID</td>
</tr>
<tr>
<td>Disbursements</td>
<td>Parameter</td>
<td>Parameter</td>
<td>Parameter</td>
</tr>
<tr>
<td>XRU Type</td>
<td>Product Family</td>
<td>Part</td>
<td>Part</td>
</tr>
<tr>
<td>Area</td>
<td>Area</td>
<td>Month</td>
<td></td>
</tr>
<tr>
<td>Country</td>
<td>Country</td>
<td>Year</td>
<td></td>
</tr>
<tr>
<td>District</td>
<td>Region</td>
<td>Month</td>
<td>Amount</td>
</tr>
<tr>
<td>Month</td>
<td>Year</td>
<td>Production Site</td>
<td></td>
</tr>
<tr>
<td>Year</td>
<td>Year</td>
<td>Measure</td>
<td>Product Family</td>
</tr>
<tr>
<td>Measure</td>
<td>Amount</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Shipment Area</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Enterprise Search tool**

As described in section 4.1, in this case Apache Solr will be used as Enterprise Search tool. The Enterprise Search tool used is Apache Solr 1.4.1, while this is currently the latest version of Apache Solr. The Enterprise Search tool should function like depicted in figure 13.
Through Content Ingestion and Content Processing and Analysis, Apache Solr is able to develop an index out of the connected databases. When a query is sent after that, the parser handles the text in the same way as the data in the index. After that the query is matched with the data in the index, and search results are returned to the user interface.

When installing Apache Solr, a few files are important. Figure 14 depicts the files that are most important in this case.

- `data-config.xml` has to be created to make the links with the data tables and to secure the indexing restrictions.
- `solrconfig.xml` is the general configuration file of Solr and can be obtained from the Solr example folder. After that it has to be adjusted for this case.
- `schema.xml` is the file where the indexing configuration is located.
- The `lib` folder needs to be created manually and the .jar file has to be downloaded and put into the folder. Further, all other plugins for Solr need to be put in this folder.
- If Solr is running, the developed index is put in the folder `data/index`.

Apache Solr is able to search through a wide variety of documents like XML, Excel, Word, or PDF. In this case however, only the indexing of databases is considered. To do this, Solr’s Data Import Handler (DIH) can be used to gather content from relational databases management systems (RDBMS). To make Solr able to search through databases, it has to be sure that this Data Import Handler is inserted in `solrconfig.xml`. The following lines have to be inserted:
Besides the Data Import Handler, a connector is needed to make Solr able to connect to the database. Because Apache Solr is based on Java, this should be done with a JDBC (Java Database Connectivity) link. In this case the server runs on MS SQL Server, so the driver sqljdbc4.jar should be downloaded and copied into the lib directory. The lib directory needs to be specified manually.

The driver should be specified in the data-config.xml file (see figure 16 below), together with the URL and if necessary a username and a password of the server. Also the databases that need to be indexed must be defined in data-config.xml, together with the field columns that Solr has to index.

To make the tables more uniform, the name of the column can be changed in the index which Solr develops. Field column XRU TYPE in the 013R00653Family table and field column Part in the PartsUsage table for example both contain XRU type. It is therefore easier to name them both Product. When querying for Product in the search interface, Solr will refer to both columns.

This defining of the available data tables and field columns restricts the flexibility of the Enterprise Search tool however. The field columns that must be indexed by Solr are described in table 23. Therefore only for the implementation of the Disbursements data table is depicted in figure 16 below. For the other three data tables, the definition must be stated in the same way under <entity>.

When data-config.xml is specified, the columns which need to be indexed by Solr also have to be specified in schema.xml as shown below. If field names occur in every database, required should be set to true. If not, Solr should not try to search for the field name in every database. To prevent errors, required should therefore be set to false.
To make Apache Solr able to index every row in the databases, these rows should all have a unique character. Because in our sample data a unique character was not always present, a column ID is added to every table. This ID column is therefore the uniqueKey.

![Figure 17: Field name definition in schema.xml](image1)

When querying for words, not every employee will type a parameter in exactly the same way. In case of the dimension Country, one will type Germany while another employee will type GERMANY. To prevent that no search results will come up, the LowerCaseFilterFactory handles every query for a certain field type as a lowercase query. This plugin enhances usability when searching for text.

![Figure 18: Lowercase Filter Factory in schema.xml](image2)

Another issue encountered in this case was that one data table denoted Switzerland as country, while another data table denoted Swiss. This problem should be solved with some sort of synonym list in Solr, which states that Switzerland and Swiss are one and the same country. Also synonyms like Zwitserland or Suisse, or common type errors, should probably be inserted in such synonym list. The implementation of such list again restricts the flexibility of an Enterprise Search environment, but will enhance usability enormously. The development of such list however was not realistic within the time span of this project.

After these preparatory steps, the index can be built out of the four selected data tables. By clicking on ![http://localhost:8983/solr/dataimport?command=full-import](http://localhost:8983/solr/dataimport?command=full-import), Apache Solr will perform a full import of the data from the different sources and develop an index. This index is stored in the data\index folder.
**ESAT environment**

When the data is indexed, it is possible to perform the queries in the user interface which is delivered by Apache Solr. However, because the search results have to be transformed and exported to Excel also, some steps would still need to be performed with a Java application. It is therefore more logical to develop a single application which is able to perform all these steps.

This application is developed with the help of PhD student Joel Ribeiro, who also participates in the IOP Data Fusion Project. The interface of this application should simplify all steps and make them user friendly. It is created in Java with Netbeans IDE software and works as a stand-alone application.

The interface is able to show the different indexed dimensions and values in a graphical way, which makes the user able to select the necessary parameters by just clicking the boxes. When all needed parameters are selected in the interface, the query is built and results are immediately exported to Excel. After that, the user is able to develop reports using the PivotTable functionality. Figure 19 below gives an overview of how the complete test environment works:

1. Apache Solr processes data from the test databases through content ingestion and content processing and analysis.
2. Solr indexes the data from the different test sources.
3. When the interface is opened, the application makes a graphical view of the indexed dimensions and values.
4. By selecting the boxes in this graphical view, the query is built.
5. The query gets parsed and matched with the index.
6. Search results are automatically transformed and exported to Excel.
7. The Pivot Table functionality of Excel is able to develop reports on the data.

![Figure 19: Functionality ESAT environment](image)

As described above, to develop the query the user interface is complemented with a graphical view of all dimensions and values that are indexed. In this way it is not only possible to type the query in the search bar, but dimensions and values can also be selected in the box view.
The box view on the left side of the application is built from the Solr index when the application started. In the boxes, the different dimensions are stated together with the underlying values. When searching for certain data, the user is able to select the relevant boxes in the box view. If a user has difficulties in finding a certain value, the name of the needed value (e.g. ‘Germany’) can be typed into the query box and the relevant box will enlighten. Because the user can see what dimensions and values can be selected in this interface, it enhances the usability.

The application consists out of 3 screens. On the first screen (figure 20), dimensions which need all their values in the query can be selected. Country could be selected for example, because there is only European data in the test databases and European data needs to be obtained. When a box is selected, it will appear in the list on the right side of the interface. If the wrong box is selected, it can be deleted again by clicking on the black X.

The second screen (figure 21) lets the user select other values, like the value 013R00653 for dimension Product. When all needed values are selected, the next screen can be approached by clicking Next.

When all values are fixed, also a measure needs to be selected. When a box is selected as a measure, it will appear with a red color.
After this, the application has combined all restrictions and has built the query. The Next button can be clicked, after which Excel automatically opens with all selected data sorted in columns. The PivotTable functionality needs to be selected to process the data. Fields need to be dragged into the table after which graphs can be constructed easily.

4.3 Developing reports

The main question of this master thesis is if an Enterprise Search environment is able to convert potentially useful data into factual useful data and if it makes reporting and analysis more easy. To examine this, it will be tried to develop the requested reports using the Enterprise Search environment, retrieving the same variables as done in the ‘ad hoc’ way of reporting stated in chapter 3.

**Focus Suite and XelusParts**

First the development of the FDRS and XelusParts report will be examined. The following dimensions and values need to be selected for the query.
Table 24: Values needed for query FDRS/XelusParts reporting

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Area</td>
<td>XE</td>
</tr>
<tr>
<td>Country</td>
<td>Austria, Belgium, Denmark, Finland, France, Germany, Greece, Ireland, Italy, Netherlands, Norway, Portugal, Spain, Sweden, Switzerland, UK</td>
</tr>
<tr>
<td>Month</td>
<td>July, August, September, October, November, December</td>
</tr>
<tr>
<td>Year</td>
<td>2010</td>
</tr>
<tr>
<td>Parameters</td>
<td>Print Volume, Parts Usage, Disbursements</td>
</tr>
</tbody>
</table>

When all values are selected in the user interface, the following query is automatically built by the application and will be sent to Solr:

```
```

Figure 24: Query built by application

When the query is executed and matched with the index, the search results are automatically exported to Excel and sorted in the right columns. After this, the user has to insert the PivotTable manually and put the right parameters in the right part of the table.

In the Active Field tab, it can be selected if the measures in the table should be ‘Average of amount’ or ‘Sum of amount’ for example. If the fields are inserted the right order, in the Calculations tab Fields, Items, & Sets can be selected to add a formula to the table. This is needed because the Print Volume column needs to be divided by both the Disbursements column and the Part Usage column.

As shown in figure 25, a Drum Life graph as in the ad hoc way can be developed easily after that. The measures are similar in both the ad hoc way (section 3.2) and the new more flexible way of reporting.

Figure 25: FDRS/XelusParts graph in PivotTable
CRUM Report

Also for the CRUM Database, the possibility of developing reports with ESAT is examined. The following dimensions and values need to be selected for the query:

Table 25: Values needed for query CRUM reporting

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Product</td>
<td>539E07630_01, 539E08260_00, CD3P58200_00, CDC300496C00, CD3P55600_00, CDC300716C00, CD3P35700_00</td>
</tr>
<tr>
<td>Production Site</td>
<td>Country X</td>
</tr>
<tr>
<td>Month</td>
<td>July, August, September, October, November, December</td>
</tr>
<tr>
<td>Year</td>
<td>2010</td>
</tr>
<tr>
<td>Parameters</td>
<td>CRUM</td>
</tr>
</tbody>
</table>

The following query is then automatically built in the application and is sent to Solr:

```
(Year:2010) AND (Product:CD3P58200_00 OR Product:CD3P55600_00 OR Product:539E07630_01 OR Product:CD300496C00 OR Product:539E08260_00 OR Product:CD3P35700_00 OR Product:CD300716C00) AND (ProductionSite:X) AND (Parameter:CRUM) AND (Month:October OR Month:August OR Month:July OR Month:November OR Month:December OR Month:September)
```

Figure 26: Query built by application

The figure below shows the result which is obtained from ESAT. These results are similar with the results which can be obtained by getting data out of the database with the CRUM web interface (page 20).

Figure 27: CRUM graph in PivotTable
4.4 Evaluate results

As shown in the previous section, it is possible to develop the requested XRU reports out of the search results retrieved with the application. During the implementation and testing of ESAT however, several difficulties are encountered. It can therefore be said that Enterprise Search in general and the ESAT environment both have major advantages as well as some serious issues. Some of them can be seen as implementation issues, while other issues are clear usability issues. Some issues can also be seen as both, while usability often coincides with implementation. A usability issue is often caused by an implementation issue and can be solved by changes in the implementation. In table 26, the advantages and issues are summarized.

**Enterprise Search**

**Advantages**

Enterprise Search tools are able to connect to different sources and combine the data from these sources in an index. Compared to a data warehouse, in an Enterprise Search environment data is not copied and stored in an entire new system. While no separate system is needed to store the data, a lot of costs can be saved on the maintenance of a new system. While different data sources can be combined, it is also possible to combine data from different departments. This could provide employees with new insights. Another point is that only one user interface is needed, because the data is combined. If reports should be developed like in this case, this could be very time saving. Apache Solr develops an index out of the connected databases, and the search results Solr gives come straight from the index. If the index is updated regularly, it would be ensured that the retrieved data is always current.

**Issues**

The first issue is the access to the different data sources. In case of Xerox, the useful data sources are Xerox Europe’s most extensive data systems. The Enterprise Search tool should be connected to these systems in order to be able to index data from these sources. Due to security regulations, it is hard to gain this access. In the test environment of this thesis this is no big issue, while testing the applicability of the search tool was more important than copying the situation at Xerox. In a real environment however, there must be a connection to the real databases. This implies huge security regulations that should be implemented together with the implementation of Enterprise Search. Not all data in the data systems should be indexed for example, because data could be confidential for own employees. Further, not all indexed data should be accessible for every employee. Employees could therefore need some sort of access level. All this will involve a huge operation. Another point is that in Apache Solr, all databases and data tables with their corresponding columns need to be predefined before they can be indexed. This is an important point, while this restricts the flexibility of the Enterprise Search tool seriously. Because of this it should be known already what data should be indexed. The predefined columns can also cause issues when changes in the databases are made. For example when a column is added or deleted in a data table. In this case, an added column would just result in an index where this column is not taken into account. However, deleting a column causes errors, while Solr is searching for a column which is not available anymore. It might be that other Enterprise Search tools do not have this issue.

Next is that different systems are connected to each other, using a different approach with a different structure and a different naming. Dimensions and values which have the same meaning can be denoted in a different way. Different naming will be shown in the application, like the use of ‘Switzerland’ in one system and ‘Swiss’ in another system, but it is important to select the right one if you need data from a particular database. Such issue could be solved by some sort of
synonym list, but then again you should know what needs to be indexed and which values are in that index to be able to develop a synonym list.

**ESAT**

**Advantages**

With the ESAT environment as described in section 4.3, all tools needed for this examination are combined in one single application. This enhances usability enormously, while it is less time-consuming than executing every step manually. The graphical box view which is developed shows employees which data is available in a clear way. After that, by typing the name of a value just by selecting the different boxes, the query will be built. Finally, the application automatically exports the search results to Excel, after which Excel automatically opens the file. The dimensions and values selected in the application are now given in columns. With the *PivotTable* functionality, developing a graph is very easy.

**Issues**

The user interface is developed to make building the query more easy and straightforward. While this application is still in a premature state, lot of improvements are possible. Just as with Enterprise Search, the fact that different systems use a different approach with a different structure and a different naming can cause issues. Dimensions and values which have the same meaning can be denoted in a different way. The application is not able to solve this yet. The selection of months and years is currently not really practical for example. Months are not depicted in a logical order in the tree map, and it is also hard to select a certain time interval. When needing data from July 2010 to June 2011 for example, it is needed to select all months and both 2010 and 2011. When data is exported to Excel after that, data for two complete years is retrieved instead of data for one year. This states that not all dimensions are suitable to select in the graphical view of the user interface. For this first version of this application however, these issues are manageable. Another point of improvement is the visualization of some structure, or ontology, in the tree map. If a value of a certain data table is selected, like ‘Germany’ for dimension ‘Country’, it would be desirable if from dimension ‘District’ all districts that belong to Germany would enlighten. These enhancements were however not realistic within the time span of this thesis. Scalability is the last point of improvement. In this test environment, data tables were small and therefore the boxes in the interface could be selected easily. In a real situation, the index could become enormous and it could be that it is not possible anymore to click on the boxes. This should be taken into account.
All advantages and issues are summarized in the table below:

Table 26: Advantages and issues

<table>
<thead>
<tr>
<th>Enterprise Search</th>
<th>ESAT</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Advantages</strong></td>
<td></td>
</tr>
<tr>
<td>Ability to merge data from different sources</td>
<td>All tools in one application</td>
</tr>
<tr>
<td>Development of an index, so no separate system is needed</td>
<td>Graphical view with boxes</td>
</tr>
<tr>
<td>Only one application needed to approach sources</td>
<td>Changes in index are automatically adjusted in the box view</td>
</tr>
<tr>
<td>Current data can be used</td>
<td>Easy building of the query</td>
</tr>
<tr>
<td>Data from different departments can be combined</td>
<td>Automatically export to Excel</td>
</tr>
<tr>
<td><strong>Implementation issues</strong></td>
<td></td>
</tr>
<tr>
<td>Access to data sources</td>
<td></td>
</tr>
<tr>
<td><strong>Usability issues</strong></td>
<td></td>
</tr>
<tr>
<td>Predefinition dimensions of the sources</td>
<td>Scalability</td>
</tr>
<tr>
<td></td>
<td>Selecting boxes is not practical for all dimensions (years and months)</td>
</tr>
<tr>
<td><strong>Both</strong></td>
<td></td>
</tr>
<tr>
<td>Different structure and naming in different sources</td>
<td>Different structure and naming in different sources</td>
</tr>
<tr>
<td>Changes in databases</td>
<td>No visual structure between values (ontology)</td>
</tr>
</tbody>
</table>

There are some issues that should be solved in future, but while they are all technical details, they are manageable for further projects. We can conclude with the fact that an Enterprise Search environment is able to merge field feedback data, and also convert potentially useful field feedback data into factual useful data. With the help of the application this data can also be put into Excel, after which reports on the data can easily be developed.
Chapter 5: Conclusion and recommendations

Chapter 5 draws an overall conclusion on the project and gives recommendations to both Xerox and the IOP Data Fusion Project.

After the examinations in chapter 3 and chapter 4, it is possible to draw a conclusion on the applicability of the Enterprise Search and Analysis Tool. Because there are two main stakeholders in this project, a recommendation for Xerox as well for the IOP Data Fusion Project is given.

5.1 Conclusion

Project conclusion

This particular project was divided into three research questions to give the report a structured approach. The project conclusion can therefore be based on these research questions.

Which available field feedback data sources contain potentially useful data to support product design development?

The project started with this research question. The different channels through which XRU field feedback data returns to Xerox are identified, together with their corresponding data sources. The data required for developing the requested reports proved to be available, together with a great amount of other potentially useful XRU field feedback data. Negative point is that from the FDRS system, the reliability of the print volume values is questionable. This is one of the most important parameters for the requested reports. In the CRUM database this value is reliable, because the information is directly downloaded from the chip in the XRU.

To what extent can an Enterprise Search environment be used to convert this potentially useful data into factual useful information?

To check if the potentially useful data can be used to develop the requested reports, an ad hoc way to develop these reports manually is examined. While the selected data sources are complemented with a user interface to make employees able to obtain data from these sources, these interfaces are used. It proved to be possible to retrieve the data and to develop the desired reports.

Because developing the reports in this ad hoc way is quite time-consuming and diffuse, a more flexible way to develop the same reports would be more desirable. The main issue of this master thesis was therefore the question if an Enterprise Search environment can be used to convert this potentially useful data into factual useful information. To examine this, Enterprise Search tool Apache Solr is implemented together with a user-friendly interface.

While a direct connection to the different data sources was impossible due to security regulations, everything had to be built in a test environment with data tables in flat file format. These tables are adjusted to ensure this environment is still approaching reality.

The Enterprise Search environment proved to be able to combine data from various sources. An overview of all available data is retrieved, which can enhance business insight. Merging data is therefore successful.
To what extent can analysis tools contribute to the simplification of reporting and analysis of this information?

While only merging data does not prove if data is actually useful, it is examined if an analysis tool can contribute to the simplification of reporting and analysis. The developed user interface provides a clear graphical way to view the available dimensions and values. When all values are selected in this interface, the query is automatically built and search results are automatically exported to Excel. The PivotTable functionality proved to make it easy for employees to develop reports out of the data after that. Therefore the analysis tool combined with the application can give a substantial contribution to the simplification of reporting and analysis of the field feedback data. While the application is still in a premature stadium however, a lot of improvements are possible.

The analysis of the data after merging has some issues, because to a certain amount it is needed to know what you want to analyze, which makes the Enterprise Search environment move to a data warehouse environment. With adjustments to the developed user interface, as described in section 4.4, these issues can be minimized.

To what extent can an Enterprise Search environment convert potentially useful field feedback data into factual useful field feedback data in a flexible way and therefore support product design improvement?

The general problem statement, as described above, can be answered in a positive way. It can be concluded that Enterprise Search tools are applicable and can convert the potentially useful field feedback data into factual useful data, which is the main issue of this case. With an analysis tool, it is even possible to develop reports out of that data in a very simple way. Although a couple of issues will need to be fixed, ESAT seems to be a more flexible solution than a data warehouse or the ad hoc way of merging, analyzing, and reporting.

Data Fusion conclusion

Except for this particular project, the applicability of an Enterprise Search environment can also be reviewed in a broader perspective. While the tools are able to connect to different systems and merge data, chances for companies are created and insights which would normally not have been discovered can be obtained.

First of all, while data from different systems can be combined, companies get an overview of the available data. Employees are often not aware of all available data systems. As also noted at Xerox, one employee is for example familiar with system A and C, while another employee only knows system B and C. By creating an overview with the Enterprise Search environment, employees can observe all available data which enhances the chance that they will also use the data.

Another point is that data merging with an Enterprise Search environment connects people and departments within a company. Different departments within a company often use their own system. While the focus in these systems is on the usability for their own department, other departments are not likely to use these systems. By combining data from different systems and making it also available for different departments, departments can obtain new business insights which would otherwise probably stay unnoticed.

A good example is the XelusParts system which was connected to the Enterprise Search tool in this project. This planning system is usually only used in Xerox’ distribution centers. When this system is accessible for the product development department, the disbursement data can be used to check variation in parts usage for a certain area. This brings new insights to the product development department.
This case also showed that it is possible to link data from different sources. When the print volume which is stored in one system was divided by the parts usage stored in another system, conclusions about a product's drum life could be drawn.

In this particular project only the merging of structured data is tested. In the IOP Data Fusion project however, also advanced techniques are being developed to analyze free text. In case of Xerox, these techniques could enhance insight in ‘free text’ sources like the Xerox Community and the complete post-mortem reports.

5.2 Recommendations

**Recommendations Xerox**

As an Enterprise Search environment proved to be useful in merging data from different sources, it would be a useful solution for the data difficulties at XMDU. Combining different data sources, whether or not from different departments, would definitely support their product development. It would provide them at least with an overview of all available data, but also reporting and analysis on their data proved to be possible in combination with Enterprise Search.

However, the developed tools are currently in a premature stadium and the application as it is right now would still require a lot of effort to become a workable application. Another point is that within one or two years, Xerox Europe will probably have changed to only using ESAP instead of using all the in-country systems. All European order and failure call data will then be recorded in one system. This would make implementation of Enterprise Search probably more useful, while it will be much easier to implement. There will also be less difference in parameters and naming because of that. Until then, using the ‘ad hoc’ way of reporting will probably be more feasible for Xerox than invest in an Enterprise Search solution.

If all technical issues would have been solved in future, a data fusion application based on Enterprise Search technology could provide major advantages to Xerox. As discussed above, the combination of data from different sources, and even from different departments, could provide a department with new insights. Also the analysis of free text, which is a part of the Data Fusion Project, can give insights that would not have discovered otherwise. A good example is the Xerox Community. This forum, which can be used for customer to customer communication, can give very useful information about problems customers have with Xerox products.

**Recommendations IOP Data Fusion Project**

An Enterprise Search environment proved to be useful in merging data from different sources. It will therefore be useful for the Data Fusion Project to elaborate the functionality of this software further. The developed application also proved to be useful for building the query and exporting the data to Excel. Issues like stated in section 4.4, which are mostly technical issues, should be examined and resolved. If a decent working data fusion application based on Enterprise Search technology can be developed, it will create serious chances for companies.

The PhD students who are participating in the Data Fusion Project can also gain advantage in their work with an Enterprise Search environment.

In case of knowledge sharing, this solution can give advantages while an overview of the available data can be retrieved with the application. This could make people more willing to use this data. Because data from different sources is merged, also data systems from different departments can give employees insight in useful data which would normally stay unnoticed.

In case of linking and analyzing data that is based on free text, an Enterprise Search environment can also gain advantage. Besides databases based on structured data, the Enterprise Search tool is also able to index sources with unstructured and free text data. Data from a forum can be indexed for example, after which the free text analysis could extract information from this data.
Another point is that in the Literature Study, also Content Analytic tools are briefly discussed. These tools are able to interpret and analyze unstructured data and free text. It could be useful to examine these types of software to gain ideas for future Data Fusion applications.
References


<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>ESAT user interface</td>
<td>VIII</td>
</tr>
<tr>
<td>2</td>
<td>ESAT environment</td>
<td>IX</td>
</tr>
<tr>
<td>3</td>
<td>Order Life (k) graph</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>Complaint handling at Xerox in 2007</td>
<td>7</td>
</tr>
<tr>
<td>5</td>
<td>Entity-relationship model</td>
<td>14</td>
</tr>
<tr>
<td>6</td>
<td>CSO Web Portal - Reporting and Analysis</td>
<td>15</td>
</tr>
<tr>
<td>7</td>
<td>FDRS and its sources</td>
<td>16</td>
</tr>
<tr>
<td>8</td>
<td>XelusParts planning system</td>
<td>16</td>
</tr>
<tr>
<td>9</td>
<td>CRUM Database user interface</td>
<td>17</td>
</tr>
<tr>
<td>10</td>
<td>FDRS/XelusParts graph in an ad hoc way</td>
<td>19</td>
</tr>
<tr>
<td>11</td>
<td>CRUM graphs in an ad hoc way</td>
<td>20</td>
</tr>
<tr>
<td>12</td>
<td>Connection test databases with Apache Solr</td>
<td>24</td>
</tr>
<tr>
<td>13</td>
<td>Apache Solr functionality</td>
<td>26</td>
</tr>
<tr>
<td>14</td>
<td>Apache Solr folders</td>
<td>26</td>
</tr>
<tr>
<td>15</td>
<td>Data Import Handler in solrconfig.xml</td>
<td>27</td>
</tr>
<tr>
<td>16</td>
<td>Definition sources in data-config.xml</td>
<td>27</td>
</tr>
<tr>
<td>17</td>
<td>Field name definition in schema.xml</td>
<td>28</td>
</tr>
<tr>
<td>18</td>
<td>Lowercase Filter Factory in schema.xml</td>
<td>28</td>
</tr>
<tr>
<td>19</td>
<td>Functionality ESAT environment</td>
<td>29</td>
</tr>
<tr>
<td>20</td>
<td>Screen 1 user interface</td>
<td>30</td>
</tr>
<tr>
<td>21</td>
<td>Screen 2 user interface</td>
<td>30</td>
</tr>
<tr>
<td>22</td>
<td>Screen 3 user interface</td>
<td>31</td>
</tr>
<tr>
<td>23</td>
<td>PivotTable functionality</td>
<td>31</td>
</tr>
<tr>
<td>24</td>
<td>Query built by application</td>
<td>32</td>
</tr>
<tr>
<td>25</td>
<td>FDRS/XelusParts graph in PivotTable</td>
<td>32</td>
</tr>
<tr>
<td>26</td>
<td>Query built by application</td>
<td>33</td>
</tr>
<tr>
<td>27</td>
<td>CRUM graph in PivotTable</td>
<td>33</td>
</tr>
<tr>
<td>28</td>
<td>Charging</td>
<td>44</td>
</tr>
<tr>
<td>29</td>
<td>Exposure</td>
<td>44</td>
</tr>
<tr>
<td>30</td>
<td>Development</td>
<td>44</td>
</tr>
<tr>
<td>31</td>
<td>Transfer</td>
<td>44</td>
</tr>
<tr>
<td>32</td>
<td>Xerographic process</td>
<td>45</td>
</tr>
</tbody>
</table>
Terms and Abbreviations

API  Application Programming Interfaces
CRM  Customer Resource/Relationship Management
CRU  Customer Replaceable Unit
CRUM  Customer Replaceable Unit Module
CSO  Customer Service Organization
DIH  Data Import Handler
ERP  Enterprise Resource Planning
ERU  Engineer Replaceable Unit
ES  Enterprise Search
ESAP  European SAP
ESAT  Enterprise Search and Analysis Tool
FDRS  Field Data Reporting System
HTML  HyperText Markup Language
ICS  In-Country Systems
IDC  International Data Corporation
IOP  Innovatiegericht Onderzoeksprogramma
IPCR  Integral Product Creation and Realization
IS  Information Systems
iSRVE  Integrated Xerox Services Reporting Vision Environment
IT  Information Technology
JDBC  Java Database Connectivity
OLAP  Online Analytical Processing
RDBMS  Relational Database Management System
RUG  Rijksuniversiteit Groningen
SQL  Structured Query Language
TU/e  Eindhoven University of Technology
UML  Unified Modeling Language
URL  Uniform Resource Locator
XC  Xerox Community
XE  Xerox Europe
XMDU  Xerographic Material Delivery Unit
XRU  Xerographic Replaceable Unit
XSAP  Xerox SAP
Appendix I: Xerographic principle

Step 1) Charging. The drum surface is charged to a constant electrostatic voltage. This is done with a corotron, which consists of a tungsten wire that is partly surrounded by a metal shield. The voltage difference between the wire and the shield is brought to a level that is high enough to cause gas discharge. After that the charged particles charge the drum electrostatically.

Step 2) Exposure. After the charging, a laser illuminates the drum surface on the spots that correspond to ‘black’ on the image to be printed. In this way the image to be printed is ‘written’ on the drum surface. Since the drum coating is photoconductive, the electrostatic charge leaks away on that spot. Therefore the image is on the drum as an electrostatic charge pattern, also called latent image.

Step 3) Development. The magnetic toner and its additions are attached to a magnetic roll to form a toner brush. The magnetic roll is biased to a voltage that is in between $V_{\text{high}}$ and $V_{\text{low}}$. For the black areas, the electrostatic voltage on the drum is lower than the magnetic roll voltage and the charged toner particles are attracted to the drum by the electric field (= developer field). For the white background areas, the electrostatic drum voltage is higher than the magnetic roll voltage and the resulting electric field (= cleaning field) repels the toner particles from the drum. As a result there is a developed toner image on the drum.

Step 4) Transfer. Now the paper is brought against the drum, with the back of the paper electrostatically charged. This electrostatic transfer charge attracts the toner particles from the drum surface to the substrate.

Step 5) Detack. Due to the electrostatic forces caused by the transfer charge on the back of the substrate, the substrate is tacked to the drum. To strip the substrate from the drum, the transfer charge on the back of the substrate is partly neutralized. This is achieved by another corotron, the detack corotron.

Step 6) Erase. After the process described above, the drum still contains electrostatic charge and residual toner. This electrostatic charge is removed by illuminating the photoreceptor with a LED array so that all the remaining charge leaks away.

Step 7) Cleaning. In this last step all residual toner is removed from the drum surface. The cleaning is achieved by setting a polyurethane cleaner blade against the drum, which scrapes the toner from the drum.
Figure 32: Xerographic process
Appendix II: CRUM Parameters XRU

Content deleted due to confidentiality reasons
Appendix III: XRU Field Feedback Systems

A website with links to the XRU field feedback systems is developed by the student to give the XMDU employees an overview of these sources. The North-American field feedback data environment, iSRVE, is also added to the site to give employees the possibility to compare results.
Appendix IV: Ad hoc reporting: Xelus and FDRS

Content deleted due to confidentiality reasons
Appendix V: Ad hoc reporting example: CRUM

Content deleted due to confidentiality reasons