Data Classification in Product Data Management

Master Degree Project in Industrial Informatics
One year, Master Level 30 ECTS
Spring term 2013

Author: Iman Morshedzadeh
Department of Technology and Society
University of Skövde
Supervisors: Tehseen Aslam, Florian Siegmund
Examiner: Amos Ng
Declaration

Submitted by Iman Morshedzadeh to the University of Skövde for master in Automation Engineering in School of Technology and Society

September 2009

I hereby certify that all material in this dissertation which is not my own work has been identified and that no work is included for which a degree has already been conferred on me.

Signature:

Iman Morshedzadeh

__________________.
Acknowledgement

I want to thank my supervisors of the University of Skövde Mr. Aslam and Mr. Siegmund and my supervisor in Volvo Cars Engine, Mr. Jenefeldt and Mr. Windolf, for their support, availability and confidence throughout this project, and beside my supervisors, I would like to thank my examiner, Professor Amos Ng for his helpful and beneficial comments.

This Master thesis is developed for Volvo Cars Engine, Skövde, Sweden, and I am thankful for all the support that the employees have given me.

I especially thank my wife for her support throughout this project.
Abstract

This report is about the product data classification methodology that is useable for the Volvo Cars Engine (VCE) factory's production data, and can be implemented in the Teamcenter software. There are many data generated during the life cycle of each product, and companies try to manage these data with some product data management software. Data classification is a part of data management for most effective and efficient use of data. With surveys that were done in this project, items affecting the data classification have been found. Data, attributes, classification method, Volvo Cars Engine factory and Teamcenter as the product data management software, are items that are affected data classification. In this report, all of these items will be explained separately. With the knowledge obtained about the above items, in the Volvo Cars Engine factory, the suitable hierarchical classification method is described. After defining the classification method, this method has been implemented in the software at the last part of the report to show that this method is executable.
# Table of contents

Abstract ............................................................................................................................. III  
Table of contents ................................................................................................................ IV  
Table of figures .................................................................................................................. VI  
List of abbreviations ........................................................................................................ IX  
1 Introduction .................................................................................................................... 1  
  1.1 Background ................................................................................................................ 1  
  1.1.1 Product data management (PDM) ........................................................................ 1  
  1.1.2 Volvo Cars Engine factory and Teamcenter software background ...................... 2  
  1.2 Aim and objectives .................................................................................................... 6  
  1.3 Research methodology ............................................................................................. 7  
  1.4 Thesis organization ................................................................................................. 7  
2 Literature review ............................................................................................................. 9  
  2.1 Literature review of PLC, PLM and PDM ................................................................. 9  
   2.1.1 Product and product life cycle ............................................................................. 9  
   2.1.2 Product life cycle management (PLM) and product data management (PDM) 11  
  2.2 Classification methods ............................................................................................. 15  
  2.3 Teamcenter data model ............................................................................................ 29  
  2.4 Conclusion ................................................................................................................ 32  
3 Research methodology .................................................................................................. 33  
  3.1 Scientific area .......................................................................................................... 33  
  3.2 Volvo Cars Engine area (Data collection) ............................................................... 36  
  3.3 Teamcenter area ...................................................................................................... 45  
   3.3.1 Classification application ..................................................................................... 46  
   3.3.2 Classification admin application ......................................................................... 48  
   3.3.3 Resource manager application ......................................................................... 51  
   3.3.4 Data entering in the Teamcenter Software ......................................................... 52  
   3.3.5 Definition of attributes in the Teamcenter Software .......................................... 55  
  3.4 Data classification method ....................................................................................... 57  
   3.4.1 Product ................................................................................................................ 58  
   3.4.2 Process ............................................................................................................... 59  
   3.4.3 Resources .......................................................................................................... 65  
  3.5 Conclusion ................................................................................................................ 68  
4 Implementation and results ............................................................................................ 69  
  4.1 Implementation of method on the bicycle ............................................................... 69  
  4.2 Entering data of two types of documents in classification ....................................... 74
# Table of figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Volvo Cars Engine Factories</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td>Volvo cars Engines (Kraft, 2013)</td>
<td>3</td>
</tr>
<tr>
<td>3</td>
<td>Volvo Cars Engine PDM Project steps</td>
<td>4</td>
</tr>
<tr>
<td>4</td>
<td>Siemens PLM Roadmap</td>
<td>4</td>
</tr>
<tr>
<td>5</td>
<td>Teamcenter steps</td>
<td>5</td>
</tr>
<tr>
<td>6</td>
<td>Teamcenter Functional Structure</td>
<td>5</td>
</tr>
<tr>
<td>7</td>
<td>Research Methodology</td>
<td>7</td>
</tr>
<tr>
<td>8</td>
<td>PLC phases (Crnkovic, et al., 2003)</td>
<td>10</td>
</tr>
<tr>
<td>9</td>
<td>PLC phases from different point of view (Stark, 2006)</td>
<td>10</td>
</tr>
<tr>
<td>10</td>
<td>Overlapping PLC phases (Stark, 2006)</td>
<td>11</td>
</tr>
<tr>
<td>11</td>
<td>The evaluation of PLM (CIMdata, 2002)</td>
<td>12</td>
</tr>
<tr>
<td>12</td>
<td>PDM history (Rodger, 2003)</td>
<td>14</td>
</tr>
<tr>
<td>13</td>
<td>Different types of classified data</td>
<td>16</td>
</tr>
<tr>
<td>14</td>
<td>Equal interval classification method</td>
<td>17</td>
</tr>
<tr>
<td>15</td>
<td>Quantile classification method</td>
<td>18</td>
</tr>
<tr>
<td>16</td>
<td>Natural breaks classification method</td>
<td>18</td>
</tr>
<tr>
<td>17</td>
<td>Standard deviation classification method</td>
<td>18</td>
</tr>
<tr>
<td>18</td>
<td>Different hierarchical procedures</td>
<td>24</td>
</tr>
<tr>
<td>19</td>
<td>A sample group of items</td>
<td>26</td>
</tr>
<tr>
<td>20</td>
<td>Three types of classification</td>
<td>27</td>
</tr>
<tr>
<td>21</td>
<td>Hierarchical classification (1. color, 2. shape, 3. size)</td>
<td>28</td>
</tr>
<tr>
<td>22</td>
<td>Classification according to type</td>
<td>29</td>
</tr>
<tr>
<td>23</td>
<td>Teamcenter data model structure</td>
<td>30</td>
</tr>
<tr>
<td>24</td>
<td>Abbreviated view of the Teamcenter POM and business object hierarchy</td>
<td>31</td>
</tr>
<tr>
<td>25</td>
<td>Product life cycle phases and sub phases</td>
<td>34</td>
</tr>
<tr>
<td>26</td>
<td>Data classification items</td>
<td>35</td>
</tr>
<tr>
<td>27</td>
<td>Data and information sources</td>
<td>36</td>
</tr>
<tr>
<td>28</td>
<td>Data collection methodology</td>
<td>37</td>
</tr>
<tr>
<td>29</td>
<td>Documents and data features</td>
<td>38</td>
</tr>
<tr>
<td>30</td>
<td>Location of advanced engineering data in business idea sub phases of the PLC</td>
<td>39</td>
</tr>
<tr>
<td>31</td>
<td>Manufacturing engineering data in different phases of the PLC</td>
<td>39</td>
</tr>
</tbody>
</table>
Figure 32. Equipment suppliers’ data in different phases of the PLC ........................................ 40
Figure 33. Equipment discipline’s data in development phase of the PLC .............................. 40
Figure 34. Packing discipline’s data in the development phase of the PLC ......................... 41
Figure 35. Quality control discipline’s data in development phase of the PLC ...................... 41
Figure 36. Gauging and tooling discipline’s data in development phase of the PLC .......... 42
Figure 37. Productivity engineering discipline’s data in development phase of the PLC ....... 42
Figure 38. Customers’ data in operation and maintenance phase of the PLC ....................... 43
Figure 39. Volvo Cars Engine Factory Data Map with focusing on manufacturing engineering scope.......................................................................................................................................... 44
Figure 40. Teamcenter Application flow chart for classification............................................ 45
Figure 41. Classification interface ......................................................................................... 47
Figure 42. Classification procedure In TC ............................................................................ 48
Figure 43. Searching procedure in classification application ................................................ 48
Figure 44. Classification Admin Procedure in TC ................................................................. 49
Figure 45. Classification Admin interface ............................................................................. 50
Figure 46. Creating tool assembly in the resource manager application procedure .............. 52
Figure 47. Resource manager interface .............................................................................. 52
Figure 48. Differences between item, form and class ............................................................ 54
Figure 49. Some type of drill bits .......................................................................................... 56
Figure 50. Geometrical values for multi-step drill bit ............................................................. 56
Figure 51. First and second levels of classification hierarchy ............................................. 58
Figure 52. Product Classification ....................................................................................... 59
Figure 53. Process classification .......................................................................................... 60
Figure 54. Machining process classification ........................................................................ 61
Figure 55. Drilling process classification ............................................................................. 63
Figure 56. Conventional drilling attributes ......................................................................... 64
Figure 57. Assembling and disassembling classification ..................................................... 64
Figure 58. Resources classification ..................................................................................... 65
Figure 59. Machining tools classification ........................................................................... 66
Figure 60. Tools components classification ......................................................................... 67
Figure 61. Tool assemblies classification ............................................................................ 68
Figure 62. First and second level of the hierarchy ............................................................... 69
Figure 63. Resource and process classification hierarchy ................................................... 70
Figure 64. Conventional drilling attributes in Teamcenter ................................................ 71
Figure 65. Resource classes attributes ...................................................................................... 72
Figure 66. Bill of material and factory plan............................................................................. 72
Figure 67. Bill of Process........................................................................................................... 73
Figure 68. Bill of process (OP80) ......................................................................................... 74
Figure 69. Process features table ........................................................................................... 75
Figure 70. Drilling process attributes .................................................................................... 75
Figure 71. Search results for drilling processes with 50 mm diameter .................................. 76
Figure 72. A part of PFMEA for OP80.................................................................................. 76
Figure 73. Drilling class attributes ......................................................................................... 77
List of abbreviations

BOP            = Bill of process
BOM           = Bill of material
CNC            = Computer numerical control
DFMEA      = Design Failure modes and effect analysis
GCS             = Guided component search
ICO             = Instant classification object
IDE             = Integrated development environment
Key-LOV   = Key-List of values
NC              = Numerical control
PDM           = Product data management
PLC            = Product life cycle
PLM           = Product life cycle management
TC              = Teamcenter
VCE           = Volvo Cars Engine
1 Introduction

Different variety and high number of products has caused more production management. Product Data classification is a part of data management that will be in this thesis. This thesis is including two main parts. The first part is about product data management (PDM) and data classification and the second part is about implementation of data classification in a special PDM software (Teamcenter) and in a special production factory (Volvo Cars Engines factory). Teamcenter software ability and factory information accessibility make some restrictions for classification that should be considered in the both parts of this thesis. The first part of this chapter is backgrounds explanation. After explaining of background, aim and objectives of this thesis will be described and the scope of the work will be clarified. The last section of the introduction chapter is explaining some information about the thesis organization.

1.1 Background

In this chapter, the reports’ background will be divided into two separate parts and these parts will be explained. One is about some main concepts like product, product life cycle and product data management for giving a preparation, for understanding of next steps like data classification to the readers. The second is a background about the Volvo Cars Engine factory and Siemens Teamcenter software.

1.1.1 Product data management (PDM)

Anything that is produced and offered to the markets can be a product. Computers, software, table and foods are different kinds of products. Each product is created in a specific time and has a life cycle. Duration of a product life cycle can be only a few seconds or many years, but this life ends at a specified time. The life of the product from first stage to the end is defined as Product Life Cycle (PLC).

There are many benefits in the management of the product life cycles. For example, with the managing of PLC, products can be offered to the markets at the right time. On the other hand, with PLC management (PLM), companies can reduce product related costs. Finding customer requirements, faster product development, scheduling of maintenance programs and producing with lower costs are some of the main advantages of PLM.
The main way of PLM is managing of product data. There are a lot of data produced or used in the product life cycle and with managing of these data and information, product data management (PDM) can be implemented.

History of product data management shows that PDM started from the time that products’ drawings were on paper. They had been classified and managed in metal or wooden files. When computer and software used for generating of data and information, some software introduced for product data management. The initial version of these programs was just used for the management of drawing’s files. But nowadays, these software programs are trying to cover all parts of PLC information.

This huge amount of data that is generated in different part of product life cycle should be grouped and arranged in a structured way. This categorizing of data is data classification. Without data classifying, finding previous data is not possible or it will be time consuming. Data classification helps the data producers to place their data in a structure for faster search and reuse of them. The data classification structure is based on data attributes. This thesis tries to find the suitable way of product life cycle data classifying, according to the company and software limitations.

1.1.2 Volvo Cars Engine factory and Teamcenter software background

Volvo Cars Engine factory production started in 1927. On the 14th of April 1927, a large door opens at a factory on the island of Hisingen, Gothenburg. Out rolls the first-ever production Volvo. An open car with a four-cylinder engine, its model name is ÖV4. The new car is priced at 4,800 kronor, and the later PV4 saloon version will cost 1,000 kronor more. The aim is to build 500 of each model, but only 297 are sold in the first year. The brand’s founders, managing director Assar Gabrielsson and technical director Gustaf Larson, are to remain at the helm of the company until the mid-1950s (Volvo cars website, 2007). Until now many different car models produced in the Volvo Cars factories and now, different parts are produced and assembled in different cities in Sweden. Volvo Cars Engine factory (VCE) includes the component plant in Floby and the engine plant in Skövde (Figure 1). Totally 2,180 employees working in these factories (1,662 peoples in Skövde and 518 people in Floby) from 21 nationalities, where 19.1% of them are women. Now, three kinds of engines are produced in VCE: 5-cylinder petrol, 5-cylinder diesel and 4-cylinder diesel (Figure 2) (Kraft, 2013).
In the year 2012, 64,022 numbers of 5-cylinder petrol engine, 159,858 numbers of 5-cylinder diesel engine and 188,413 numbers of 4-cylinder diesel engine was produced in VCE. Each engine has five main components: 1: cylinder block, 2: cylinder head, 3: crankshaft, 4: camshaft, 5: connection rod (Kraft, 2013).

There is a project defined for implementation of the Siemens Teamcenter PDM software in the Volvo Cars Engine factory. It starts in 2012 and it will be completed in 2020. There are different steps for completing this project shown in Figure 3. This thesis is the third step of this project, after Bill of process foundation and plan simulation steps.
The Teamcenter is a product life cycle management software. More than 6,400 customers use this software, across about 9,900 operations with 5 million seats (CIMdata, 2010). The Teamcenter software was initially generated for computer-aided design (CAD) data file management in 1980’s. Since that date, Teamcenter was updated on new versions and in each new version; it meets some new customer needs (Figure 4).

**Figure 4. Siemens PLM Roadmap**
The Teamcenter software can give managers a better control of product life cycle with its applications. Teamcenter applications can be divided into three steps (Figure 5). The first step is start and in this step, product designs, documents, BOM and data are managed and shared and with standardizing of workflows and change processes, process will be managed.

The next step is extending of the value by managing of requirements and suppliers and connecting engineering with manufacturing and service.

The last step for product life cycle management is business transforming. The Siemens Company provides some application for quality management, product cost management, sustainability and environmental compliance and system engineering for business transforming.

Figure 5. Teamcenter steps

With this variety of Teamcenter application, managing of product life cycle is possible in a comprehensive way (Figure 6).

Figure 6. Teamcenter Functional Structure
1.2 Aim and objectives

This thesis is a part of the PDM project in the Volvo Cars Engine (VCE) factory and the aim is to find the suitable classification method for product data and information in the PDM system that is used by VCE.

There is a big amount of product data generated in the PLC. Data classification is arranging data in a data structure. The re-usage ability that data-classifying offers to users, reduces costs by reducing work resources as well as the products will enter the market in a shorter time. Classification is a core part of product data management, because it gives a list of data about previous resources for reusing them. Effective classification is therefore crucial for the PDM system.

In reaching this aim, several objectives should be achieved respectively. The four main objectives are:

1. Defining some concepts like PLC, PLM and PDM and specifying, how these concepts can be implemented in the Volvo Cars Engine PDM project. Achievement of this objective makes a general view of the PLM project possible in the Volvo Cars Engine and clarifies the concepts and implementation method of the PLM project in the factory. It is clarifying the way of this project and the project progress up to now.

2. All of the relevant PLM data and information will be identified and some of them will be chosen for classification. The base of data classification is data and all produced and related data to the PDM should be specified. Then some data will be selected for classification from all data, with respect to their usage and their attributes.

3. The next objective is to find a classification method and to create a data hierarchy. To achieve this objective data and their attributes must be clarified and there are some restrictions about the company and software that should be considered like the type of company’s data and software abilities.

4. The last objective of this thesis is to implement this classification method in the PDM software.
1.3 Research methodology

This thesis’ methodology starts from three points. This approach was chosen because of the nature of this research and three main areas that affect this thesis. The first one is the scientific area and it is about product life cycle phases and effective items on classification. The second area that affects the thesis method is Volvo Cars Engine factory’s data collection and the data collection methodology. The last is the PDM software’s (Teamcenter) abilities and its procedures for classification (Figure 7).

When searching and studying of these areas is completed, then, with this knowledge, the product data classification method will be prepared and generated. The last area of this thesis is the implementation of this method in the Teamcenter software.

1.4 Thesis organization

This thesis includes Six different chapters. The first chapter is the introduction and it is explaining the background that readers need to know before starting the core part. It is also showing the aim and objectives of this thesis and the general explanation of the method. The second chapter is dedicated to previous studies and findings about product, PLC, PDM, and classification methods. The third chapter is extending each area of the thesis methodology. The fourth chapter is explaining the implementation of the finalized classification method and presenting an example, from the first level to
the end. The fifth chapter is the conclusion and discussion chapter, and it is divided into two parts. In the first part, the whole thesis will be reviewed, and in the second part, the next steps about classification and PDM project at the VCE, will be explained. In the last chapter, some questions that are raised during the completion of this thesis and the answers given to them, according to this project will be explained. These questions can be as the base for future works and researches about the PDM and data classification. At the end of this thesis, all references that are used in this project are listed.
2 Literature review

In this thesis, the literature review is separated into two parts. One part is about PDM and the other part is about classification. In the PDM part, some concepts like product, product life cycle management and product data will be introduced and in the classification part, literature about classification methods in general and data classification will be reviewed.

2.1 Literature review of PLC, PLM and PDM

For better explanation, this part of the literature review is divided into two sections. The first section is about product and product life cycle. The second section is about product life cycle management and product data management. Product data management is defined as a core of product life cycle management, but sometimes this core occupies nearly the whole product life cycle management.

2.1.1 Product and product life cycle

Kotler explained product from a business point of view as anything that can be offered to a market that may satisfy a want or need (Kotler, et al., 2006). Anything that is manufactured or built can be a product (Crnkovic, et al., 2003). With these definitions about producing a high variety of products are justifiable and they can be in different kinds. Some kinds of products are tangible like cars, bicycle, book and some of them are intangible like software and services. Every product without consideration of its type has a life cycle. Their life cycle started from the beginning as an idea to the end as disposal materials. The product life cycle is divided into different phases. Each phase is characterized by many things as the activity that happened in that phase, the role of product and the responsibility of different disciplines. Crnkovic has divided product life in six different phases. Business idea, requirement management, development, production, operation and maintenance, and the last phase is disposal (Figure 8). He explained that the feasibility study performs in the business idea phase. He mentioned that in the requirement management phase, requirements identification and their analysis and specification would be cleared. In the development phase, the product is designed, and it will be manufactured in the production phase. He explained that for a software product, the development phase required more energy and cost than the production phases and for hardware products, vice versa. This happened, because for...
the software products, all design and programing that required lots of human working, done in development part and only copying of software and distribute of it, is done in production phase. On the other hand, for hardware like a car, manufacturing needs more budget than other phases. Operation and maintenance is the next phase and the only phase that consumers can see. Disposal is the last phase that Crnkovic defined and this phase depended on the kind of product and its effects on the environment (Crnkovic, et al., 2003).

Stark (2006) explained the life cycle from two different viewpoints:

a. Manufacturers of product
b. User of products

He divided PLC into five phases, but with different descriptions, from both product manufacturer and product user point of view. From the manufacturer point of view, he defined PLC to Imagine, Define, Realise, Support (Service) and Retire phases. From the user point of view, he defines PLC to these phases: Imagine, Define, Realise, Use (Operate), Dispose (Recycle) (Figure 9). He also explained that the phases have not seen one after the other by the user, and they overlap each other (Figure 10) (Stark, 2006).

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Business idea</td>
<td>Requirements management</td>
<td>Development</td>
<td>Production</td>
<td>Operation and maintenance</td>
</tr>
</tbody>
</table>

Stark (2006) explained the life cycle from two different viewpoints:

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Manufacturer view</td>
<td>Imagine</td>
<td>Define</td>
<td>Realise</td>
<td>Support (Service)</td>
<td>Retire</td>
</tr>
<tr>
<td>User view</td>
<td>Imagine</td>
<td>Define</td>
<td>Realise</td>
<td>Use (Operate)</td>
<td>Dispose (recycle)</td>
</tr>
</tbody>
</table>
Michael Grieves (2006) divided PLC into five categories in his book about product life cycle management. Plan, design, build, support and dispose are five defined categories that he explained for product life cycle (Grieves, 2006).

In this report and in the chapter 3, section 1, the product life cycle deviations into different phases and sub phases that happened with respect to Volvo Cars Engine factory’s product will be explained.

### 2.1.2 Product life cycle management (PLM) and product data management (PDM)

“Product life cycle management (PLM) is a systematic, controlled method for managing and developing industrially manufactured products and related information” (Saaksvuori & Immonen, 2004). John Stark (2006) defined PLM as an activity of managing products all the way across their life cycle in the most effective way (Stark, 2006). There are many reasons for companies for managing their products’ life cycle. Nowadays, products’ functionalities are increased and faster development of product is required. This is one reason why it is essential to manage the life cycle of the product. Supporting and developing product services is another reason for PLM. Faster production of products with lower costs is one of the most important reasons of PLM. Product life cycle management helps companies to conduct better and clear feasibility studies. PLM offers different benefits to different peoples and managers (Stark, 2006).

CIMdata (2002) defines PLM as: “A strategic business approach that applies a consistent set of business solutions in support of the collaborative creation, management, dissemination, and use of product definition information across the extended enterprise from concept to end of life integrating people, processes,
business systems, and information” (CIMdata, 2002). CIMdata believes that businesses have three main challenges. The first challenge is customer need. Customers’ needs are increasing very fast and businesses must prepare themselves for these increases and predict customer needs, for the future. Achieving operational excellence is the second challenge that businesses should aim at for their operations and their partners’ to improve their product quality. The last challenge is providing product leadership. If a company can pass these challenges successfully, it can offer the right product to the right market at the right time for the right cost. Businesses are able to pass through obstacles, with doing of some fundamental improvements. Innovation in different phases of product life cycle, like in creating new products, processing products and in supporting them is one of the ways to pass the challenges. Businesses also must have the ability of globalization. If a business wants to be a global business, it must make effective use of global organization without any boundaries. It must have efficiency to merge works together and be able to organize some virtual product teams, consisting of people from different places. That business must enable for developing and supporting of products, 24 hours a day and 7 days a week. PLM is the best solution for giving these abilities to the businesses. CIMdata described PLM evaluation over time (Figure 11). In the 1980’s PLM focused on the design engineering activities, but later this view expanded and included workflow and process. Today’s PDM solutions are viewed as enterprise solution and strategy to improve business efficiency (CIMdata, 2002).

Figure 11. The evaluation of PLM (CIMdata, 2002)
CIMdata grouped benefits of PLM in three different groups. The first group is delivering more innovative products, services, and marketing. Second is reducing costs, improving quality, shortening time to market, and the third group is establishing more comprehensive, collaborative, improved relationship with customers, suppliers and partners. CIMdata also gives an example of specific benefits achieved with the PLM system implementation (CIMdata, 2002):

a. An ~40 percent improvement in product change cycle times
b. A 15-30 percent reduction in prototypes
c. A 40% reduction in lead times
d. A 25% productivity increase in design engineering
e. Reduced development time for a household product by 75% - from 18 months to 4 months
f. Reduced time to cost a product (the time that a manufacturer must pay the cost of a product) from 5 days to 5 minutes
g. Reduced an engineering review process by 83% - from 12 days to 2 days

Saaksvuori & Immonen (2004) explained the core of PLC in his book as the creation, preservation and storage of data relating to a company’s products and activities (Saaksvuori & Immonen, 2004). Kennet McIntosh (1995) also defined engineering data management (PLC) as a systematic way to design, manage, direct, and control all the information needed to document the product through its entire lifespan: Development, planning, design, manufacture and use (McIntosh, 1995).

In the product life cycle, a lot of data and information is produced and managing of this data is called product data management (PDM). As defined in different books, like explained before, PLM is management of product life cycle with information and data. Because of this, PLM and PDM are joined together and PDM is the main part of PLM. Crnkovic (2003) defined PDM as the discipline of controlling the evolution of a product and providing other procedures and tools with the accurate information at the right time in the right format during the entire PLC (Crnkovic, et al., 2003). There is a difference between PDM and a PDM system. Crncovic mentioned that PDM is a generic name for a set of rules and methodologies that comprise functions, from different systems, but a PDM system is an implementation for managing product and process data. From the first days that some drawing was generated for products on papers, product data management was
started with classifying these drawings, but manually. As computer technology evolved and creating digital files became more common, PDM software focussed on engineering tasks; like 2D and 3D drawing data. However, in further versions, the PDM software producer tried to cover the entire product life cycle. With the increasing of PDM software abilities, PDM and PLM will be more similar to each other. (Figure 12)

![Figure 12. PDM history (Rodger, 2003)](image)

There are many different features and functions defined for PLC and PDM. From gathering all definitions, PLC and PDM features and functions can be explained in below categories (Saaksvuori & Immonen, 2004) (Crnkovic, et al., 2003) (Rodger, 2003) (CIMdata, 2010):

a) Item management  
b) Product structure management and maintenance  
c) Information retrieval and backup management  
d) Change management  
e) Configuration management  
f) Workflow and process management  
g) Data vault and file/document management  
h) File and data vault  
i) Classification management  
j) Portfolio, program and project management  
k) Formula, package and brand management  
l) Supplier relationship management  
m) Simulation process management
2.2 Classification methods

Classification is defined as “Systematic arrangement in groups or categories according to established criteria” (Merriam-Webster, 2013). Any things that have the same criteria, can be categorized in different groups and classified. The there are many types of classification, for different things. Because this thesis is about data classification, the classification of data is the research area. There are many methods for data classification, and they depend on the point of view that data has to be classified and they depend on the use of classification. Margaret Rouse (2007) defines data classification as the categorization of data for its most effective and efficient use. She explained that computer data can be classified according to any criteria like file type, file size, time of creation (Rouse, 2007). Data classification is not a complex process, but it should be well planned. The first step in data classification is finding the answer to the question: “What do we want to achieve?”. The best way of answering this question is a discussion with people who are involved with data classification (computer weekly, 2009). The classification process, depending on the size and scope of the organization, can be long and complicate. Classification is a human decision-making process and classification software is only a tool for automatic application of classification policies but it does not classify by itself. Bigelow (2005) recommended “While there are certainly software and hardware products available to help discover your data within the enterprise, determine its location, set policies on that data and measure the adherence to those policies, no product has the intelligence needed to determine the "value" of your corporate data” (Bigelow, 2005). Bigelow also mentioned that volume of data and the number of data sheets are not reasons for starting data classification, but business needs and risk tolerance could be some good reasons for enforcing companies to do data classification (Bigelow, 2005). The organizers of Storage Expo show in London on October 2008 said, “Poor data classification can cost companies millions of pounds”. From a survey showed by Storage Expo found that 67% of companies classified their data because of access control, 21% did this for retention control and 12% for retrieval and discovery reason (Savvas, 2008). Alan Pelz-Sharpe, an analyst at CMS Watch (Content Management System Watch) said, companies should place more importance on the impact of retrieval and discovery, with costs in this area reaching one million pounds per terabyte of data. He also mentioned that 1GB of storage costs
about 10 pounds, however the cost of legal discovery on 1GB of storage would be at least 1,000 pounds, so storing everything may seem cheap on the one hand, but can become very expensive, should something go wrong (Savvas, 2008).

In this report, two examples of data classification, that are useful in different working areas, will be explained. Then the weakness of these methods for this specific project will be clarified and the hierarchical classification method will be introduced as a solution.

One common aspect for data classification is data security. From this aspect, for example, data can be classified into eight different classes: Secret, Sensitive, Confidential, Internal, Restricted, Private, Public, Unclassified (Figure 13) (Maintec, 2012).

![Figure 13. Different types of classified data](image)

Some of these classes can be eliminated from this classification and it depends on the company decision about the number of classes. There are some methods that most of times have been used for classification of geographical data.

There are four different data classification method introduced in this area:

- Equal interval
- Quantile
- Natural breaks
- Standard deviation
In figures 14 to 18 the results of four types of classification for a sample data is presented. In this sample, the data about under 5 years old children’s population, in different areas in the city has been gathered. Now, the aim is grouping of these areas (Classification) for future uses.

**Equal interval:** In this type of classification, the range of attribute values is divided into equal-sized sub ranges. This technique emphasizes the amount of an attribute value relative to other values. This method is useful when classification steps are nearly equal-size and attribute values bare distributed in the whole range of values. Otherwise, it is possible that some classes do not have any members. (Geographic Information and Mapping, 2009). For population sample, as shown in Figure 14, the minimum population’s percent is 3 and the maximum population percent is 18. The value range is divided into five equal classes. The right graph shows, that most of areas belong to the second group (6% to 9%). The left mapping graph also specifying classes with different colors (Figure 14).

![Figure 14. Equal interval classification method](image)

**Quantile:** In this classification method, each class holds the same number of members. The quantile method is very useful for linearly distributed data. The size of this distribution is stable by setting the number of classes. The main weakness of this method is the gaps that may occur between the observations (Geographic Information and Mapping, 2009) (Stern, et al., 2011). In the Figure 15, the number of areas that belong to each class are equal.
Natural breaks: This method is based on natural groupings inherent in the data. The data set has been grouped by considering visually logical and subjective aspects. This classification method is useful for mapping values that are not equally distributed on the histogram (Geographic Information and Mapping, 2009) (Stern, et al., 2011). For example, in the Figure 16, members are divided into classes whose boundaries are set where there are relatively big jumps in the data values.

Standard deviation: In this method the mean values and the standard deviation from the mean have been calculated. Class breaks are then created using these values. This method (also known as mean-standard deviation) is useful when data have a normal distribution and it is important to find which members are above or below an average value (Figure 17) (Geographic Information and Mapping, 2009) (Stern, et al., 2011).
Because of the wide area of classification, there are many classification methods from different points of view, but they are not directly useful for product data classification. All the methods mentioned above, are useful for data classification, but according to a one attribute.

In the first method, data are classified according to their confidence levels as their classification attribute. In other methods, the collected data considered as the attribute, for classification. In the previous example, the percentage of the population of the people who are less than 5 years old is an attribute. With respect to the nature of product life cycle data, a classification method for objects with multi-attribute is needed.

The aim of this project is finding a suitable product data classification method for Volvo Cars Engines factory, and it is not product data modeling of them, but having some knowledge about different data modeling is essential for this aim. In the following, the different types of information and data modeling will be studied briefly, and data modeling concepts will be used in the further steps.

**Information and data modeling**

Tina Lee described an information model as a representation of concepts, relationships, constraints, rules, and operations to specify data semantics for a chosen domain of discourse (Tina Lee, 1999). She wrote that there are different practices in developing an information model and the underlying methodologies for the recent modeling practices are based on three approaches:

- The Entity-Relationship (ER) approach
- The functional modeling approach
- The Object-Oriented (O-O) approach

**Entity-Relationship (ER):** The ER data model was defined by Peter in 1976 and this model was defined as a way to unify the network and relational database views. Simply stated the ER model is a conceptual data model that views the real world as entities and relationships. A basic component of the model is the Entity-Relationship diagram which is used to visually represent data objects (Chen, 1976). From that time
until now, this method has been developed. The benefits of the ER model are (Windows Enterprise Support Database Services, u.d.):

- It maps well to the relational model. The constructs used in the ER model can easily be transformed into relational tables.
- It is simple and easy to understand with a minimum of training. Therefore, the model can be used by the database designer to communicate the design to the end user.
- In addition, the model can be used as a design plan by the database developer to implement a data model in a specific database management software.

**Functional modeling:** The second approach is functional data modeling. Functional data models are a form of the Semantic data model which appeared early in database history. They use the mathematical formalism of function application to represent and follow associations between data items. Functions are usually applied to variables whose values may be object identifiers or record instances. This data modeling has some advantages (Gray, 2009):

- The syntax is universally understood, and provides a means of expressing schemas, queries and constraints independently of any supplier-dependent language
- Functional expressions follow the rule of Referential transparency – expressions of equal value can be substituted for variables without altering the sense or value of the expression.

**Object-Oriented (O-O):** The last approach is Object-Oriented data modeling. Blaha defined object orientation as “a strategy for organizing systems as collections of interacting objects that combine data and behavior. It applies to many technology areas, including hardware, programing languages, databases, user interfaces, and software engineering”. He also mentioned that “the object Object-Oriented philosophy creates a powerful synergy throughout the development life cycle by combining abstraction, encapsulation, and modularity”. He said that abstraction allows to focus on essential aspects of an application while ignoring details and also preserve design freedom until later stages of development. Encapsulation separates external specification from internal implementation and modularity promotes
coherence, understandability, and symmetry by organizing a system into groups of closely related objects (Blaha & Premerlani, 1998).

There are some essential concepts should be identified for Object-Oriented modeling, but different researchers introduced different concepts. Lee explained that the major concept of this approach is the object, which incorporate both data structure and function. She introduced object classes, attributes, operations and associations (Relations) as building blocks in the Object-Oriented model (Tina Lee, 1999).

Booch explained some principles for the object model in his book. The abstraction, encapsulation, modularity, hierarchy, typing, concurrency, and persistence are principles that he clarified and he mentioned that these principles are not new by themselves, but in the object model these elements are brought together in a synergistic way (Booch, 1994).

Conway identified object, class, method, inheritance and polymorphism as the things that should be clarified to understand the theory of object orientation. He said that by learning of these five things, 90 percent of object orientation theory be understood. He also explained four extra concepts that are particularly relevant to Object-Oriented in Perl method: interface vs. implementation, aggregation, genericity and persistence (Conway, 2000).

Blaha divided object orientation modeling concepts into three main areas. The first one is “Object and class concepts” that include objects, classes, values and object attributes, and operations and methods. The second types of concepts are “Link and association concepts” that include link and associations, multiplicity, roles, link attributes, association classes, and qualified association. The last one is “Generalization concepts” that includes generalization and inheritance (Blaha & Premerlani, 1998).

Ashrafi presents the basic concepts that apply to objects in her book about Object-Oriented systems analysis and design. The concepts that she presents are (Ashrafi & Ashrafi, 2009):

- Real Objects
- Identity
- Attributes
In this project, there are some virtual objects in the Teamcenter software that are representations of real objects in the factory, like parts, processes, tools and etc. These objects should be classified according to their attributes and it is similar to Object-Oriented modeling. Therefore, from concepts that was mentioned previously, the most important of them which are related to this project will be described.

**Objects:** Ashrafi described two types of objects in her book: real objects and virtual objects. Real objects and virtual objects. The real object is something that is perceived as an entity and referred to by name; something that perceptible by one or more of the senses; something intelligible or perceptible by the mind (Ashrafi & Ashrafi, 2009). Virtual means it appears to be real, although it may not have a physical presence and virtual objects are embody the same concepts as real objects, in the information systems and all characteristics of real objects apply to virtual objects. Booch said that an object has state, behavior, and identity (Booch, 1994).

**State:** State is the condition of an object at the certain stage in its lifetime (Ashrafi & Ashrafi, 2009). (Booch, 1994).
**Behavior:** Behavior is how an object acts and reacts, in terms of its state changes and message passing and state represents the cumulative results of the object’s behavior (Booch, 1994).

**Identity:** Identity is that property of an object which distinguishes each object from all others (Khoshafian & Copeland, 1986).

**Class:** A class is a description of a group of objects with similar properties (object attributes), common behavior (operation and state diagrams), similar relationships to other objects and common semantics (Blaha & Premerlani, 1998). Conway defined class as a formal specification of the attributes of a particular kind of object and the methods that may be called to access those attributes (Conway, 2000).

**Attributes and values:** Ashrafi defined attributes as features, properties, qualities, or characteristics that are related to an object (Ashrafi & Ashrafi, 2009). Conway said: “the data to which an object provides access are known as the object’s attribute values. The containers storing those attribute values are called attributes. Attributes are usually nothing more than variables that have somehow been exclusively associated with a given object.” (Conway, 2000).

**Abstraction:** Abstraction is identifying those characteristics of an that distinguish it from all other kinds of entities (Ashrafi & Ashrafi, 2009).

**Encapsulation:** Encapsulation is the packing of data and processes within one single unit (Ashrafi & Ashrafi, 2009). Encapsulation serves to separate the contractual interface of an abstraction and its implementation (Booch, 1994).

**Hierarchy:** Hierarchy is a ranking or ordering of abstractions (Booch, 1994). In the hierarchy structure, there are different levels of classes and each level allows grouping objects, according to different attributes.

There are two common types of hierarchy structuring procedure. (Han & Kamber, 2006).

**Agglomerative (Bottom up):** In this procedure, each object starts with its own class as the first step, and in the next steps, classes that are more similar to each other will be merged, until a class will be generated, that includes all objects. A tree diagram frequently used to show the arrangement of the groups produced by hierarchical
classification is called dendrogram. Figure 18 shows an example dendrogram of agglomerative classification.

**Figure 18. Different hierarchical procedures**

*Divisive (Top down):* In this procedure, all objects are considered as one class and after that, they will be divided to offspring classes in the next levels. An example dendrogram of classification, with the divisive procedure has been shown in the right section of Figure 18.

**Relationships:** Rambaugh defined relationship as a logical binding between objects and he explained three types of relationships: generalization, aggregation, and association.

*Generalization:* A generalization or is a relationship partitions a class into mutually exclusive subclasses. Generalization may have an arbitrary number of levels.

*Aggregation:* Aggregation is an assembly-component or a-part-of relationship. Aggregation combines low level objects into composite objects. Aggregation may be multilevel and recursive.

*Association:* An association relates two or more independent objects. Associations do not exhibit existence dependency (Rumbaugh, et al., 1988).

Booch introduced several common approaches that are evolved in programming languages to capture the Rambaugh three types of relationships.

- Association
• Inheritance
• Aggregation
• Using
• Instantiation
• Metaclass

He also mentioned that, inheritance is perhaps the most semantically interesting of these concrete relationships, and exists to express generalization relationships (Booch, 1994).

**Inheritance**: Inheritance is the mechanism by which a subclass incorporate the behavior of an upper-class (Ashrafi & Ashrafi, 2009).

**Superclass and subclass**: The superclass, or parent class, contains all the attributes and behaviors that are common to classes that inherit from it (Weisfeld, 2009).

**Polymorphism**: Polymorphism is the ability of objects belonging to different classes to perform the same operation differently (Ashrafi & Ashrafi, 2009).

This project is about data classification, and that data are some virtual objects. These virtual objects are representative of some real objects in the factory like parts, processes, tools and etc. in the PDM software. Most of the Object-Oriented concepts previously described, are applicable for this project. Some concepts like objects, classes, attributes and values, abstraction, superclass and subclass, Inheritance, polymorphism and hierarchy can be used in the product data classification.

As explained before Han & Kamber defined two types of hierarchical structures: agglomerative and divisive. Because of three main reasons, the divisive procedure is selected for the hierarchical structure. The first reason is the PLM software. In the Teamcenter software, the classification application uses the divisive procedure. The second reason that makes the divisive procedure better is the classification of new objects. Every day, numbers of data are created in PLC, and these data should be classified. If the agglomerative procedure is used, then these data must be compared with all the previous objects, and because of the number of data (objects), it is impossible. The last reason is the classes in the last level of hierarchy. In this project,
classes with more than one member are needed. When the divisive procedure is used, it is possible to stop this grouping in any level of classification that is required.

Here, for a better understanding of the hierarchical classification with the divisive procedure, a sample is provided. There is a group of objects with different shape, size and colors (Figure 19).

![Figure 19. A sample group of items](image)

As shown in Figure 19, objects are in three types of colors (blue, red and green), three sizes (small, medium and large) and three shapes (circle, square and triangle). The aim is classifying these objects in different groups. First step is the definition of attributes. For the classification, three types of attribute can be defined. Color, Shape and size are different attributes that these objects can classify according to. If a classification according to one attribute were enough, then three types of classification are generated, as shown in Figure 20.
But, if the aim is classifying of objects according to all attributes, then classification should happen in different levels. At each level, one attribute is used for classifying. Figure 21 shows one type of hierarchical classification for this sample. In this type of classification, color of the objects is the first attribute, and the objects are classified according to it. At the next level, each group’s members are classified according to their shapes, and in the last level they are classified according to their size. It is possible to classify them in another way by changing the attribute level.
In this simple example, all objects have all three types of attributes with different values, but sometimes, attributes are common only in some of the objects. Each branch of the hierarchy tree can have its own attributes for the next levels of classification.

In the above sample, objects are grouped according to their value of attributes. For example, in the first level, they are grouped according to their values for color. The blue objects are gathered in one group, green objects are gathered in another group, and red objects are grouped with each other.

But, in this project, objects will be grouped according to the type of attributes not to the values of attributes. Objects with the same type of attributes are gathered in one group or class. Figure 22 shows an example of classification according to the type of objects. In this example, attributes are diameter and volume. All objects with similar attribute types are grouped with each other.
Booch specified different types of relationships between objects (Booch, 1994) and in the Teamcenter software, objects can have different relationships with each other in different applications, but in the classification application, they don’t have any direct relationships. Each class has an inheritance relationship with its superclasses and subclasses, according to its attributes. The classified objects have common attributes according to their class’s attributes.

### 2.3 Teamcenter data model

Having some knowledge about the method of data modeling that has been used by Teamcenter, can help to find the suitable data classification method. Because of this, in this section Teamcenter data modeling will be explained briefly.

Figure 23, shows the Teamcenter data model structure (Siemens(1), 2012) Teamcenter has its own data model and it is structured into two layers:

- Class and attributes
- Business object and properties

![Figure 22. Classification according to type](image-url)
A class can be thought of as a storage class in the database. It is the persistent representation of an object table in the database, and each row in that table is an instance of the class. The columns of the table are the attributes of the class, and attributes are the characteristics of the class. The attributes are also persistent, since they are stored in the database (Siemens(1), 2012).

Business objects, on the other hand, represent the business data you handle in Teamcenter. They let you define the values to store in the database, and set how objects behave, such as their lists of values, naming rules, display names, and so on. Properties are the values on the business objects. Many business objects have a storage class just for them that has the same name, and their properties are stored as attributes on that storage class. These are known as primary business objects. Primary business objects are persistent in the database and have persistent properties. They are persistent because they are stored in the database in their associated storage class. Secondary business objects are business objects that store their information on their parent’s storage class, and do not have a storage class of their own (Siemens (1), 2012). For example, items, folders and tools are primary business objects, and texts and PDF are secondary business objects.

In Teamcenter, properties are the values attached to business objects, (as attributes, which are attached to classes).

There are four forms of properties: persistent, compound, runtime and relation (Siemens(1), 2012).

1. **Persistent:** Persistent properties are stored in their business object’s storage class as attributes. In other words, attributes of a storage class are expressed as persistent properties on the business object that uses that storage class.
2. **Compound**: Compound property values are retrieved from other properties. When you create your own compound property, you specify the property on another business object, and the value of that property is retrieved and used for your compound property.

3. **Runtime properties**: Runtime properties are properties that are calculated at runtime. Basically, there’s some ITK code somewhere that is invoked every time you ask for the value of that property.

4. **Relation**: Relation properties define relationships between objects. Any relation type can be added as a relation property of a business object. It is possible to create own relation properties, just as possible to create own persistent, runtime, and compound properties.

**POM schema**

At the very top of the Business Objects view the parent of all business objects is displayed as POM_object. Persistent Object Manager (POM) defines an architecture (schema) for managing Teamcenter data in a database and in a running Teamcenter session.

*Figure 24. abbreviated view of the Teamcenter POM and business object hierarchy*
POM is a layer between Teamcenter and the database. As such, POM performs many tasks, such as managing concurrent access to persistent data to avoid multiple users overwriting each other's changes. POM manages data in tables within a relational database, such as Oracle. The basic concepts to keep in mind about POM are that (Siemens(1), 2012):

- Each POM class is represented in the database by a table.
- Each instance of a POM class is represented by a row in the class table.
- Each class attribute is represented by a column in the class table.
- An object that is derived from multiple classes requires a join across each of the class tables from which it is derived.

Classes are the persistent representations of the POM schema and Business objects (types) are abstract implementations of POM classes. Each POM class is mapped to a primary business object whose name is the same as the POM class name. Primary business objects are derived from classes such as Dataset, Folder, Form, Item, ItemRevision, and so on. Sub-business objects are under these primary business objects (Siemens(1), 2012). Figure 24 shows an abbreviated view of the Teamcenter POM schema and Teamcenter business object hierarchy.

2.4 Conclusion

In this chapter previous researches about PLC, PDM, data classification methods and data modeling was studied and enough knowledge has been gained, before starting research methodology. In the next chapter, the data classification method for product data management in the Volvo Cars Engine factory with respect to the PDM software will be explained in detail by following the research methodology steps. It is important for the readers to know, that in this report, after this section, the meaning of the “attribute” are classes’ attributes specifically, not the attributes that classes are divided according to.
3 Research methodology

According to the research methodology that was explained in the introduction chapter (Figure 7), there have been four areas in this thesis. In this section, each area will be explained in detail, and each step guides the readers to reach the objectives.

The scientific area section is explaining how the product life cycle is divided into phases and sub phases. In VCE section, different documents and data that are gathered from Volvo Cars Engine factory will be described and they will be placed in product life cycle phases. Data features are also explained in this section. In Teamcenter software area, the software procedure for classification will be explained. In this section, there is a description about the best way for defining attributes and a part is allocated to describe different ways of adding data, to the Teamcenter software. In the last section, the procedure for classification method will be explained. In the last section, the methodology for data classification will be described.

3.1 Scientific area

As mentioned before, in the literature review chapter, different researches divided the product life cycle of phases in different ways. In this project with respect to all kinds of division, PLC is divided into six main phases where each contains several sub phases. This division happened, according to the manner of activities that happens in each phase (Figure 25).

First phase is the business idea, and in this phase, the concept of a product will be completed. The market of this product must be estimated, and the required technology for production, must be determined. After specifying a key requirement for this product, the feasibility study can be completed.

In the feasibility study, payback period and other important information that decision makers need to decide about a product, are presented, and if decision makers decide to start producing it, then the product will be entered into the next phase. There, all detailed requirements for production will be specified.

The development phase is the main phase of product life cycle from the document management point of view. All product designs, process drawing sheets, simulation data and lots of other information are managed in this phase.
The next phase is product phase and it is included sub phases that are related to production. When a product is produced, it will be offered to the markets. Then the operation and maintenance phase is the phase that relates to this section of the product life cycle.

Figure 25. Product life cycle phases and sub phases

<table>
<thead>
<tr>
<th>Phases</th>
<th>Business idea</th>
<th>Requirement management</th>
<th>Development</th>
<th>Production</th>
<th>Operation and maintenance</th>
<th>Disposal</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Perception</td>
<td>Requirement specifications</td>
<td>Conceptual development</td>
<td>Tooling</td>
<td>Operation</td>
<td>Delivery</td>
</tr>
<tr>
<td></td>
<td>Assessment of market and technology</td>
<td>Requirement Analysis</td>
<td>System-level design</td>
<td>Machining</td>
<td>Maintenance</td>
<td>Disassembly</td>
</tr>
<tr>
<td></td>
<td>Key requirement</td>
<td></td>
<td>Detailed design</td>
<td>Assembly</td>
<td></td>
<td>Recovery of recyclable materials</td>
</tr>
<tr>
<td></td>
<td>Feasibility Study</td>
<td></td>
<td>Detailed requirement</td>
<td>Robotics</td>
<td></td>
<td>Disposal of no recyclable material</td>
</tr>
<tr>
<td></td>
<td>Simulation</td>
<td></td>
<td>Testing and refinement</td>
<td>Plant</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Process planning</td>
<td></td>
<td>Quality control</td>
<td>Production</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Production ramp up</td>
<td></td>
<td></td>
<td>Supply</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Quality</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Disposal phase is about product disassembly, recycling and all things that will be have happened to a product when it “dies”. In the next chapter will be described, how the Volvo Cars Engine factories' collected data, are devoted to each of these phases.

**Items affecting data classification:** With this information about product life cycle phases and classification methods that are described in the literature review chapter, a model can be created for finding classification method. In this model, items affecting on classification methods are specified (Figure 26).

Data classification is the categorization of data according their attributes for its most effective and efficient use. Then data, data attributes and method of categorization are three main effective items in classification methods (Figure 26).
In this specific thesis project, two other effective items must be considered. One of these items is the type of data and information that shall be classified, which depends on the Volvo Cars Engine factory's data and information. The other item is the PDM software and it is about the procedure of data classification in that specific software (Teamcenter). In the next sections, more information about these two items will be described.
3.2 Volvo Cars Engine area (Data collection)

The Volvo Cars Engine factory started its work from more than half a century ago and in these years, a lot of data and information was generated and used. At present, there is a lot of data and information generated and transferred between different disciplines and used by different people.

Each day that passes, some data are added to the previous data will be updated. There are different sources for finding and collecting data and information. Documents, software and computer files are different data sources (Figure 27).

![Figure 27. Data and information sources](image)

Most of data and information can be found in documents and forms. Technical specification, control plan, standards and packing instructions are some kind of documents that include lots of data and information. Some of the data has been generated in engineering software, as product designs and computer aided manufacturing files. Accounting data and store data are not engineering data, but they are generated in specialized software. There are also some regular files that most of the time, different personnel generate them in their computers. There are some differences between regular files and special software files. Most of regular files are informal and used by the owner or attached to the formal documents. They are stored in the personal computer and they are not accessible from shared software.

**Data collection methodology:** The aim of this project is to classify the Volvo Cars Engine data. Because the responsibility of the VCE factory in Skövde city is a production of Volvo car engines, most of the data in this factory is related to the manufacturing and process.
A procedure has been defined for collecting, checking and arranging of VCE data (Figure 28). During that procedure, some meetings with different factories’ disciplines happened. At those meetings, different types of data that have been produced or used in that discipline were collected and the methodology of classifying them was defined. After that, a workshop with different parties of this project was held, every week. Each type of data was discussed and was placed in its phase of the PLC. Volvo Cars Engine factory, Siemens Company and University of Skövde are three parties that attended the workshops.

After plenty of meetings with different factory disciplines, the most important data has been gathered. To be able to analyzing of these collected data and documents, their features have been defined in the meetings also. With respect to the type of data and documents, fifteen types of features were specified (Figure 29). With finding data’s features, data and documents will be more clear for the next steps. Because of the variety of data and documents, some features are not meaningful to them.

Title, is the first feature of the data. In the description feature, a short explanation about documents matter will be described. Feature “Type” specifies the type of data. It shows that it is documented, form, file, information or any other type of data. If the data is a file type, then feature “Format” explains the format of the file like Microsoft Excel or Microsoft Word, etc. and the feature “Language” shows the language of the document.
At the features “Producer”, “Sender”, “Receiver” and “User” features, the disciplines, which relate to the document, are specified. If there are any rights about data usage, they will be explained, in the feature “Rights”.

The “size” feature for softcopy data is the size of the file and for hard copy data is the number of pages. If the data have different versions, it will be indicated in the “version” feature and in “periodic/sometime” feature, it will be clarified that data is generated one time or in some periods. In the last feature, the “archive place” of this document is specified.

As explained in data collection methodology, data are collected from the different disciplines of the factory. At each meeting that was held with a discipline, some special data related to that discipline, was collected. In the following, collected data will be presented according to their relation to the Volvo Cars Engine factory disciplines, and the phase of PLC that they are used or produced in (Figure 25). Some of these disciplines have been in the Volvo Cars Engine factory as separate disciplines, and some of them are one section of another discipline and they are not known as separate disciplines.

**Advanced engineering:** This discipline works with three types of data:” Wanted industrial position”, “Technical strategic” and “Product content file”. All of these data are placed in the first phase of PLC and they relate to the strategy of the Volvo Cars Engine factory (Figure 30).
These data are very useful when creating a new product starts and these data are placed in the sub phase of "assessment of market and technology”. Because Volvo Cars Engine factory in Skövde is a factory for producing different types of engines, most of the data and documents are not related to the first phase of the PLC.

**Manufacturing engineering discipline:** Most of disciplines that are presented in this report, is related to the manufacturing engineering discipline, but to clarify of data features, only those data that are related directly to this discipline are placed in this group. Data that are generated or used in this discipline are placed in two phases of the PLC. Some of them are placed in the requirement management phase and some of them are placed in the development phase.

The process requirement description and restriction model are two types of data, the designers need to know, before starting to design a new product (Figure 31).

Other manufacturing engineering data are grouped in the development phase of PLC (Figure 31). A flow chart is a document that gets information about production line. Fixturing is a drawing that describes how the product is fixed in the process line. The Volvo Cars Engine factory has defined certain standards and these standards are given in “standards” data.
Process sheets, activity instruction, capability study and control instruction are other types of documents, and data that is generated or used by manufacturing engineering discipline, in the development phase.

**Equipment suppliers:** Equipment supplier is not a real discipline in the Volvo Cars Engine factory, but it can be seen as a discipline, which generates some product life cycle data.

Most of the times, when a supplier sells equipment to the customer, the supplier gives some documents like machine layout, operation time and hole number specification. These documents are used in process planning (Figure 32).

![Figure 32. Equipment suppliers’ data in different phases of the PLC](image)

**Equipment discipline:** When equipment placed in the factory and before starting its job, for usage of this equipment, many documents should be generated.

Figure 33 shows, different data and documents that are related to equipment’s usage, operation and their tests.

![Figure 33. Equipment discipline’s data in development phase of the PLC](image)

**Packing discipline:** In the packing discipline, all data and documents that relate to the packing process are generated. "Packing drawing" is a drawing about packing of a product.
Packing instruction is another type of document that gives some data about the way of packing a product, and data about the final look of the product, after packing (Figure 34).

**Figure 34. Packing discipline’s data in the development phase of the PLC**

<table>
<thead>
<tr>
<th>Development</th>
<th>Processing</th>
</tr>
</thead>
<tbody>
<tr>
<td>Detailed requirement</td>
<td>Process planning</td>
</tr>
<tr>
<td>Packing drawing</td>
<td>Packing instruction</td>
</tr>
</tbody>
</table>

**Quality discipline:** Quality control discipline is one of the disciplines that generates and works with data in different phases of the PLC. In the development phase, many analyses happen, and many instructions are generated in the production phase.

Figure 35 shows the different types of data and documents that are related to the quality control. Some documents like process failure mode and effects analysis are placed in two phases of the PLC.

**Figure 35. Quality control discipline’s data in development phase of the PLC**

<table>
<thead>
<tr>
<th>Development</th>
<th>Production</th>
</tr>
</thead>
<tbody>
<tr>
<td>Conceptual development</td>
<td>Quality control instruction</td>
</tr>
<tr>
<td>Detailed design</td>
<td>Quality</td>
</tr>
<tr>
<td>Testing and refinement</td>
<td>Process control</td>
</tr>
<tr>
<td>Process planning</td>
<td>PV Test</td>
</tr>
<tr>
<td>Design failure mode and effects analysis</td>
<td>First piece control</td>
</tr>
<tr>
<td>(DFMEA)</td>
<td>Quality control instruction</td>
</tr>
<tr>
<td>Quality control plan by product and process control</td>
<td>Quality control instruction</td>
</tr>
<tr>
<td>Measurement System Analysis</td>
<td>Process failure mode and effects analysis (PFMEA)</td>
</tr>
<tr>
<td>Process failure mode and effects analysis (PFMEA)</td>
<td>KISA</td>
</tr>
</tbody>
</table>

FMEA is a document that explains what failures can happen in the process, and clarifies the effect of these failures on the product. In the conceptual development phase of PLC, PFMEA document was generated, but this document can be used and updated in production’s quality sub-phase.

**Gauging and tooling discipline:** In this discipline all data about tools and gauging instruments, are generated and used (Figure 36).
Gauging tools are instruments that are used for scaling and measuring of product in different sections of the process. Cutting tools include all types of tools that be used by production machineries and equipment. Tool and gauging drawings are about tool’s dimensional and geometrical data.

**Productivity engineering discipline:** Two main types of data are generated in this discipline. One type of them is the data related to simulation of production line and factory. In the Volvo Cars Engine factory, the “Plant Simulation” software has been used for these types of data.

The second type of data is the data that are used in process operations. The operation’s instructions and machineries programs are in this group of documents. Figure 37 shows this discipline’s data, in the PLC phases.

**Customers:** When a product has been sold, then customers can give lots of feedback to the producer. These feedback data and information cannot be placed in the previous discipline’s data. Because of this, an imaginary discipline is defined as customers.

Customer remarks are placed in the phase operation and maintenance of product life cycle (Figure 38).
Most important data and documents, with focusing on the manufacturing engineering scope are explained above. There are a lot of other data and documents, which are generated in the life cycle of a product. However, in this thesis and in this level of the PLM project in the Volvo Cars Engine factory, documents and data have been selected according to their usage and importance. Some of these data has abbreviated or Swedish names that is given in Appendix 4.

When all this data is gathered and their features are specified, with determining responsible for each type of data, a “Data Map” of Volvo Cars Engine factory is generated (Figure 39). This figure is named as "Data Map", because it provides guidance to find required information, from different data and documents. In the next chapter, there is a sample about how classes’ attributes are obtained from these data.
Figure 39. Volvo Cars Engine Factory Data Map with focusing on manufacturing engineering scope

<table>
<thead>
<tr>
<th>Business idea</th>
<th>Requirement management</th>
<th>Development</th>
<th>Production</th>
<th>Operation and maintenance</th>
<th>Disposal</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wanted industrial Position</td>
<td>Restriction model</td>
<td>DFMEA</td>
<td>Cutting tool drawings</td>
<td>Packing drawings</td>
<td>Quality control instruction</td>
</tr>
<tr>
<td>Technical strategic</td>
<td>Process requirement description</td>
<td>PFMEA</td>
<td>Cutting tool list</td>
<td>Packing instruction</td>
<td>PFMEA</td>
</tr>
<tr>
<td>Product content file</td>
<td>Quality control plan</td>
<td>Wear parts</td>
<td>Operation instruction</td>
<td></td>
<td>KISA</td>
</tr>
<tr>
<td></td>
<td>First piece control</td>
<td>Gauging tool list</td>
<td>FU operator instruction</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Measurement system analysis</td>
<td>Gauging tool drawings</td>
<td>Technical specification</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>PV-Test</td>
<td>Operator instruction sheet</td>
<td>Functional test assembling</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Fixturing</td>
<td>work element sheet</td>
<td>Functional test machining</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Flow chart</td>
<td>NC program</td>
<td>Operational reliability electrical</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Control instruction</td>
<td>Measurement program</td>
<td>Operational reliability mechanical</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Standard</td>
<td>Flow simulation</td>
<td>Functional description assembly equipment</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Process sheets</td>
<td>Operation time</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Activity instruction</td>
<td>Hole number specification</td>
<td>Functional description machinery equipment</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Capability study</td>
<td>Machine layout</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Discipline**

- Advanced Engineering
- Manufacture Engineering
- Equipment
- Equipment suppliers
- Packing
- Quality control
- Gauging and tooling
- Productivity engineering
- Customers
3.3 Teamcenter area

As mentioned before, Teamcenter is a product life cycle data managing software that the Siemens Company produced. This project is about product data classification and this classification should be implemented in the Teamcenter software. Because of this, having some information about this software is necessary. The procedure of creating a classification hierarchy, creating attributes, linking the attributes to classes, classifying data, adding data to the attributes and other Teamcenter characteristics are creating some restrictions for finding a classification method.

Two applications of Teamcenter have a role of data classification and one other application is highly relevant to the classification (Figure 40). The classification application and classification admin are two applications that are used in data classification. The classification admin application is an application for generating groups, classes, attributes and finalizing the classification hierarchy to use in other applications like in the classification application. However, in classification application the users can add data to classes with their attributes. The third application is the resource manager that is using classifying tools for making assembly tools. Assembly tools are a kind of tools that have different components, like the drilling system that includes a drill bit and drill holder.

In this section at first, classification application will be explained, to show the procedure of classifying an item by the Teamcenter user. Then the classification admin application will be explained, to show the procedure of creating classes and other essentials for classification. In addition, at the end of this section, a brief explanation about one part of the resource management application that relates to the classification will be given.

Before different applications’ explanation starts, some terms should be defined.
**Classification Object (ICO):** Representation of a Teamcenter object within the classification system that extends the object with classification data. ICOs specify the values of attributes defined by a particular storage class within the classification hierarchy.

**Class:** Classes or storage classes are used to store ICOs, that can be related to workspace objects.

**Abstract Class:** Abstract class used to combine common attributes for use in classes. Classification instances cannot be stored in abstract classes.

**Group:** Groups are a collection of related classes. Groups cannot get attributes.

**Attributes:** Attributes are inherent characteristics that describe and identify an object within a group of objects. For example, you can use the nut height and thread diameter attributes to distinguish particular nuts within a group of nuts.

**Hierarchy:** Displays the Classification Hierarchy pane for accessing the group, class, and view definition panes.

**View:** View tailored representation of attributes within a class. The views are associated with abstract and storage classes. Attribute properties can also be applied.

**Key-LOV:** List of legal values that can be associated with an attribute definition.

**GCS Types:** Displays the GCS Types pane for creating and maintaining data required by the guided component search.

### 3.3.1 Classification application

The classification application is used for:

- Add classification objects (ICO) to the classification hierarchy.
- Classify workspace objects.
- Find classification objects.
- Modify the attribute values of classifying objects.
- Delete classification objects from the classification hierarchy.

The classification administrator must have already created a classification hierarchy and specified attributes in the classification admin application; otherwise, there is not any class in classification application, which users put their objects in. It is assumed that classes and their
attributes are pre-defined, and now, the user wants to classify an object in the classification application.

Figure 41. Classification interface

The user should send the item to the classification application and assign ICO to the item. Also, it is possible to add an ICO to a class directly, without any relation to previous generated item. There are different classes in the hierarchy, and the user must select a class that fits, to the selected item. Each class has its own attributes (Figure 41- Section 1). Some attributes are inherited from the upper level of the hierarchy, and some of them are class’s particular attributes (Figure 41- Section 2). Sometimes, the classification administrator puts some information about class and attributes, in the class viewer part. (Figure 41- Section 3). Attribute values should be specified for the item, and they must be saved. After saving, if there is any information or drawings embedded in that object, they will appear in the instance viewer section (Figure 41-Section 4). With this procedure, a user can classify an item in the classification hierarchy (Figure 42).
The aim of a classification is easy and fast searching, and for the searching in classification application, there is a procedure (Figure 43). Users must go to the classification application, in the Teamcenter software and then they must select the class or abstract class that they want to search in (Figure 41- Section 1). Then they need to enter the desired attributes (Figure 41- Section 2) and they can search it in that class (Figure 41- Section 5). If there are some items with desired attributes in the class, users can find them in the table.

In the classification application in the Teamcenter software, users can modify or delete an ICO object also.

### 3.3.2 Classification admin application

As mentioned in the previous section, before users can classify their items, the classification administrator should establish a classification hierarchy and create groups, classes, and views. In addition, attributes must be defined and be associated to the classes.

There is a procedure for preparing the classification application, and the first part of this procedure is creating the classification hierarchy (Figure 44).
The admin user can create the classification hierarchy by defining groups, abstract classes and classes (Figure 45-Sections 1&2). Groups are containing abstract classes and classes. It is not possible to assign attributes to a group. Abstract classes contain different classes. They can get some attributes, but it is not possible to add classification objects in abstract classes. Classes (Storage classes) are the last level of the classification hierarchy. They inherit their abstract class attributes, and they have their own attributes, also. Users can classify their objects in classes in classification application.
Attributes should be defined in the dictionary tab, before assigning to abstract classes and classes. Dictionary tab is shown in Figure 45- section 4 and the admin user can create, modify and delete attributes in this tab.

Admin users can create different attributes with different format types. Available format type of attributes are:

- Key-LOV
- String
- Integer
- Real
- Date

There are many settings available in dictionary tab for attributes. For example, the admin user can define a special attribute, which gets its value from an object’s features; like ID number.

Key-LOV is a special attribute format type that should be defined before an attribute is being created. KEY-LOV has been selected as the attribute’s format, when the admin user wants to allocate an attribute to the class with predefined values. When the admin user wants to classify an object, for giving a value to this attribute, they can select one of several preset values. For example, if there are two types of screwdrivers; hand and machine, then admin user, makes a
Key-LOV for it. When users want to classify screwdriver in its class, for the type of a screwdriver, they can select hand type or machine type.

Assigning attributes to abstract classes and classes are the next step of the procedure. Attributes can be added to classes with different settings, and these settings can be changed in the class attribute tab (Figure 45 – Section 2). It is also possible to set an access level of different users for classes and abstract classes.

By adding views to classes and abstract classes in the hierarchy, the admin user can set attribute arrangements and general view of attributes in the classification application. In other words, view item is determining the output shape of attributes.

Some components have the ability to attach to other components. Guided component search (GCS) defines a connection point for these kinds of components. For example, if there is a drill holder, it should be attached to a drill bit, to become a usable item. The drill bit and the drill holder should have some connected attributes. The diameter of drill bits should be less than the size of the drill holder. Admin user should create an attribute for this connection in the dictionary and in GCS tab, then he should set a rule for this attribute, between socket and plug. For this example, the rule is:

Drill bit diameter (Plug) < Drill holder Size (Socket)

When these types of rule are assigned to classes and class objects (ICO), users have the ability to find matching components in the resource manager application. For example, they can easily find all drill bits that can be used with a special kind of drill holder.

The next section is about resource manager application and how guided component search is used in this application.

### 3.3.3 Resource manager application

The “Resource Manager” application is used by NC programmers, process planners, and tool designers on a daily basis to store, modify, and retrieve information, about the resources they use in their respective processes.

One of the resource manager usage is making a resource hierarchy (Figure 46). Sometimes users want to assemble two or more tools together before using them in a process. For this objective, users should create a new component, or they can open an existing component, that
represents the roots of a new structure. Then they must add other components, to build up the structure hierarchy (Figure 47- Section 1).

**Figure 46. Creating tool assembly in the resource manager application procedure**

They can select this component from classified objects. If the selected component has guided search component settings, then users can find all possible components for attaching to the root. The root component should be classified, and its attributes values should be specified, before it can be saved and used in a process (Figure 47- Section 2).

**Figure 47. Resource manager interface**

When users assemble some components together, then their 3D models will be attached to each other (Figure 47- Section 3). With the resource manager application, users can create different tool assemblies with a group of components. The classification admin user must create some assembly classes for these assembled items.

**3.3.4 Data entering in the Teamcenter Software**

In the last part of chapter 3.2, many different types of data and documents have been presented, but how these data can be entered into the Teamcenter software. For entering data,
a template should be created and values can be assigned to this template. For example, if the rotation speed of a drilling process is 2000 rpm, then a template (attribute) should be defined with the name of rotation speed. Then this attribute is assigned to the drilling process, with the value of 2000 rpm. After some researches in the Teamcenter software, three main ways were found for data entering to the TC.

The first way is defining attribute, in the item’s master form. When a user wants to create an item like a process or tool, one form is created automatically with this item. The admin user can create different type of item (like drilling process item) and define the desired attributes in the master form (like rotation speed). Admin user must create the new type of it in a special software named “Business Modeler IDE (Integrated Development Environment)”. Then anytime, end-users want to create a drilling process in TC, they can create a new drilling process item, and they can assign a value to the rotation speed attribute, in its master form.

The second way is creating a form, with the desired attribute. Admin user creates different forms with preferred attributes in business modeler IDE softer. For example, he creates a form with drilling process attributes. When the end-user wants to add data to an item, he selects suitable form and gives a value, to the form’s attribute, and then connects that form to an item.

The last way is creating classes with the desired attributes. The admin user creates different classes with preferred attributes. Then, end-user can give a value to the attribute, when classifies an item.

Each type of above data entering method has specific benefits and limitations. Figure 46 shows differences between these three methods. According to these differences, the best method for data entering should be selected.

If there is a data for all of the objects, then it is better that an item is defined with an attribute for this data in its master form. Then any time, a user wants to create this type of item, he can fill the attribute of this item, and this data is placed in the master form. For example, time is a common attribute in all kinds of processes, because of this, an item with the time attribute in its master form, can be defined for processes. When a user wants to create a new process object, then he can fill time of the process simultaneously, in the time attribute. The main advantage of this method is requesting data from the users when they create their new items. However, it should be considered, that there is a lot of different types of data that are not common for all of items and creating lots of items’ type is a big disadvantage. For example,
there are many types of the drilling process, and each type has its own attributes and data. For entering all of these data to the Teamcenter software with the item method, for each type of drilling, one item should be defined, and this is increasing item types. If there are lots of item types, then creating a new item becomes a very complicated procedure.

The second type of data entering is using forms. Using forms has some benefits and some disadvantages. Different forms can be created according to different type of data and can be connected to the related items. Forms can have different revisions and this can be very

![Figure 48. Differences between item, form and class](image-url)

<table>
<thead>
<tr>
<th>Item (Master Forms)</th>
<th>Form</th>
<th>Classification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Create or change item/class/form</td>
<td>Yes (requires Business modeler IDE)</td>
<td>Yes (requires Business modeler IDE)</td>
</tr>
<tr>
<td>Create or extend attributes</td>
<td>Yes (requires Business modeler IDE)</td>
<td>Yes (requires Business modeler IDE)</td>
</tr>
<tr>
<td>Search based on the attributes</td>
<td>Yes Saved queries need to be defined</td>
<td>Yes Saved queries need to be defined</td>
</tr>
<tr>
<td>Changes item’s schema when creating new classes or form</td>
<td>Yes (requires Business modeler IDE) Can be done with system online</td>
<td>No (requires Business modeler IDE) Must be done when no users are logged in</td>
</tr>
<tr>
<td>Changes schema when adding / modifying / deleting attributes</td>
<td>Yes (requires Business modeler IDE)</td>
<td>No (requires Business modeler IDE)</td>
</tr>
<tr>
<td>Display an image with the class/form</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Change classification/form to another class/form</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Support multiple classifications/form</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Have hierarchical structure</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Support revisions</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>
beneficial. It is better that the collected data and documents, which have been as a form in the Volvo Cars Engine factory now, be entered in the Teamcenter as a form. One disadvantage of form is the way of search, based on the attributes. For searching based on the attributes, some templates should be defined in another application named query builder. For creating a new form, special software (Business modeler IDE) needed.

Classification is the last way of entering data into the Teamcenter software. Figure 48 shows differences between classification and other methods, and these differences show classification benefits and disadvantages. Creating templates for data in classes and searching based on the attributes are easy and fast. It is possible to add images to the class for explaining attributes and class features. Because of hierarchical representation of classes, finding a suitable class is easier. However, the impossibility of having revisions of data is the main disadvantage of classification. In the Volvo Cars Engine factory, the data that relate to the item’s features can be added to the classes and users can classify their items by giving values to the attributes.

According to different types of data entering, all collected data and documents should be discussed, one by one and the way of entering them to the Teamcenter software should be selected. At the end of the implementation and result chapter, implementation of two documents from the collected data, in the classification way, will be explained.

### 3.3.5 Definition of attributes in the Teamcenter Software

As mentioned in chapter 3.1, one of effective items on the classification method is the attributes. With a good attribute’s definition, less numbers of classes needed to create in the hierarchy, without losing classes features.

Consider that, there are some objects with different attributes and they should be classified. There is conflict in this situation. If all attributes were considered for classification then many classes should be defined (one class for each object). In other hand, if the number of classes be reduced, then some attributes will be missed. The only way to solve this problem is merging attributes to each other and defining comprehensive attributes. When attributes are comprehensive, fewer classes are required.

For example, there are different types of drill bits (Figure 49). Some of them are conventional, but some of them have different steps, for making a hole with different diameter. Some drill
bits make counter sink and some of them counter bore. If for each type of drill bit, one class be created, then there have been many drill bit’s classes and it makes the classification hierarchy large. If it is possible to define a comprehensive geometry attribute that covers all kinds of drill bits, then all types of drill can be placed in one class.

Figure 49. Some type of drill bits

This comprehensive attribute is explained in Figure 50. For all complex drill bits, each step has three geometrical values: diameter (D), height (L) and angle (α). Some rules should be defined for users:

- If in any step, there is counter bore then the angle (α) is zero.
- For drill bit with a conical head, the first diameter (D1) is zero.
- For drill bit with a round head, the first diameter (D1) is zero and notifying with R

Figure 50. Geometrical values for multi-step drill bit

With this geometry attribute and defending some rules, all types of drill bits can be placed in one class, with different attribute values.
3.4 Data classification method

For finding the classification method, all knowledge that was obtained in the three previous areas will be used. In the scientific area, prerequisite knowledge for the classification was specified. In the VCE area, Volvo Cars Engine factory documents and data have been collected and their features have been determined. In the Teamcenter area, three main applications that are used for data classification have been explained. Now, it is the time for the main objective of this project: “Finding the data classification method”. According to the classification admin procedure that was explained in chapter 3.3.2, the first step of the classification is the definition of the classification hierarchy.

As explained in chapter 2.2, in the hierarchical classification, objects should be divided into groups according to their features. In this project, data are objects, and they should be grouped according to their features. According to the collected data features, all data are placed in the three main groups in the first level, and these groups will be divided into smaller groups in the next levels. In addition to the data features, there is another reason for selecting this method for classification. The main objective of data classification is easier and faster data searching and to achieve this aim, the classification hierarchy should be designed in a way that users can easily and quickly do the classifying and search their data. According to negotiations that happened with different engineers and users, this method of classification is perceptible and usable for them.

The first level of the classification hierarchy is to include all data and can have the factory name like "Volvo Cars Engine factory". In the TC software, the first level of the hierarchy must be defined as a “Group” because it does not need any attributes.

As explained about different life cycle phases and factory work scope, there are three main groups of data in VCE.

1. Data that relates to products
2. Data that relates to process
3. Data that relates to resources

Then next level of the hierarchy is these three groups, and the “Abstract Class” is used for these groups (Figure 51). The reason that Abstract Class is used is that Abstract classes can get attributes and they must have the ability to contain some classes in the next level. Only for
the last level of hierarchy, the “Class” type will be used, because it has possibility to contain Classification Objects (ICO).

Some attributes can be defined for abstract classes in the second level of the hierarchy, but it must be considered that classes of next levels will inherit the defined attributes in this level. Because of this, the attribute definition in this level needs high precision. The defined attributes in this level should be meaningful for all objects, which are in the lower classes. For example, each object in TC has a unique ID number. The ID number can be one attribute for second level classes. It gives a possibility to the users for searching an object, in all subclasses according to their ID number. Each abstract class in the second level can be divided into other classes or abstract classes. In the next step, the method of classification in each abstract class (Product, process and resources) will be explained.

3.4.1 Product

The basis of Teamcenter software is the product and product designs. Because of this, the Teamcenter software has lots of applications and capabilities for this kind of data. Designers can generate a product hierarchy that contains different part of the product and connects them to each other (Bill of material). It is also possible to define different variations of the product. Because of this, in this project, product classification is limited to some classes for standard parts like bolts, nuts, pins, etc. (Figure 52). In other projects, products can be classified same as the product’s bill of material, and it should contain different parts’ variants and part attributes.
Classification admin users can set any class in the third level as an abstract class, and add more classes in the next level. This classification should be continued until the attributes of a class are meaningful for all objects of that class. As mentioned before, if the admin user can find some comprehensive attributes that make it meaningful for lots of objects, while the normal users can find the objects they want easily, then less levels of classes are created and the classification hierarchy will be more simple. Each class has two types of attributes. Some attributes are inherited from abstract classes in the upper level, and some others are specified in that class. Admin classification users can find attributes for adding to the classes, from books, standards and other references, and from the collected data from the company. In chapter 3.3.5, the method of defining attributes was described.

3.4.2 Process

Engine production is the main job of the Volvo Cars Engine factory in Skövde. Because of this, process classification is very important.

Process classification is based on the manufacturing process types. The manufacturing process is converting material into things. Because of different existing kinds of view, there are many different classification methods defined by researchers. Robert H. Todd classified processes in two main groups (H.Todd, et al., 1994). The first process group is the process used for modifying work piece geometry, and he called them “shaping” processes. The other group is “Non-shaping” processes that are used for modifying properties of materials. In the next level, he grouped shaping process to “mass-conserving” processes, “mass-reducing” processes, and mass-increasing or “joining” processes. Non-shaping processes have been grouped into “heat treatment” and “surface finishing” (H.Todd, et al., 1994).

Degarmo classified processes in his book about manufacturing in seven main groups (DeGarmo, et al., 1988):
• Casting, Foundry, or molding processes
• Forming or metalworking processes
• Machining (material removal) processes
• Joining and assembly
• Surface treatments (Finishing)
• Heat Treating
• Other

In this project, because of the nature of the product (car engine), a specific classification is defined for processes (Figure 53).

Figure 53. Process classification

Processes are divided into seven main groups.

• **Machining:** Machining is the removal of undesirable material from a workpiece in the form of chips to obtain a finished product of wanted size and shape.

• **Assembling and disassembling:** Assembling is joining two or more parts together with mechanical, thermal or chemical couplers. Disassembly is separating two or more jointed parts.

• **Casting and molding:** The molten material, poured into a mold or cavity, which contains the material in suitable shape during solidification.

• **Forming:** Forming is fashioning metal parts through mechanical deformation

• **Packing:** Packing is the process of protecting the product and preparing for distribution, storage and transport.

• **Gauging and measurement:** Gauging and measurement are the act of measuring and they are the fundamental activity of an inspection.
- **Other**: Other processes that cannot be in previous groups are placed in this group like painting, hardening, annealing etc.

Type of process is the main feature for the process classification. There is not any casting process or forming process in the Volvo Cars Engine factory in Skövde, but it can be in the classification hierarchy by default, then this classification can be useful for other Volvo Cars factories.

Packing class can be defined as a part of the “assembling and disassembling” class, but in this method because of the importance of it, it is defined as a separate class. The “Other” class contains some processes that are not classified in other groups, like “painting”.

These groups are a general type of processes and each one of them can be divided into some detail processes. They should be defined as “Abstract classes” in Teamcenter because they should have the ability to contain the next level of classes. The “Machining” process is one of the most important processes in VCE. In this report, machining abstract class will be classified into the next levels as a guideline for other classes. As mentioned before, the machining process is the removal of undesirable material from a workpiece in the form of chips to obtain a finished product of wanted size and shape, and in this method, machining is divided into nine different process types (Figure 54). In the following, those process types are described.

**Figure 54. Machining process classification**
• **Drilling**: Creating holes are one of the major processes in manufacturing and a large proportion of them are made by drilling.

• **Turning**: Turning is a machining process of conical and external cylindrical surfaces and it is generally implemented on a lathe.

• **Milling**: Milling is a machining process by which a surface is generated progressively by the removal of chips from the work piece, fed into a rotating cutter.

• **Reaming**: In the reaming process, a small amount of material will be removed from the surface of holes.

• **Threading/Tapping**: Threading is manufacturing of threads on external diameter and tapping is holes threading.

• **Filling/Grinding**: In a filling process, chips are removed by cutting teeth that are arranged in succession along the same plane on the surface of a tool. A grinding process is the same as filling, but in grinding, chips are removed by abrasive particles on grinding wheel or paper.

• **Broaching**: Broaching is a machining process that removes material with toothed tool. Broaching was originally developed for machining of internal keyways.

• **Gear Manufacturing**: Shaping and hobbing is two principal methods for gear manufacturing.

• **Finishing**: Finishing processes are mechanical cleaning and finishing that is done for surface treatments.

These process types can be defined as a “Class” or an “Abstract class” in the Teamcenter. It is depending on the process type, but most of them must be abstract classes because each of them includes different types of manufacturing process. Each abstract class can get some attributes that are common between all the included classes. For example, there are different types of a drilling process but all of them have some common attributes like Rotation speed. Then rotating speed can be a drilling abstract class attribute.
This classification must be continued until some of the classification objects (ICO) have common attributes. When some items have common attributes, then they can be gathered in one group. In this report, the drilling process will be classified as an example.

As defined before, drilling is hole producing and depends on the type of holes. There are different kinds of drilling process (Figure 55). Each type of the drilling process is inheriting attributes of the parent class and has its own specific attributes.

Figure 55. Drilling process classification

Now, conventional drilling process attributes will be explained as an example sample, to clarify how the classification admin user can find attributes.

Drilling is one of the most important manufacturing processes in the Volvo Cars Engine factory. There are many holes in a cylinder block that should be drilled, and there are lots of data about them in documents and data that are collected before. When those data and document are reviewed, lots of the drilling process attributes will be found in the “Process Sheets” (Appendix 1). Another document that is helping to find drilling attributes is the “hole number specification” (Appendix 2). PFMEA is a document that presents the failure mode effects analysis and gets more process attributes (Appendix 3). Figure 56 shows some conventional drilling attributes. Some of these attributes like rotation speed inherit from parent classes, and some of them like hole diameter are particular class attributes. These attributes should be implemented in the Team center software with the procedure that was explained before in chapter 3.3.2.
Assembling and disassembling is another class of machining process. It can be divided into different classes in the same way as drilling process. Figure 57 shows the next level of assembly and disassembly classes. Each of these classes is able to get attributes and can be divided to lower level classes if needed.
When the process classification is completed and the admin classification user defines attributes for classes, the foundation prepares for end-users, to classify their items. Any user can put his items in a specific class and insert attributes values for future searches and uses.

### 3.4.3 Resources

In this method, resource classes include resource data that are related to the production, directly and indirectly. The different process tools, machineries, robots and carriers are some kind of resources. The resources abstract class is divided into ten main groups (Figure 58).

![Figure 58. Resources classification](image)

Many different classification methods have been defined for tools, but in this thesis resources are classified with respect to process classification.

Resource classes are:

- **Machining tools**: Machining tools are tools that used for a different kind of machining like drills, tappers, Shanks, ETC.

- **Fixtures**: Fixtures are devices that support or hold work pieces in the manufacturing process like assembly.
• **Carriers:** The carrier class is including cranes, buffers, rails and all equipment that is used for staff’s transportation.

• **Robots/CNC/Automatic machines:** All automatic equipment and machineries like CNC and robots are placed in this class of resources.

• **Casting Resources:** Casting resources are tools and accessories that are used for casting like molds, furnaces, sand rammers.

• **Forming Resources:** Rolling equipment and drop hammers are two kinds of forming resources.

• **Gauging Resources:** The gauging resources class is including all gauging and measurement instruments that are used for inspection and quality control.

• **Packing Resources:** All equipment and tools that are used for packing have been located in this class.

• **Services Tools:** Some tools have general usage like screwdrivers, hammers, wrenches and pliers. Most of the time, these types of tools are used for services, and they are placed in the services tools group of resources.

• **Other:** Hand tools and power tools that used in assembling and other kind of tools are placed in this group.

The next level of resource classification is different from normal classification methods that were explained before. In chapter 3.3.3 of this report, it has been noted that some tools can be attached to each other, and make assembly resources and they are used in processes. These assembly objects must be classified also. The next level contains two groups of resources. The first abstract class is components, that include different classes of tools before attaching them to each other, and the other abstract class is assemblies (Figure 59). Tool assemblies that have been created in the resource manager application are placed in classes of this abstract class.

![Figure 59. Machining tools classification](image-url)
This deviation between assembly and components happen in this level of classification because in previous levels, abstract classes are very different from each other and there is not any assembling between their components. But, as it will be explained in the next level of these classes, there have been some attachable components like collets and drill bits in the classes. In this report, only machining tools, classified in the next level as an example for other resources’ groups.

After the machining abstract class was divided into two groups of assembly and component, then each group should be classified separately. Figure 60 shows how machinery tool components can be classified.

Some of these tools’ classes were created according to the process that they represent, like milling cutters, broaching tools, and tapping tools. Some of them are used with other components like collets, spacers, inserts, and chunks.
The tool assemblies’ abstract class should be classified also. Because only assembly tools that are used in processes are classified in tool assemblies classes, this abstract class can be divided into less number of groups (Figure 61).

**Figure 61. Tool assemblies classification**

Tool assembly attributes, should be defined in the assembly classes. These attributes are different from the component's attributes.

### 3.5 Conclusion

With the classification methodology that is explained above, the hierarchical classification can be completed and classes attributes can be defined. Because this method follows the divisive (Top down) technique that was explained in chapter 2.2, it is possible to add any classes to any level of the hierarchy, if required.

The last part of the methodology is the implementation. When the classification method is completed, then it should be implemented in the software. When users can classify their objects, then they have their object’s data in the software, and they can search them easily and fast. After that, they can reuse their previous data. It shows that the presented method is applicable and efficient. Because of this, the implementation of this method is considered as the results of this project and it will be explained in the next chapter in detail.
4 Implementation and results

In this project, a new Teamcenter software was installed, and because it was a new software, there was not any data and information inside it. For solving this problem, data and information should be added to the software. For the better understanding of the classification implementation, it was decided that the classification method will be explained on a simple product such as a bicycle, with five components. However, the classification hierarchy and attributes will be defined according to Volvo Cars Engine factory’s documents. It was also decided that the hierarchy must be completed from the root to the end-node in some branches as an example and those attributes should be implemented on these classes.

After implementation of the method on a bicycle production line, the implementation of two types of real documents will be examined. These two types of documents are PFMEA and process drawing, and they are used in the VCE currently.

4.1 Implementation of method on the bicycle

The five bicycle components are the seat, handle, body, rear wheel, and front wheel, and in addition there are two virtual components. The virtual components are two holes for the seat and handle on the bicycle body. The processes of drilling these holes and resources that are needed for these processes are specifying the classification hierarchy branches. These branches should be completed with attributes.

The start of the classification is the creation of a classification hierarchy in the classification admin application. The classification hierarchy will be created, according to the method that was explained in section 3.3.2 of this report,. In this hierarchy, there are three main abstract classes: Product, process and resources (Figure 62).

Figure 62. First and second level of the hierarchy
Then on the next levels, all of these abstract classes will be divided into different classes and abstract classes. As mentioned before, completion of one of the branches of this hierarchy would be enough. Here, because process and resources are the two main abstract classes, then from the process class, the machining process branch and from the resources class, the machining tools branch will be selected for continuing. Figure 63 shows the classification hierarchy of process and resources. It is possible to add icons and images to classes and abstract classes.

As mentioned before, there are two holes as virtual parts, in the bicycle for connecting seat and handle to the body. For creating these holes, there should be two drilling processes. Because of this, from the process part of the classification hierarchy, the attributes of “conventional drilling” classes will be defined. For the resource part of the classification hierarchy, the “drilling assembly”, “drill bit”, and “collets” classes will be completed and then it is possible to classify drilling resources.

**Figure 63. Resource and process classification hierarchy**
For finding conventional drilling attributes, process drawings are very useful (Appendix1). The drilling data contained in that document are nearly the same as attributes that are required for this class. Figure 54 shows the defined attributes for conventional drilling, and Figure 64 shows these attributes in the Teamcenter software. Now, if users create a conventional drilling process in Teamcenter, then they can classify it by adding some values, to the attributes in the conventional drilling class.

Figure 64. Conventional drilling attributes in Teamcenter

<table>
<thead>
<tr>
<th>Bike Type</th>
<th>Process ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>Coordinate Point</td>
<td></td>
</tr>
<tr>
<td>X</td>
<td>mm</td>
</tr>
<tr>
<td>Y</td>
<td>mm</td>
</tr>
<tr>
<td>Z</td>
<td>mm</td>
</tr>
<tr>
<td>Diameter</td>
<td>mm</td>
</tr>
<tr>
<td>Drilling Depth</td>
<td>mm</td>
</tr>
<tr>
<td>Rotation Speed</td>
<td>r/min</td>
</tr>
<tr>
<td>Feed</td>
<td>mm/min</td>
</tr>
<tr>
<td>PFMEA</td>
<td>RPN</td>
</tr>
<tr>
<td>Angle</td>
<td></td>
</tr>
<tr>
<td>Material</td>
<td></td>
</tr>
<tr>
<td>Cutting Fluid</td>
<td></td>
</tr>
</tbody>
</table>

Drill bits and collets are two components, for the drilling process as a resource. Because of this, “drill bit” and “collets” class attributes must be defined. Drill bit should be connected to collet and makes a drilling assembly tool, and this tool will be used in the drilling process. Attributes of the “drilling assembly tool” class should be defined before a drilling assembly tool can be generated in the resource manager application (Figure 65).
For both “collet” and “drill bit” classes, a guided component search was defined. This helps users in the resource manager application to find drill bits that are matched with the selected collet.

As mentioned before, seat, handle, body, rear wheel, and front wheel are bicycle components. The next step of adding data to the Teamcenter is creating the bill of material (BOM). BOM is a list of the raw materials, sub-assemblies, intermediate assemblies, sub-components, parts, and the quantities of each component or material needed to manufacture a product. In addition to these five components, those two holes as virtual components are also added to the bill of material (Figure 66).

When the creation of the BOM is finished in Teamcenter, then the factory’s plan should be defined in Teamcenter. The sample bike factory has three main areas: Machining section,
assembly section, and packing section, and in each area, there are a number of machineries (Figure 66).

The next step is creating resources for the drilling process. Two kinds of drill bits and one kind of collet were created in Teamcenter as resources for drilling processes. Then, with assigning attributes to these resources, they were classified. Afterwards, two kinds of drill assembly were created in the resource manager application and they will be classified.

The last step is creating a bill of process (BOP), and classifying the selected processes from this BOP. Bill of process (BOP) is a list of processes and operations that connect to each other as a workflow. When bill of process was created the bill of material and factory plan items will be linked to the bill of process. Two assembly-drilling resources that have been generated before, in the resource manager application, should be connected to the drilling processes also. At the end, those two drilling processes have been classified in the “conventional drilling” class (Figure 67).

![Figure 67. Bill of Process](image-url)

<table>
<thead>
<tr>
<th>Process Structure</th>
<th>Item Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>000036/A1-Bicycle Process (View)</td>
<td>Drilling a hole in body for seat</td>
</tr>
<tr>
<td>000036/A1-Machining process (View)</td>
<td>Virtual Item - Hole for Seat</td>
</tr>
<tr>
<td>000091/A1-Drilling Hole No1 (View)</td>
<td>Drilling a hole in body for handle</td>
</tr>
<tr>
<td>000098/A1-Hole No1</td>
<td>Virtual Item - Hole for Handle</td>
</tr>
<tr>
<td>000068/A1-First Drilling machine (View)</td>
<td>1.1 Shank and collet</td>
</tr>
<tr>
<td>000150/A1-Assembly Drill 1 (View)</td>
<td>1.1 Shank and collet</td>
</tr>
<tr>
<td>001113/A1-Holder (View)</td>
<td>Packed CPs x 2</td>
</tr>
<tr>
<td>000092/A1-Drilling Hole No2 (View)</td>
<td>1.1 Shank and collet</td>
</tr>
<tr>
<td>000089/A1-Hole No2</td>
<td>1.1 Shank and collet</td>
</tr>
<tr>
<td>000095/A1-Second Drilling machine (View)</td>
<td>1.1 Shank and collet</td>
</tr>
<tr>
<td>000162/A1-Assembly drill 2 (View)</td>
<td>1.1 Shank and collet</td>
</tr>
<tr>
<td>001114/A1-Holder3 (View)</td>
<td>1.1 Shank and collet</td>
</tr>
<tr>
<td>001115/A1-Drill bit 2 (View)</td>
<td>1.1 Shank and collet</td>
</tr>
<tr>
<td>000003/A1-Body (View)</td>
<td>1.1 Shank and collet</td>
</tr>
<tr>
<td>000021/A1-Body (View)</td>
<td>1.1 Shank and collet</td>
</tr>
<tr>
<td>000094/A1-Body Putting (View)</td>
<td>1.1 Shank and collet</td>
</tr>
<tr>
<td>000093/A1-Body Input (View)</td>
<td>1.1 Shank and collet</td>
</tr>
<tr>
<td>000095/A1-Rear wheel assembly (View)</td>
<td>Assemble rear wheel on body</td>
</tr>
<tr>
<td>000026/A1-Rear wheel</td>
<td>1.1 Shank and collet</td>
</tr>
<tr>
<td>000072/A1-Rear wheel assembly (View)</td>
<td>1.1 Shank and collet</td>
</tr>
<tr>
<td>000096/A1-Front wheel assembly (View)</td>
<td>Assemble Front wheel on body</td>
</tr>
<tr>
<td>000127/A1-Front wheel</td>
<td>1.1 Shank and collet</td>
</tr>
<tr>
<td>000070/A1-Front wheel assembly (View)</td>
<td>1.1 Shank and collet</td>
</tr>
<tr>
<td>000097/A1-Seat assembly (View)</td>
<td>Assemble Seat on body</td>
</tr>
<tr>
<td>000024/A1-Seat</td>
<td>1.1 Shank and collet</td>
</tr>
<tr>
<td>000073/A1-Seat assembly (View)</td>
<td>1.1 Shank and collet</td>
</tr>
<tr>
<td>000098/A1-Handle assembly (View)</td>
<td>Assembly handle on body</td>
</tr>
<tr>
<td>000022/A1-Handle</td>
<td>1.1 Shank and collet</td>
</tr>
<tr>
<td>000074/A1-Handle assembly (View)</td>
<td>1.1 Shank and collet</td>
</tr>
<tr>
<td>000096/A1-Packing process (View)</td>
<td>1.1 Shank and collet</td>
</tr>
<tr>
<td>000066/A1-Packing (View)</td>
<td>1.1 Shank and collet</td>
</tr>
<tr>
<td>000020/A2-Bicycle (View)</td>
<td>1.1 Shank and collet</td>
</tr>
</tbody>
</table>

73
The main objective of a classification is a quick search possibility. When the data classification is finished, then any users can find their required data by specifying their search criteria in the classification application, by getting values for the attributes. For example, when other users want to find a drilling process with a rotation speed of 2400 r/m, they can search in the “drilling process” class and find all previous drilling processes with 2400 r/m rotation speed. Then they can use the selected process data, and get information like the tools that have been used for this type of process.

4.2 Entering data of two types of documents in classification

In this section, two types of documents that relate to a specific production line in the Volvo Cars Engine factory will be explained. After that, by entering their data in the classification, the abilities and limitations of classification applications for this data will be cleared. The production line under review is part of cylinder block production that includes some drilling and milling process steps. Figure 68 shows the bill of process for these process steps. It has been defined in the university’s Teamcenter software.

Figure 68. Bill of process (OP80)

The first document for data extraction is a Process sheet (Appendix 1). Figure 69 shows some part of this document and thirteen process steps have been selected. In that table, different features about drilling processes are presented and these features can be defined as drilling class attributes.
The first row in the table gives the attribute names, and the second row gives the units of the attributes. The next rows are the attribute values that should be assigned to the related process, when they are classified. Figure 70 shows some of the drilling class attributes and includes attributes that are obtained from the process sheet.

When the drilling class and its attributes are defined in the classification application, then all thirteen processes steps that are in the Figure 66 can be classified with the information in the Figure 69. When processes classification is finished, then with the search’s abilities of the software, users can search drilling processes with specific values. The results of an example query are shown in Figure 71.
This search result shows that process sheet’s data can be entered into the classification application easily and it is a good way of specifying process features.

The second document that will be analyzed is PFMEA. PFMEA (Process Failure Mode Effect and Analysis) is a document that recognizes and evaluates the potential failure of a process and the effects of that failure. It also identifies actions that would eliminate or reduce the chance of the potential failure occurrence. Figure 72 shows a part of a PFMEA excel file that has been generated in the factory for previous drilling processes.

For entering PFMEA data in Teamcenter, the first row of the table should be defined as attributes for the drilling process class. Because these attributes are fixed, for all types of processes, they can be defined for upper levels of the hierarchy like the process class, and those attributes will be passed to lower level classes like the drilling process class. Each attribute's format should be defined as a string and it should be an array for receiving different values. Each cell of the table, under each attribute in Figure 72 is one value for the attribute. Figure 73 shows the drilling class attributes menu that includes PFMEA table.
When the PFMEA’s data entry for a specific drilling process was started as attribute value, a problem happened in this procedure. There was a limitation for the maximum length of a string attribute, that it should be less than 100 letters. Because there have been explanations about failures and analyzing these failures in the PFMEA documents, then attributes have not enough space for these explanations.

Testing of these two types of documents shows that classification is a good way of entering object feature data in the Teamcenter software, but it is not good for entering analysis and explanations data.

### 4.3 Conclusion

In this chapter, the classification method was implemented in the Teamcenter software, for a simple bicycle as a product, and the bill of material and bill of process was defined for it. The classification hierarchy and classes were generated by the admin user. Attributes of drilling class were extracted from documents and defined for that class. The result of this experiment shows, that it is possible to classify drilling process by assigning its features to the attributes, and after classifying of that process, it is possible to search for that drilling process, among all other process and reuse it. When this experiment tested by VCE engineers, they found it so applicable, and they confirmed that it is easy for them to classify their objects and search for them. This simplicity of classifying and searching shows that it is a suitable classification method for Volvo Cars Engines factory’s product data management system. It causes an easy and fast reuse of previous data and objects, and also makes considerable decrease of time and cost in the engineering works and production.
5 Conclusion and discussion

In this chapter, at first, conclusions about this project will be explained. The manner of completion of the project will be described briefly, and the way of achieving goals will be reviewed. Afterwards, some critical questions that were arose in different steps of this project, will be described. These questions were discussed many times with VCE engineers, during the project, and they have been answered according to this project and the Volvo Cars Engine factory

5.1 Conclusion

This thesis was about finding classification methods for production data. After searching different previous studies, effective items in the classification were specified. Each item was surveyed separately, and relations between them and classification methods were specified. This research showed that surveying these items and their relationship is very important and necessary for finding a suitable classification method.

Many different classification methods were discussed in chapter 2.2, but for this project, hierarchical classification was selected. This selection is because of two reasons. The first reason is a multi-attribute classification need, and the second reason is the classification method that the Teamcenter software uses. The hierarchical classification method is modeling a hierarchical structure of information (information includes data and documents) based on attributes. Because of this, the most important data and documents in the Volvo Cars Engine factory were collected and their features were specified. Then, the data map of the factory was generated. The data map of a factory shows the diffusion of data in different phases of the product life cycle.

After that, with the availability of data, attributes and classification method, the procedure of entering these objects to the Teamcenter software and using them was evaluated.

The main objective of this project is defining a classification methodology for Volvo Cars Engine factory’s data. By studying all effective items, the factory's data classification methodology was prepared. In this methodology, all data were grouped into three classes (product, process and resources). Then each class was divided into lower classes, as explained
in chapter 3.4, and their attributes were defined according to the data that was collected before.

Then, the classification methodology, hierarchy and attributes were ready to be implemented in the Teamcenter software. For testing of the applicability of this method, a typical bicycle was considered as the product, and the classification method was implemented in the Teamcenter software for this product. Some data was entered into the software, and this data was classified. This test showed that the provided data classification methodology, is applicable for a simple product like a bicycle, and it can be implemented in the Volvo Cars Engine factory.

5.2 Discussion

During the completion of this thesis, some questions arose in different steps that they must be answered for passing those steps. These questions were answered according to this project and the Volvo Cars Engine factory. For finding these answered lots of researches happened and several discussions happened. Questions and answers are as follows.

1- **What are the effective items in the data classification method?**

   In normal data classification, data and attributes are two main items that are effective in finding a data classification method. But in this project there have been two other items that make some restrictions for finding that method. The Volvo Cars Engine factory is one of the items, because all data should be gathered from this company’s documents and information. The other restriction item is Teamcenter as PDM software. There are specific procedures in this software for classification and searching. Without considering these procedures, the classification method is not beneficial.

2- **Where are product data and information?**

   In this project, many meetings were held for gathering data from different factory disciplines. Data and information are extracted from three different types of resources.

   - Documents: This type of resources contains formal data like standards, specifications and manuals. There are a lot of useful data in documents for generation of class attributes.

   - Engineering and specialized software: After the utilization of different software programs in industries, a lot of data and information has been saved in these
software programs. In the Volvo Cars Engine factory, also many specialized software programs have been used like the Teamcenter and Plant Simulation. These software programs are storing a lot of data and information.

- Computer files: There are many files on a company’s computer that only have been stored in that computer and they are not organized in a stored system. Different employees create them and use them individually. It is possible to find useful data in these types of files.

3- **What is the suitable classification method for product life cycle data?**

Because of the need for multi-attribute classification in this project, the hierarchical classification was selected. According to the effective items that have been explained before, a classification method starts from creating the hierarchy and defining classes. The first level of the hierarchy is divided into three branches: Product, processes and resources. These branches are again divided into different branches as explained in section 3.4 of this thesis. Then, attributes are specified for these classes according to the matter of the classes.

4- **According to which criterion should data classification be done?**

Data should be categorized according to its attributes for its most effective and efficient use and quick and easy search. For effective and efficient use, the hierarchy should be designed in such a way that users can find their desired class easily. Classes should be distinctly different from each other so that users can select their classes certainly. For quick and easy search, the attributes must be defined correctly. Attributes of classes are the basis of searching for users.

5- **What are the main benefits from data classification?**

Most new product offerings include numerous design elements used in earlier products. Typically, these re-usable elements constitute up to 85% of the new design content, and inefficient design re-use has a huge impact on cost & time-to-market. When data have been classified, then time to market can be reduced, with possibilities of classification to re-use related product and process data in new projects or programs.

Manufacturing engineers can also use these capabilities to evaluate alternative and substitute parts before finalizing their designs. Therefore, data classification retrieves and reduces costs, by allowing to reuse existing parts and tools, and saves time by using of the generated data.
With data classification, the better visibility of the classified objects within the company will be provided, and duplicate or outdated parts will be visible, and can be eliminated.

By allowing users to manage previously designed, validated and approved product and tool definition data, the reuse capabilities expedite product development, reduce its cost and lessons learned into their new product initiatives.
6 Future work

According to the surveys conducted during the thesis period about the Volvo Cars Engine factory and its product data management project, lots of requirements and deficiencies became clear. Some discussions took place, about the next steps of the PDM project with the project manager and the engineers that are involved in this project. Finally, the future work can be divided into two parts. The first part is about the future work in data classification and implementation of the method in the Volvo Cars Engines factory. The second part is the future work in data management and the Teamcenter project, in VCE. They will be described below.

6.1.1 Future work in classification

With the completion of this thesis, the classification method is specified. Affective items were shown and described. Some of the classes were defined as models for creating next classes.

The next step for VCE, in classification, is defining all classes and abstract classes in all levels of the classification hierarchy. For better execution of this task, several meetings with different disciplines of factory are needed. When an abstract class is divided into several classes, complete knowledge about those classes and abstract classes is required.

In the classification of resource data, a complete list of resources such as machining tools, gauging tools, cranes, etc. is needed. Then all specifications and data that relate to each type of these resources should be defined for creating the classification hierarchy and defining class attributes. Finding comprehensive and appropriate attributes for classes is also placed in the future work of classification. When the classification hierarchy design is finished, and classes are completed, then all resource data and drawings should be entered into the Teamcenter software.

About the process classification, the first step is preparing a list of all types of processes that are performed in the Volvo Cars Engine factory. Then, from the collected data and documents, the data and information that states process features should be extracted, and the classification hierarchy and classes attributes must be defined in the software. For the last step, the processes that are generated before should be classified.

For the classification of product data, it is suggested that some meetings should be held with the design department, for classification of some standard parts. This classification helps designers to re-use data that has been already generated.
6.1.2 Future work in product data management project at VCE

As mentioned before, a project was defined in the Volvo Cars Engine factory about implementation of the Teamcenter product data management software. This thesis is one step of this project, and as you have it here in Figure 3. Volvo Cars Engine PDM Project steps “Volvo Cars Engine PDM Project steps”, it is one of the early stages of this project. Within these PDM implementation project steps, another thesis was completed, and it was about the effective use of product, process and resource modeling in product data management systems (Thijs, 2012). It helped the Volvo Cars Engine factory to find the best way to generate the “Bill of process”, and in that report the classification method was identified as one of the next steps for the Volvo Cars Engine factory.

In this chapter, the next steps for the implementation of the Teamcenter software in the Volvo Cars Engine factory will be explained and the most important applications that relate to process engineering will be listed.

As shown in Figure 3, the next steps of the Volvo Cars Engine factory PDM project are implementation of bill of process for multiple plants and line balancing.

With the implementation of multiple plants BOP, there will be the flexibility to control the propagation of a change affecting multiple plants. When a change affects multiple plants, there must be the full flexibility and control of how the changes must be propagated to each and every plant.

With the help of Teamcenter line balancing abilities, the distribution of workloads to assembly stations can be simply optimized. Users can ensure that the lines are balanced and resources are efficiently utilized. By using an interactive chart, users can clearly visualize stations that are overburdened, and they can distribute the operations from one station to another using a simple drag-and-drop function and quickly achieve a balanced production line.

The classification application that was explained before is not the only useful application in process management. The Teamcenter software has lots of applications that give many abilities to the VCE for managing its processes. Some of these applications are explained below (Siemens(2), 2012).

**Manufacturing Process Planner:** Allows users to design a detailed plan how to manufacture a product that is an assembly. The manufacturing process plan includes a top-level structure of
the process needed to manufacture the product, as well as a detailed design of the individual processes and activities included in the plan.

**Part Planner:** Allows users to design a plan that shows in details how to manufacture a piece part product, such as a piston or engine fan blade. You can plan the production process from raw material to end product, including cutting, drilling, milling, turning, and quality checking operations. In this environment, there is typically no production line, but several resources such as machine tools, fixtures, cutting tools and gages are necessary.

**Plant Designer:** Allows users to create, modify, import, and export a factory structure.

**Resource Manager:** Allows users to store and retrieve resources such as tools, fixtures, machines, and process templates. Resource data is held in a database that is accessible to all users. Resource Manager stores this data in combination with classification information, and it is organized into a hierarchy that is specific to your company.

**Resource Browser:** Allows users to retrieve classification-related data, such as a hierarchy with corresponding groups, classes, and instances from a Teamcenter database when working in an external application. Resource Browser is delivered as a dynamic link library (DLL). To use it within an external application, users must develop code that uses this library.

**Report Generator:** Allows users to create reports about the manufacturing process plan and related operations, activities, product structure, and plant structure.

**Multi-Structure Manager:** Allows users to capture the data in a Teamcenter process or product structure into a collaboration context object for review, sharing, or comparison. Users can also create compositions and perform basic structure editing tasks with this application.

**Process Simulate:** The Process Simulate application allows users to design, analyze, simulate and optimize manufacturing processes from the factory level down to lines and work cells. It provides optional advanced capabilities that are not available in other Manufacturing Process Management applications. Process Simulate is fully integrated into Teamcenter, and you can send data to it from other applications.

In addition to the above applications, there is software that had been mentioned in this report frequently. The “Business Modeler IDE (Integrated Development Environment)” is a tool for configuring and extending the data model. The data model objects define the objects and rules
used in Teamcenter. As mentioned before in chapter 3.3.4, the Business modeler IDE software is needed to create required items and forms, with the desired attributes.

With the completion of data classification in the Volvo Cars Engine factory, data will be classified according to their attributes for its most effective and efficient use and after completing the PDM project, all data that are generated in different phases of the product life cycle can be managed.
References


[Accessed 03 11 2013].

[Accessed 31 03 2013].


[Accessed 11 04 2013].


[Accessed 11 04 2013].


Appendixes

Appendix 1: Process Sheet

<table>
<thead>
<tr>
<th>SKÄRDATA FÖR</th>
<th>OPERATION</th>
<th>I BEARBETNINGSFÖLJD</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nr</td>
<td>kaina</td>
<td>kaina</td>
</tr>
<tr>
<td>0012</td>
<td>07143</td>
<td>644183</td>
</tr>
<tr>
<td>0013</td>
<td>07144</td>
<td>644183</td>
</tr>
<tr>
<td>0014</td>
<td>07144</td>
<td>644183</td>
</tr>
<tr>
<td>0015</td>
<td>07144</td>
<td>644183</td>
</tr>
<tr>
<td>0016</td>
<td>07144</td>
<td>644183</td>
</tr>
<tr>
<td>0017</td>
<td>07144</td>
<td>644183</td>
</tr>
<tr>
<td>0018</td>
<td>07144</td>
<td>644183</td>
</tr>
<tr>
<td>0019</td>
<td>07144</td>
<td>644183</td>
</tr>
<tr>
<td>0020</td>
<td>07144</td>
<td>644183</td>
</tr>
<tr>
<td>0021</td>
<td>07144</td>
<td>644183</td>
</tr>
</tbody>
</table>

**OBSERVERA**

Diagrammet är en innehållsnotering för VOLVO, det kan inte tolkas som teknisk eller tjänsteförrättarinformation.
Appendix 2: Hole number specification
## PFMEA

### FELEFFEKTSANALYS/FAILURE MODE AND EFFECT ANALYSIS PROCESS

<table>
<thead>
<tr>
<th>Operation</th>
<th>Failure Type</th>
<th>Possible Causes</th>
<th>Possible Effects</th>
<th>Control</th>
<th>Priority</th>
<th>RPN</th>
<th>Recommended Action</th>
<th>Decision Taken</th>
<th>Next Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1.3</td>
<td>Shaft breakage</td>
<td>Wrong position of the deep hole</td>
<td>Engine breakdown</td>
<td>2, 6, 6</td>
<td>50</td>
<td>Set interval for checking position.</td>
<td>Checking 130 during startup of production</td>
<td>2, 6, 3, 0</td>
<td></td>
</tr>
</tbody>
</table>
Appendix 4: Most important collected documents and data that are used in Volvo Cars Engines factory:

1. Wanted industrial Position (WIP)
2. Technical strategy
3. Product content file
4. Restriction Model
5. Process requirement description (PKB)
6. Design failure mode and effects analyzer (DFMEA)
7. Process failure mode and effects analyzer (PFMEA)
8. Quality control plan (Styrplan)
9. First piece control (Första Bit)
10. Measurement System Analysis (MSA-Mätbåge)
11. PV-Test
12. Cutting tool drawings
13. Cutting tool list
14. Wear parts (Slitdelar)
15. Operations time (Time study)
16. Hole number specification (Hålbildsritning)
17. Machine Layout (Maskinlayout)
18. Fixturing
19. Flow Chart
20. Control instruction (Kontroninstruktior)
21. Standards
22. Process sheets (Operationsritning)
23. Activity instruction (VI-45)
24. Capability Study
25. Operator instruction sheet (OIS)
26. Work element sheet (WES)
27. NC Program
28. Measurement program
29. Flow Simulation
30. Operation Instruction (PKI)
31. FU operator instruction (Fu instuktioner)
32. Technical Specification (TS)
33. Functional Description assembly Equipment (FKB Assembly)
34. Functional Description machinery Equipment (FKB-machinery)
35. Functional test machining
36. Functional test assembling
37. Operational reliability electrical (FFDS1100)
38. Operational reliability mechanical (FFDS1200)
39. Packing drawings
40. Packing instruction
41. Gauging tools list
42. Gauging tool drawings
43. Quality control instruction (Kontrollinstruktion)
44. KISA
45. Customer remarks