Master Degree Project

UNIVERSITY OF SKÖVDE

SUPPORT COMPONENT REUSABILITY BY INTEGRATING AUGMENTED REALITY AND PRODUCT LIFECYCLE MANAGEMENT

Master Degree Project in Industrial Systems Engineering
One year Level 22.5 ECTS
Spring term 2016

Raquel Quesada Díaz

Supervisors: Jan Oscarsson
Candidate Iman Morshedzadeh
Examiner: Tehseen Aslam

Abstract

In an ever changing market that expands continuously and where innovations cycles become shorter, there is an important increase of the renewal frequency of the electrical and electronic equipment (EEE) and vehicles. This makes the manufacture of EEE and vehicles a fast-growing source of waste in terms of used products. The immense amount of information generated by all these technological products which are currently in the market must be managed throughout the whole life cycle of the products. The problem is to provide information about the technological product's reusability in the recycling process given the colossal complexity of many products and the lifespan of operation. This includes instructions about the components qualifications as elements in a new product. Technologies such as augmented reality (AR) combined with product lifecycle management (PLM) systems can provide the platform for an information system that provides the necessary information and support for the decommissioning process of EEE and vehicles at the end of their life cycle.

The present project describes the framework of integration between AR and PLM with the purpose of recycling a technological product at the end of its life cycle. The proposed method of integration could be considered to constitute both an innovation and a possible improvement if compared with the current approach. It is believed that the development of a method that addresses the issue of integration between AR and PLM could provide with a secure, efficient management of stored data related to various products and their properties related to the recycling process at the end-of-life of the product. The result of this approach is an AR-PLM system architecture which assists the circular economy's recycling process by the use of visual information superimposed on the physical technological equipment.

Keywords: augmented reality, product lifecycle management, sustainability, remanufacturing, integration, WEEE, ELV, RoHS recycling, PDM, CAD, computer vision, tracking, interactive workspace, decommission, diminished reality and wearable computer.

HÖGSKOLAN
I SKÖVDE

Acknowledgements

The research and work behind this thesis was carried out during the spring semester of the year 2016 at the Department of Automation Engineering of the University of Skövde, Sweden.

First and foremost I gratefully acknowledge the support of my main supervisor Dr. Jan Oscarsson for his encouraging, great support and patience throughout the duration of this thesis. Your guidance has been fruitful and rewarding.

In addition, I want to thank my examiner Dr. Tehseen Aslam for his support, advice and the opportunity to perform this project.

Also, I would like to thank my co-supervisor PhD candidate Iman Morshedzadeh for his guidance and good will during this thesis work.

Moreover, I would like to express my gratitude to PhD candidate Oscar Danielson and Marcus Barstorp for their support, guidance and advice regarding AR technologies and Java programming respectively.

Last but not least, I would like to thank my family and friends for supporting me in general, and in my work and studies in particular. Thank you all for being there at all times.

Skövde, May 2016

Raquel Quesada

HÖGSKOLAN I SKÖVDE

Certify of Authenticity

Submitted by Raquel Quesada Díaz to the University of Skövde as a Master Degree Thesis at the School of Technology and Society.

I certify that all material in this Master Thesis Project which is not my own work has been identified.

Signature.

Raquel Quesada Díaz

HÖGSKOLAN I SKÖVDE

Table of Contents

1 Introduction 1

1.1 Background 1

1.2 Goals 2

1.3 Limitations 3

2 Method 4

2.1 Philosophical Paradigm 4

2.2 Research Methodology 4

2.3 Process Diagram 6

3 Frame of Reference 8

3.1 Nomenclature 8

3.1.1 According to Directive 2012/19/EU 8

3.1.2 According to Directive 2008/98/EC 8

3.2 Circular Economy 9

3.3 Global Waste Electrical and Electronic Equipment (WEEE) or E-Waste Management 12

3.3.1 European Union E-Waste Recycling Policy 15

3.3.2 United States of America E-Waste Recycling Policy 18

3.3.3 Asia E-Waste Recycling Policy 19

3.4 Product Lifecycle Management 21

3.4.1 Product Life Cycle Stages 22

3.4.2 Traceability 23

3.5 Augmented Reality 24

4 Literature Review 26

4.1 PLM, Circular Economy and Green Manufacturing 26

4.2 AR Application to the Product Life Cycle in Manufactory 29

4.2.1 AR Application for Assembly Instruction and Training 29

4.2.2 AR Application to Inspection and Maintenance 31

4.2.3 Augmented Revamping and Decommissioning 32

4.3 Augmented Reality, Product Lifecycle Management and Circular Economy 33
4.4 Review Summary 34
5 WEEE Disassembly and Recycling Process 44
5.1 WEEE and ELV Recycling Process 44
5.2 Example of Treatment Processes for WEEE and ELV 45
5.2.1 Waste Mobile Phone Treatment Process 45
5.2.2 ELV Treatment Process 46
5.2.3 Waste LCD TV Treatment Process 47
6 PLM Architecture and AR Integration 48
6.1 AR-PLM Integration Framework Requirements 48
6.2 Product Identifiers and PEID Architecture 49
6.3 AR System Architecture 51
6.4 PLM System Architecture 52
6.5 Integration Issues 54
6.6 Level of AR-PLM Integration 55
6.7 PLM Tiered Architecture 55
6.8 Middleware Software 57
6.9 Service Oriented Architecture (SOA) 58
7 AR-PLM Interface Components 60
7.1 AR System Architecture 60
7.2 Product Identification, PEID and Transfer Protocol 61
7.2.1 Teamcenter Product Identifiers 61
7.2.2 Case of Study: Berg Go-Kart 62
7.2.3 PEID 65
7.3 Communication Protocol and Level of Integration between AR and Teamcenter 66
7.4 Teamcenter System Architecture 67
7.5 Teamcenter Customization and Application Integration 71
7.6 Teamcenter Service Oriented Architecture 72

7.7 Application Framework for the Integration between AR and Teamcenter 73
7.8 Results 77
8 Conclusions 81
8.1 Summary 81
8.2 Conclusions 82
8.3 Future Work 85
9 References 87
Appendix A: Categories of EEE 100
Appendix B: Properties of Hazardous Waste 101
Appendix C: AR-TC MainCode.java 103
Appendix D: AR-TC Arguments.java 113
Appendix E: AR-TC AppXSession.java 116
Appendix F: AR-TC AppXCredentialManager.java 117
Appendix G: AR-TC WebcamQrCodeScan.java 118
Appendix H: AR-TC WebcamViewPanel.java 121
Appendix I: AR-TC WebcamMessagePanel.java 141
Appendix J: AR-TC log4j.properties.xml 143
Appendix K: AR-TC oppolicy.xml 144

HÖGSKOLAN I SKÖVDE

Table of Figures

Figure 2.1. Process diagram of the Thesis 7

Figure 3.1. The circular economy (Ellen MacArthur Foundation, 2012) 10

Figure 3.2. Existing scenarios in WEEE collection (Baldé, et al., 2015) 14

Figure 3.3. EEE put on the market and WEEE collected and treated in EU from 2007 to 2012 (EUROSTAT, 2016) 15

Figure 3.4. Recovery and recycling rate for end-of-life vehicles (ELV), EU-27 in 2012 (EUROSTAT, 2014)17

Figure 3.5. WEEE generation and recycling in pounds from 2000 to 2012 in the United States (EPA, 2014) 19

Figure 3.6. Example of Chinese RoHS (ACPEIP) labelling requirements (INGUN, 2015) 20

Figure 3.7. Product Life Cycle Stages 23

Figure 3.8. Progressed Traceability DB (Minami, et al., 2004) 24

Figure 3.9. Maintenance task on a Rolls Royce DART 510 engine (Gallagher, 2015) 25

Figure 4.1. Green product lifecycle framework (Vila, et al., 2015) 28

Figure 4.2. Diminished reality in a revamp procedure. a) Original, b) Diminished, c) Augmented (Zokai, et al., 2003) 32

Figure 5.1. WEEE general recycling process diagram (Chancerel & Rotter, 2009; United States Environmental Protection Agency (EPA), 2016) 44

Figure 5.2. Recovery and remanufacturing of waste mobile phone WEEE (Lu, et al., 2014) 45

Figure 5.3. ELV treatment process (García, 2011) 46

Figure 5.4. Recovery and remanufacturing of LCD TV WEEE (Li, et al., 2012) 47

Figure 6.1. QR code-based product tracking mechanism (L., et al., 2014) 48

Figure 6.2. Example of GS1 standardised barcodes (IndiaFilings, 2015) 49

Figure 6.3. Example of product identification label (Truesmart.com, 2016) 50

Figure 6.4. General architecture of an active PEID (Kiritsis, 2007) 50

Figure 6.5. Example of AR displays, a) spatial, b) hand-held and c) head mounted 51

Figure 6.6. General PLM system architecture 52

Figure 6.7. Examples of PLM tiered architecture 56

Figure 6.8. SOA runtime deployment n-tiered PLM system (Chesney, 2012) 58

Figure 7.1 Penny C Wear Interactive Glasses Extended 20 60

Figure 7.2. Berg Go-Kart 62

Figure 7.3 Bill of materials (BOM) of product 63

Figure 7.4. Example of a recycling instruction 64

Figure 7.5. Recycling instruction with disassembly process 64

Figure 7.6. 3D CAD models of the Go-Kart and a part of its chassis 65

Figure 7.7 Example of QR code and barcodes format for the demonstrator 66

Figure 7.8. Teamcenter system architectures available 68

Figure 7.9. Teamcenter reference implementation architecture (Singh, V., 2008) 69

Figure 7.10. Teamcenter J2EE four-tiered architecture (Singh, V., 2008) 70

Figure 7.11. Teamcenter architecture framework 73

Figure 7.12 Eclipse interface with the necessary Java packages created for the AR-PLM integration (The Eclipse Foundation, 2016) 74

Figure 7.14. Proof of concept evaluation and display of TXT and PNG information 78

HÖGSKOLAN I SKÖVDE

Index of Tables

Table 3.1. Relation between Law Restrictions and Production Process (Minami, et al., 2004) 13

Table 4.1. List of AR papers and their contents. Part I 37

Table 4.2. List of AR papers and their contents. Part II 38

Table 4.3. List of AR papers and their contents. Part III 39

Table 4.4. List of AR papers and their contents. Part IV 40

Table 4.5. List of PLM papers and their contents. Part I 41

Table 4.6. List of PLM papers and their contents. Part II 42

Table 4.7. List of PLM papers and their contents. Part III 43

Table 6.1. Commercial PLM systems (PLM Technology Guide, 2008) 53

Table 7.1. Main Code framework for the integration between AR and Teamcenter 75

Table 7.2. Framework of integration between AR and PLM and flow of information exchange 80

HÖGSKOLAN I SKÖVDE

Terminology

A

ACPEIP
Management methods for controlling pollution caused by electronic information products 19

API
Application programming interface 55

App
Application 5

AR
Augmented Reality ..... i, 1

ATF
Authorised treatment facility 42

B

BOL
Begining of life 21

BOM
Bill of materials 32

BOP
Bill of process 32

BPML
Business process modelling language 56

C

CAD
Computer Aided Design 31

CAS
Chemical abstracts service 48

CAS#s
CAS registry numbers 48

CBS
Command button system 58

CEA
Consumer Electronics Association 17

CRM
Computer relationship management 20

CRT
Cathode ray tubes display 17

CSM
Computer supplier management 20

D

DB
Database 69

E

ECUs
Electronic control units 49

ED
Eco design or design for environment 27

EDI
Electronic data interchange 56

EEA
European Economic Area 14

EEE
Electrical and electronic equipment ..... i, 1

EIP
Electronic information products 19

EJBs
Enterprise JavaBeams 67

ELV
End-of live vehicles 1

ELV Directive
Directive on End-of life vehicles 15

EMARS
Environmental material aggregation and reporting system 20

EOL
End of life 22

EPA
Environmental Protection Agency 18

ERP
Enterprise resource planning 20

EU
European Union 13

E-Waste

Waste of EEE, Waste of electrical and electronic equipment 11

F

FEM
Finite Element Method 32
FMS
File management system 66
FOV
Field of view 58

G

GD
Green development 27
GLN
Global location number 48
GM
green manufacturing or sustainable manufacturing 27
GPRS
General packet radio service 49
GTIN
Global trade item number 48
GUI
Graphic user interface 53
GUPI
Globally unique product ID 48

H

HID
Human interface device 58
HMD
Head mounted display 23
HTTP/S
Hypertext transfer protocol secure 56

I

ICT
Information and communication technologies 20
Information and communication technology 12
IMEI
International mobile equipment identity 48
IT
Information technology 5
ITK
Integration toolkit 65

J

J2EE
Java 2 Enterprise Edition 57
JIS
Japanese Industrial Standard 19
JSPs
JavaServer Pages 67

K

KARMA
Knowledge based augmented reality for maintenance assistance 30

L

LAN
Local carea network 51
LCD
Liquid cristal display 17
LED
Light emitting diode 29
LOVs
List of values 68

M

MES
Manufacturing execution systems 20
MII
Chinese Ministry of Information Industry 19
MMS
Modern manufacturing systems 52
MOL
Middle of life 21
Mt
Million metric tonnes

1 Mt = 1000000 Kg 11
MTS
Motion tracking system 58

O

OEM
Original equipment manufacturer 15

P

PARIS
PDA based augmented reality integrating speech 30
PBB
Polybrominated biphenyls 15
PBDE
Polybrominated diphenyl ethers 15
PDA
Personal digital assistant 30, 49
PDM
Product data management 20
PDML
Product data markup language 56
PDS
Projection display system 58
PEID
Product embedded information devices 46
PEUR
Act on the Promotion of Effective Utilization of Resources 19
PLM
Product Lifecycle management ..... i, 1
POM
Persistent object manager 66
PSR
Product social response 28

Q

QR
Quick response (QR code) 49

R

RFID
Radio-frequency identification 49
RoHS
Restriction of the use of certain hazardous substances ..15
RoHS Directive
European Union Directive on the restriction of the use of certain hazardous substances in electrical and electronic equipment 14
RUM
Responsible use and maintenance 27

S

SB 20 & 50
California State Electronics Waste Recycling Act 17
SES
Spider embedded system 58
SOA
Service oriented architecture 56, 69
SOAP
Simple object access protocol 56
SPL
Sustainable production and logistics 27
SPM
Supply chain management 20
STEP
Standard for exchange of product model data 56
Standard ISO 10303-203
1994 on product data representation and exchange .32

T

TaDiMa
Tangible digital master 33

U

U.S.
United States of America 17
UI
User interface 30
UPnP

Universal plug and play 49

V

VR
Virtual reality 23

W

WEEE
Waste of electrical and electronic equipment 1
WEEE Directive
European Union Directive on waste electrical and electronic equipment 14
WSDL
web services description language 56

X

XML
Extensible markup language 56

1 Introduction

In an ever changing market that expands continuously and where innovations cycles become shorter, there is an important increase of the renewal frequency of the electrical and electronic equipment (EEE) and vehicles, this makes the manufacture of EEE and vehicles a fast-growing source of waste. The present project describes the framework of integration between Augmented Reality (AR) and Product Lifecycle Management (PLM) with the purpose of recycling a technological product at the end of its life cycle. The result of this approach will be an AR-PLM system architecture which assists the circular economy's recycling process by the use of visual information superimposed on the physical technological equipment.

1.1 Background

The waste of electrical and electronic equipment (WEEE) and the end-of life vehicles (ELV) are both a complex combination of components and materials that can cause major environmental and health complications due to their content of hazardous elements. Moreover, these products possess high value as recoverable parts or components that can be reused for the manufacture of new products and produce additional benefits for the manufacturer. In addition, the reuse of WEEE and ELV components contributes to the enhancement of the resource efficiency and the reduction of the consumption of energy. As a result of this, the concept of Circular Economy emerged in order to support a sustainable environment. One of its most essential sustainability aspects is the WEEE and ELV management, which aims at the reduction of energy and material consumption by the reuse, recovery, and treatment of product components at the product's end-of-life. If it is considered the immense number of EEE products and vehicles in the market, each of which may have several variants and revisions that belong to a series produced in batches, the amount of information generated, stored and tracked is enormous. The problem is to provide information about the technological product's reusability in the recycling process given the colossal complexity of many products and the lifespan of operation. This includes instructions about the components qualifications as elements in a new product. One key issue is the identification of a specific component including features such as component revision, variant, etc. Technologies such as AR combined with PLM systems can provide the platform for an information system that provides the necessary information and support for the decommissioning process of EEE and ELV.

This project studies AR and PLM and analyses concepts from these techniques in order to identify how these concepts and theories can be used to support the circular economy. The result of this approach will be an AR-PLM system architecture which assists the circular economy's recycling process by the use of visual information superimposed on the physical technological equipment. The overlay of virtual information on real world objects can enhance human's perception of reality, hence AR-PLM interface could enable the improvement of the recycling process and supports operators in the industry in the optimal accomplishment of their tasks. The more complex the technological equipment is, the more complex the recycling operations and the greater the risk for the operation to be performed incorrectly. The use of the AR-PLM interface as a support tool during the recycling process, could provide the workers with a visual guidance easy to follow. The instructions supplied could help the operator to make the correct recycling decision at the same time that the task performance is optimised and the efficiency of the process is improved.

1.2 Goals

Each EEE and ELV product has a different background and requirements which must be taken into consideration in the design of the architecture for the AR-PLM interface. PLM applications provide generic features mainly from the product data management perspective, while addressing unique PLM requirements. The customization of the application kernel to the context of AR and sustainability involves the possible adaptation of PLM interface with the addition of new features which will meet the specifications for the AR-PLM implementation. In each and every organization data security is an important issue for performing the different tasks in the organization. Therefore, each user has specific rights on the various objects or items in the organization.

A large number of research projects, see paragraph 4.2, have been reported to use AR and PLM in order to train operators for assembly, training, inspection, maintenance, or decommissioning tasks among others. However, this methodology is normally not being used for the combination of AR and PLM with the purpose of recycling a technological product.

This project will provide with a suggestion for a possible architecture for the implementation between AR and PLM with the purpose of recycling EEE and ELV. The definition of this framework will assist the recycling process by the display of visual information superimposed on the physical assembly parts that must be recycled. The AR-PLM interface could reduce disassembly times, accelerate learning of the recycling task and provide with more quality assurance to the recycling process.

The main purpose of the present thesis is to aggregate knowledge about the relation between PLM, AR and circular economy by the definition of a framework that allows the use of AR in a conventional PLM system with the purpose of recycling a technological product at the end of its life cycle. The development of a method that addresses the issue of integration between AR and PLM could provide a secure, efficient management of stored data related to various products and their properties regarding the recycling process at the end-of-life of the product.

In order to achieve this general goal, this project can be further refined into the subsequent steps:

1. Apply data collection techniques (e.g. documents, interviews, observations, etc.) in order to obtain information regarding the main topics of the project (AR, PLM and recycling WEEE and ELV) and related to the possible development of the AR-PLM interface.
2. Analyse and evaluate the obtained information so it can be determined if it is possible to define and develop a framework for the said interface.
3. Introduce and define the necessary requirements and specifications for the AR-PLM integration with the purpose of recycling WEEE and ELV.
4. Define and evaluate the requirements for the implementation of the proposed framework in the PLM system Siemens Teamcenter and develop a demonstrator as a proof of concept.

1.3 Limitations

The present thesis counts with some limitations that must be considered. The University of Skövde counts with two types of internet connection accessible for this project, Eduroam Wi-Fi and the local area network (LAN). Due to security reasons these two networks cannot communicate with each other to preserve the integrity of the system. This poses a problem with some of the AR devices available since their main technology uses Wi-Fi consequently, that equipment has been discarded. In addition, although a demonstrator for the proposed framework of integration between AR and PLM will be developed, this prototype may be incomplete and will be implemented only as a proof of concept to test the feasibility and verify the validity of the method of integration. Since further development of the proposed framework is likely to be done and tested in future work, this demonstrator will not be considered as an early version of the product nor tested to verify the potential of utilization. Moreover since the present thesis is a preliminary study and it is expected to be improved in future studies, there will not be any study or analysis about the production and usage costs.

2 Method

The general goal of the project is to solve a problem: create a framework of integration between AR and PLM that allows the recycling of technological products within the specifications and requirements of circular economy and sustainability.

The present chapter describes the research methodology used to address the solution of the problem that motivates the realization of the present thesis. This research methodology will provide the necessary guidelines to fulfil the main goal of the project.

2.1 Philosophical Paradigm

The process of design and creation of the framework of implementation between AR and PLM will require the use of a combination of the hypothetico-deductive method (Oates, 2006) and the inductive method (Afisi, 2009), as a basic methodology for a scientific research in which the collection qualitative data will be evaluated in order to develop the AR-PLM interface. According to Afisi, prediction is embedded in probability which is one of the methods used in induction and therefore, induction enables the prediction of the outcome of a scientific test. As a consequence, scientific researches and discoveries use a combination of induction and deduction to reach conclusions (Afisi, 2009). This means that the solution of the problem will require the generation of one or various hypotheses that will allow the implementation of the said interface, and the definition of procedures that will provide with an explanation of the solution. Once this is done, the hypothesis and procedures must be tested to prove the validity of the proposed solution. A prototype will be developed as a proof of concept to test the validity of the proposed AR-PLM interface. The prototype will be modified and tested until the outcome proves the suitability of the concepts and procedures described to implement the framework of integration.

As a consequence of the aforementioned reasons positivism is considered to be the most suitable philosophical paradigm for this thesis.

2.2 Research Methodology

In order to address the research questions, an overall research approach must be selected. According to Oates (2006), there is usually one research strategy for each research question. However, there are cases in which the overall research methodology might be composed of more than one research

strategy. This paragraph includes an analysis of the selected research methodology and its suitability for the present thesis scenario.

As stated in previous paragraphs, this thesis has as general goal the definition of an AR-PLM framework that allows the use of AR together with a PLM system with the purpose of recycling a technological product at the end of its life cycle. This means that the research strategy will be focused on the development of a new information technology (IT) artefact by means of the study and evaluation of different existing techniques, concepts and theories employed in AR and PLM, and the posterior selection of an appropriate approach for the implementation and creation of the AR-PLM interface. The project will also include the development of a demonstrator (instantiation) which proves the concept of integration between AR and PLM with a recycling focus. For these reasons, it has been decided that the main research strategy used in the project will be design and creation.

In this case of study, the main domain of the project has not been previously automated, even though, it might incorporate theories or methods developed and/or being used in other disciplines, so its contribution to knowledge and the presented results will be based in literature and previous researches. Which means that different methods and theories will be investigated, analysed, compared and evaluated in order to achieve optimal results for the implementation and design of the AR-PLM interface. Due to the aforementioned reasons, it can be affirmed that this research strategy will have as main focus the IT application (App), which will be the vehicle to something else and the tangible end product, at the same time.

It has been considered the use of survey strategy for the collection of information regarding the three main topics of the project in order to determine the possibility of integration between AR and PLM. In addition, it has been studied the possible realisation of interviews to key personal in companies of the technology and recycling sectors together with a literature review of previous research studies. However, after several attempts to contact companies within the sector and the lack of notorious results, it has been decided that survey strategy and the realisation of interviews will not be a real applicable research method due to the undisclosed policy of the companies. Accordingly, it will be performed an extensive literature review of previous researches and cases of study. The literature review could provide with useful information regarding patterns of actuation and the identification of the different lines of implementation so the posterior selection, analysis and definition of the AR-PLM communication protocol can be used to develop the AR-PLM interface.

The main data collection techniques used in this thesis is documents, and will be implemented by the study and analysis of different types of documents such as books, research generated documents, white papers, secondary data, etc. This technique provides with detailed general and specific information about patterns of actuation, theories and methods previously developed, cases of study formerly analysed, the needed data for the generation of the specifications and requirements for the interface, product identification method, etc. In addition, the use of documents and can also contribute to the credibility of the study.

The solution to the problem will require the generation of one or various hypotheses that will allow the implementation of the said interface, and the definition of procedures that will provide with an explanation of the solution. Once this is done, the hypothesis and procedures must be tested to prove the validity of the proposed solution.

In order to test the validity of the proposed solution, the experiment research methodology could be used to develop a demonstrator that will serve as a proof of concept. Which means that the prototype will be manufactured to test the effectiveness of the AR-PLM interface, which will be further modify and tested until the outcome proves the validity of the design.

2.3 Process Diagram

The organization of this project is as shown in Figure 2.1. First it is included an overview that covers the legal requirements on component recycling. This review will also include aspects on the manufacturer's responsibility for the end-of-life processes. In addition, it will be presented a brief synopsis of AR technology that will be followed by an analysis of related work on the integration of AR and PLM. The main body of the project then discusses PLM and recycling information data flow to content creation and authoring for AR. The two last sections present the implementation and results, as well as conclusions of the work. The implementation will be based on AR and Siemens Teamcenter PLM system.

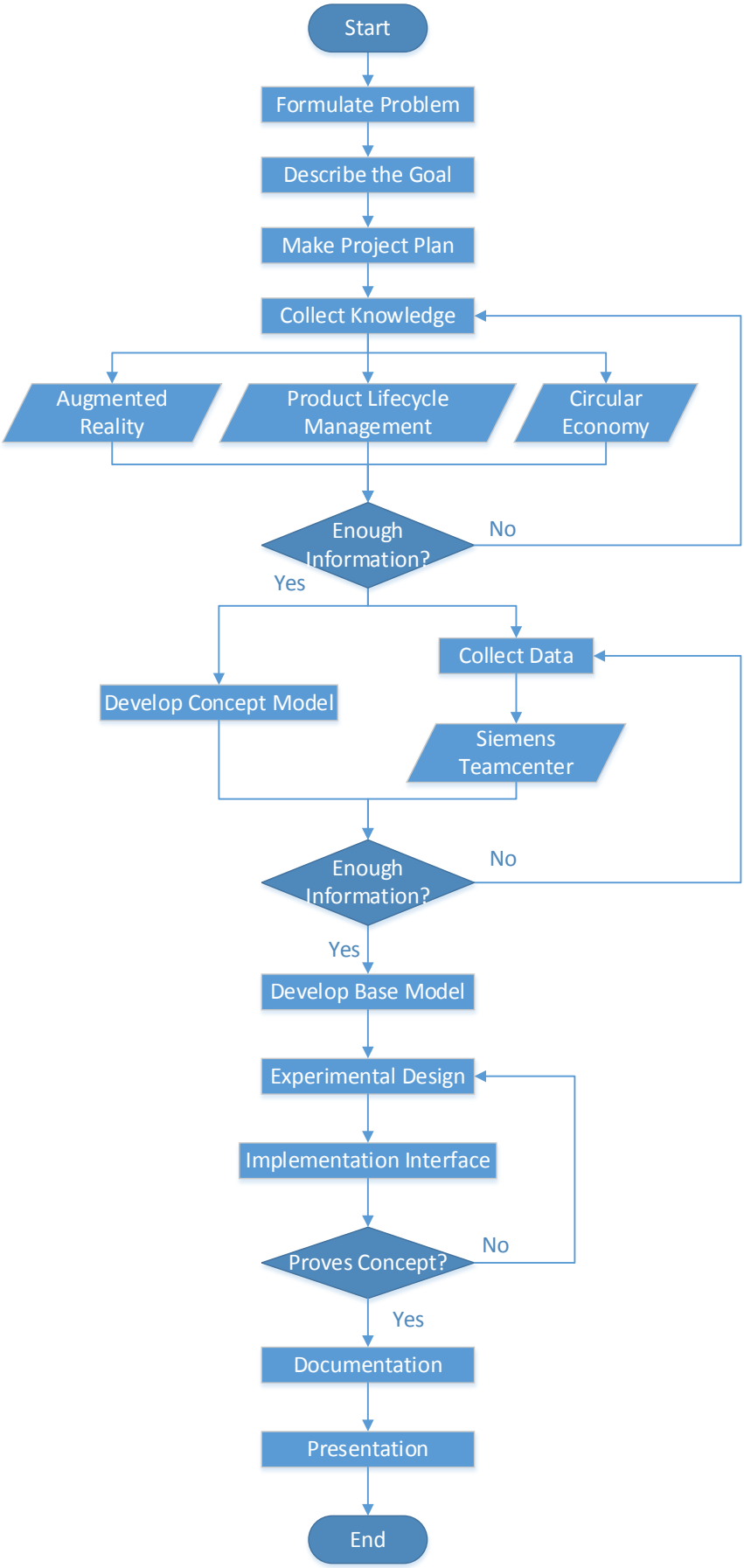


Figure 2.1. Process diagram of the Thesis

3 Frame of Reference

The present chapter presents a summary of the different concepts and definitions related to this project and provides with a context for the literature review presented in Chapter 4.

3.1 Nomenclature

For the purposes of this thesis, the following definitions shall apply:

3.1.1 According to Directive 2012/19/EU

- **Electrical and electronic equipment (EEE):** Applies to equipment which operates with electric currents or electromagnetic fields and equipment which generates, transfers and measures such currents and fields. This equipment shall be designed to not use a voltage rating exceeding 1 000 Volt (V) for alternating current and 1 500 V for direct current.
- **Waste electrical and electronic equipment (WEEE):** Refers to the electrical or electronic equipment considered to be waste within the definition stated in Article 3(1) of Directive 2008/98/EC (see Paragraph 3.1.2), and includes all the components, subassemblies and consumables which composed the technological product at the time of its disposal.
- **WEEE from private households:** Includes the WEEE which comes from both, private households and WEEE from commercial, industrial, institutional and other sources which, due to its nature and quantity can be equated to that from private households. All the EEE waste which is likely to be used by both, private households and users other than private households, shall in any event be considered to be WEEE from private households.

In addition, the definitions of hazardous waste, collection, separate collection, prevention, re-use, treatment, recovery, preparing for re-use, recycling and disposal shall be taken from Article 3 of Directive 2008/98/EC.

3.1.2 According to Directive 2008/98/EC

- **Waste:** Refers to any substance or object which is discarded, intended or required to be discarded by the owner.
- **Hazardous waste:** Relates to every kind of waste that exhibits one or more of the hazardous properties indexed in Appendix B:

(1) Mt = Million Metric tonnes = 1000000 Kg

- **Collection:** Refers to the harvest and preliminary sorting and storage of waste in order to transport it to a waste treatment facility.
- **Separate collection:** It is a collection of waste in which the different waste streams are maintained separated from each other depending on their type and nature so the application of a specific treatment can be implemented.
- **Prevention:** Includes all the measures adopted before a substance, material or product becomes waste, that aim at the reduction of the quantity of waste, the harmfulness of the generated waste to the environment and human health, and the harmful substances contained in WEEE.
- **Reuse:** Applies to any operation by which products or components are used for the same purpose for which they were conceived. This definition includes also the continued use of the equipment or components which has been returned to collection points, distributors, recyclers or manufacturers.
- **Treatment:** Refers to any recovery or disposal operation and includes the preparation prior to recovery or disposal.
- **Recovery:** Includes any of the operations by which waste that would have been otherwise used to fulfil a specific function is used to replace other materials and serves to an useful purpose, and it also includes the waste that has been prepared to fulfil that function, in the plant or in the wider economy.
- **Preparing for reuse:** Includes recovery operations such as checking, cleaning or repair by which waste components or products are prepared to be reused without any additional pre-processing operation.
- **Recycling:** Applies to any recovery operation in which waste is reprocessed into products, materials or substances either for the original purpose or for other purposes. This definition includes the reprocessed organic materials but excludes the energy recovery, which means the use of combustible waste as a means of generating energy through direct incineration with or without other waste but with recovery of the heat.
- **Disposal:** is any operation other than recovery, even if the operation has the recovery of substances or energy as a secondary consequence.

3.2 Circular Economy

The concept of circular economy is based in natural ecosystems and its main goal is to support a sustainable environment by the optimization of the entire system of material flows, see Figure 3.1. The Ellen MacArthur Foundation (2012) defines circular economy as a corrective and reconstructive

economy which aims at the maintenance of the utility and the preservation of the purpose of products, components, materials and general resources at all times, i.e. it aims at the production of goods and services while at the same time reduces the consumption and waste of raw materials, water and energy sources. Figure 3.1 illustrates how a circular economy benefits from products that are designed to enable the combination of biological and technical materials in order to facilitate the disassembly, recovery, reuse and recycling of the materials at their end-of-life. (Ellen MacArthur Foundation, 2012)

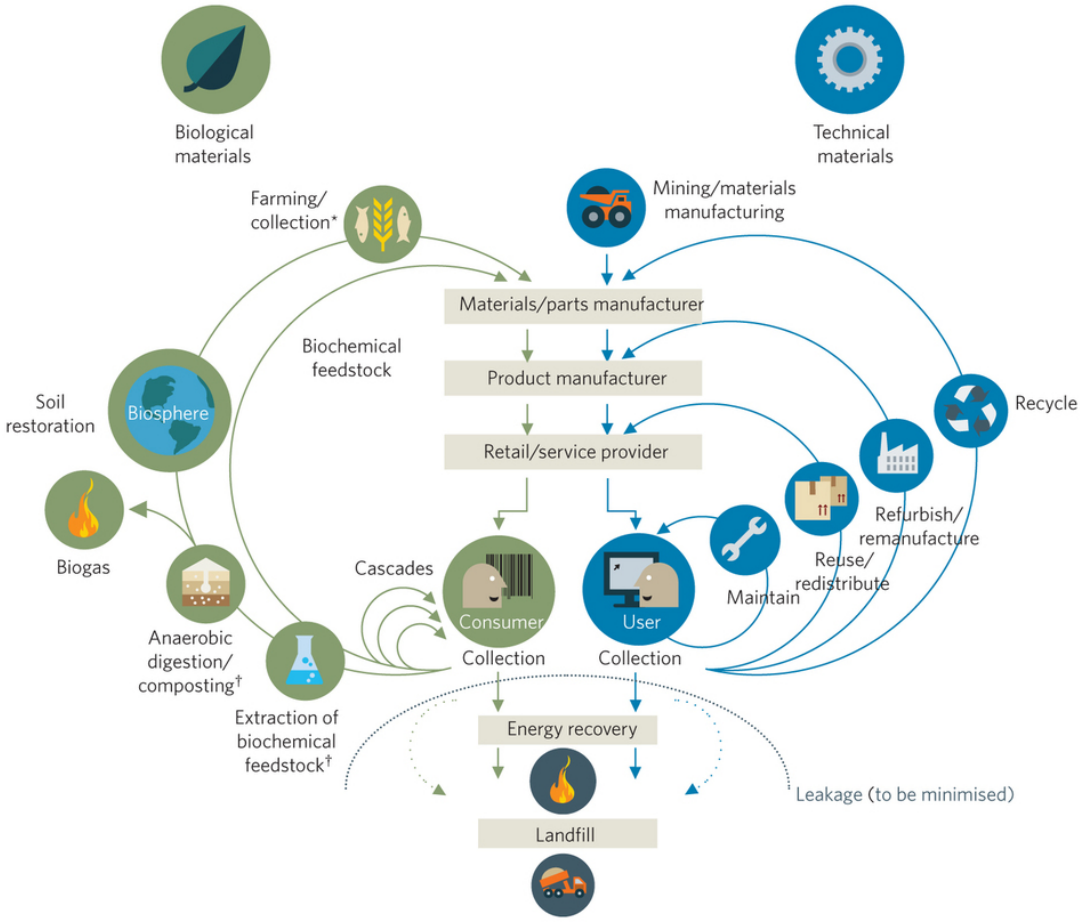


Figure 3.1. The circular economy (Ellen MacArthur Foundation, 2012)

Accordingly, a circular economy diminishes the need for new addition of energy and materials at the same time that minimises the environmental stress connected to resource extraction, emissions and waste. Furthermore, this policy requires the management of natural resources in an efficient and sustainable manner throughout their life cycles. (IMSA Amsterdam, 2013)

As demand of technological products continues to increase, the management of WEEE and ELV has become one of the most essential sustainability aspects of circular economy due to the significant effect this trend has on the resources consumption. In order to counter this situation, it is necessary to

manage the enormous amount of information generated, stored and tracked by these products, and provide the

precise information about the product's reusability or recoverability in the recycling process. A platform of integration between AR and PLM technologies could provide with an information system able to manage, track and store all the information related to a product's lifecycle, and supply the necessary recycling information to the operator. Is for these reasons that the integration of AR and PLM with the purpose of recycling could support the decommissioning process and therefore, the circular economy.

3.3 Global Waste Electrical and Electronic Equipment (WEEE) or E-Waste Management

Electrical and electronic equipment has transformed the way people communicates, entertain themselves or search for information since 1990's. Currently, technological products are ever present in everyday life and can be found in a myriad of places such as households, hospitals, transportation systems, communication networks and offices, among others. However, the technology of these products is constantly innovated, their lifespans are shorter and their sales increase rapidly. In addition, the users consider that technological equipment is obsolete at an earlier stage. As a consequence, WEEE has emerge as one of the waste streams with the fastest growth and become a challenge for waste management due to the quantity of both valuable and hazardous materials that it contains. Accordingly, governments, producers and recyclers from different countries all over the world have increased their efforts, over the last two decades, in order to create specialised collection and treatment systems that expedite WEEE acquisition from final owners so it is processed in professional recycling facilities (Baldé, et al., 2015).

Table 3.1 shows the relation between the different law restrictions and production processes. In the column "related process" the upper case letters indicate the existing regulations and the lower case letters indicate the desired regulation. As a result of these regulations, manufacturers are required to manage all stages of development/design, procurement of materials/components, production, sales, customers, maintenance and recycling/disposal.

In 2015 the United Nations University published a report which established that the global quantity of WEEE reached 41.8 million metric tonnes (Mt)[(1)] in 2014. It is forecasted an increase of the global e-waste to 50 Mt by 2018. Almost a 60% of the bulk of global WEEE in 2014 was discarded kitchen, bathroom and laundry equipment and around a 7% was comprised of personal information and

(1) Mt = Million Metric tonnes = 1000000 Kg

communication technology (ICT) devices as can be personal computers, mobile phones, and printers. (Baldé, et al., 2015)

Table 3.1. Relation between Law Restrictions and Production Process (Minami, et al., 2004)

A Development and design	B Procurement	C Manufacturing	D Sales and maintenance	E Recall/recycling

Region	Name of regulation	Item of regulation	Object products	Object region	Effect date	Related process
Europe	**RoHS**	• **6 restricted substances (Mercury, cadmium, lead, etc.)**	**EEE 100 items**	**Member countries of EU**	**Jul. 2006**	**A, B, c, d**
	WEEE	• **Recycling of EEE products** • **Recycling ratio**	**EEE 120 items**	**Member countries of EU**	**Aug. 2005**	**A,E**
	REACH	• Registration, evaluation, and authorization of chemicals	1t per year or more (each company and product)	Member countries of EU	2006	A
	EUP	• Environmental requirement for product a) Lifecycle assessment b) Energy saving c) Labeling	EEE	Member countries of EU	2006	A
Japan	Home Appliance Recycling Law	• Recall and recycling of used product • Recycling ratio	4 products (TVs, refrigerators, etc.)	Japan	Apr. 2000	A, E
	Law for chemical substances	• Specific chemicals prohibited in products	Specific products	Japan	1990	A, B, c, d
	Japan RoHS	• Under study of Japanese restrictions based on EU restrictions				
	PRTR	• Registration of pollutants in production	354 kinds of substances	Japan	2001	B, C
Asia	Chinese RoHS Chinese WEEE	• 6 restricted substances • Recycling of EEE	EEE	China	Jan. 2006	A, B, c, d, E
	Korean RoHS	• 6 restricted substances	EEE	Korea	2006	A, B, c, d
USA	Proposition 65	• Warning and announcement to customer for 800 kinds of chemical substances	Included products	California	1988	A, B, c, d
	Mercury legislation	• Labeling on products using mercury	Lamps, batteries, etc.	6 states in USA	2000	A, B, c, d
	SB20	• Recall and recycling of used products • 6 restriction substances	Flat panels, CRTs, etc.	California	Jul. 2004	A, B, c, d, E

The total volume of e-waste generated in 2014 was divided in (Baldé, et al., 2015):

- 12.8 Mt of small equipment: microwaves, video cameras, toasters, etc.
- 11.8 Mt of large equipment: dishwashers, photovoltaic panels, washing machines, etc.
- 7.0 Mt of temperature-exchange equipment: cooling and freezing equipment.
- 6.3 Mt of screens.
- 3.0 Mt of small ICT equipment: mobile phones, pocket calculators, personal computers, etc.
- 1.0 Mt of lamps.

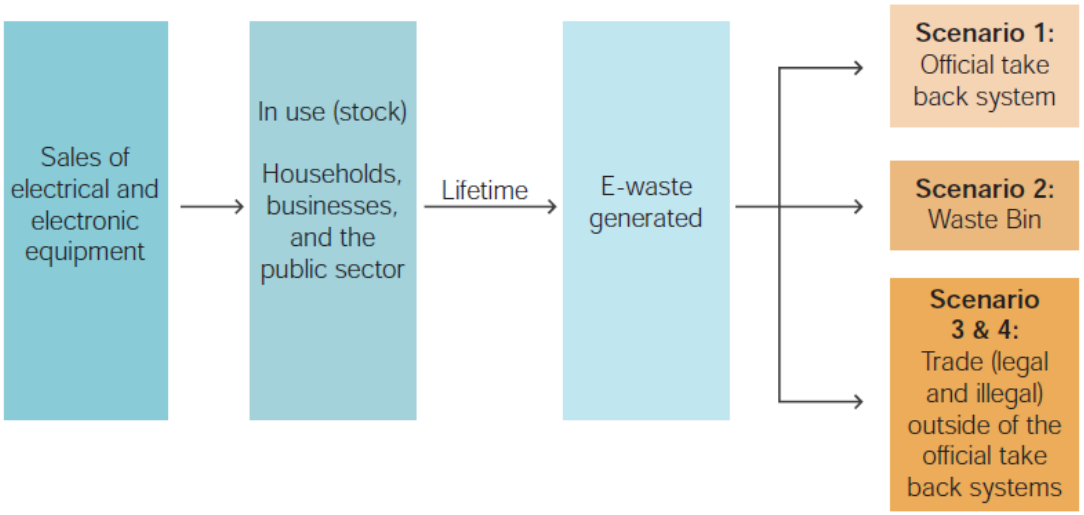


Figure 3.2. Existing scenarios in WEEE collection (Baldé, et al., 2015)

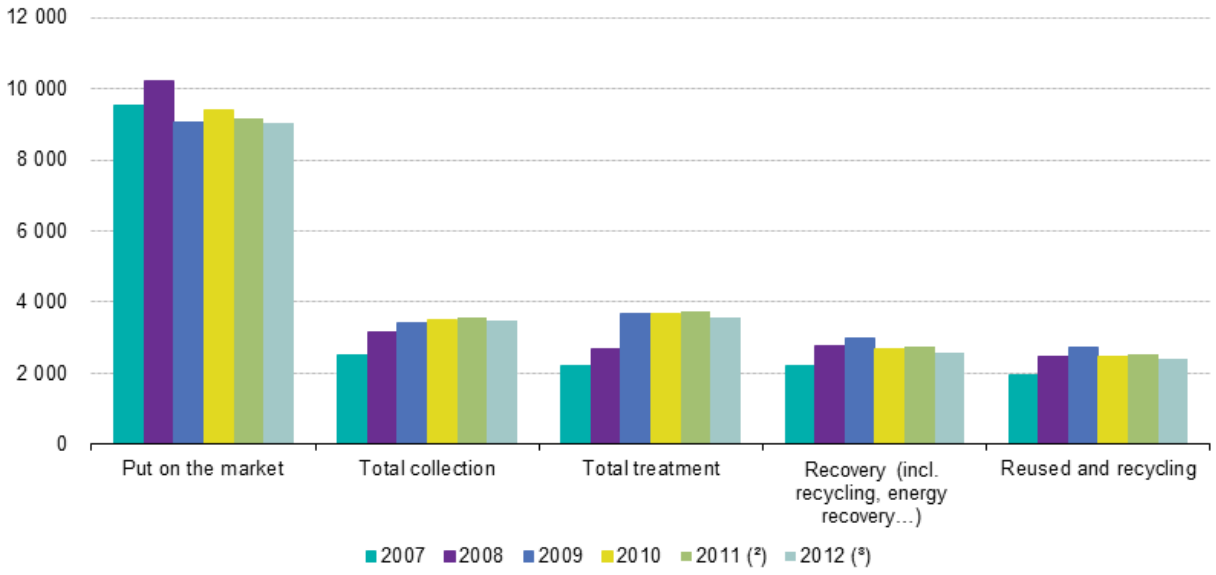
Nevertheless, despite these attempts, there is still a considerable volume of e-waste that is not being assorted or managed efficiently as a consequence of the current limitations of the state of the art treatment systems. Figure 3.2 shows the different WEEE collection scenarios. In 2014 it was reported that around 6.5 Mt of the global e-waste was collected and treated by formal national take-back systems (scenario 1). It was also reported that 0.7 Mt of e-waste were collected from the waste bin in the 28 EU Member States (scenario 2) this quantity is unknown for other regions (Baldé, et al., 2015). Scenario 3 represents the acquisition of WEEE outside the formal national collection system in developed countries. Scenario 4 represents both, the underdeveloped countries that lack a proper system of collection and treatment, and underdeveloped countries that use inefficient techniques to process and extract components and materials from WEEE received from developed countries. For example, it has been estimated that in 2010 Nigeria recycled 0.36 Mt of WEEE by means of inferior standards and that 0.1 Mt of that e-waste was imported from more developed countries (Ogungbuyi, et al., 2012).

Even though WEEE constitutes a potential reservoir of recyclable materials with an estimated value of 48 billion euro in 2014, the high amount of toxic substances present in e-waste from scenarios 3 and 4 could lead to harmful emissions, which represent a hazardous situation for both, the local environment and insufficiently protected workers, and could eventually become an environmental catastrophe in some regions. (Baldé, et al., 2015)

3.3.1 European Union E-Waste Recycling Policy

Due to the reasons stated in paragraph 3.3, the European Union (EU) considered that the hazardous components of WEEE were a major concern and the waste management and recycling phase of the product life cycle was not undertaken sufficiently, and published two directives which entered into force in 2003: The Directive on waste electrical and electronic equipment (WEEE Directive) and the Directive on the restriction of the use of certain hazardous substances in electrical and electronic equipment (RoHS Directive).

The first WEEE directive, Directive 2002/96/EC, main purpose was the promotion of reuse, recycling and other forms of recovery of WEEE by the reduction of the amount of waste to be disposed and the improvement of the environmental performance of the economic operators involved in the process (European Union, 2002). In addition, in 2008 the European commission proposed a revision of the WEEE Directive, Directive 2012/19/EU, and declared the text with European Economic Area (EEA) relevance in order to confront and halt the continuous rapid increase of the EEE waste stream and to ensure that the current regulation targets are achieved (European Union, 2012).



(¹) Includes Eurostat estimates due to missing data for several EU Member States.
(²) Does not include data for Croatia on reuse and recovery.
(³) Includes data for EU-28.

Figure 3.3. EEE put on the market and WEEE collected and treated in EU from 2007 to 2012 (EUROSTAT, 2016)

Figure 3.3 shows an overview of the developments related to WEEE in the European Union under the framework of Directive 2002/96/EC on WEEE. If it is analysed, it can be seen that in 2007 EEE such as computers, TV-Sets and cell phones generated around 9.5 Mt of waste, in 2008 the amount generated raised a 7.1% which makes a total generation of 10.2 million tonnes. It can be observed that there is a significant reduction in the amount of WEEE generated in 2009. The amount of WEEE has remained more or less stable from 2009 until 2012 and it is considered to be most likely influenced by the global financial and economic crisis. On the other hand, it can be observed that there is a steady growth of separate collection and recovery from 2007 to 2011. In 2012, 3.5 million tonnes of WEEE were collected separately and 3.6 million tonnes were treated in the EU. (EUROSTAT, 2016)

The collection target by weight of the equipment each original equipment manufacturer (OEM) sells in a year is set to a 45% in 2016 and in 2019 it is expected to increase either to a 65% of the products placed on the market that year or an 85% of the total amount of WEEE generated (European Commission, 2016).

Directive 2002/95/EC on the restriction of the use of certain hazardous substances in electrical and electronic equipment (RoHS) limits the use of noxious substances in EEE and contributes to the safety of human health and the preservation and restoration of the environment. This normative which entered into force in 2003, requires the substitution of hazardous substances such as lead, mercury, cadmium and hexavalent chromium, polybrominated biphenyls (PBB) or polybrominated diphenyl ethers (PBDE), etc., present in EEE by safer alternatives (European Union, 2003). The RoHS Directive was revised, improved and recast in 2011 although it entered into force in 2013 and member states were required to convert it into national laws by 2014 (European Union, 2011).

The motor vehicles which are no longer suitable for use reach the end of their life cycle in a process that generates between 7 to 8 Mt of waste in the EU every year. In order to provide the guidelines to manage this source of waste correctly, the EU published the Directive 2000/53/EC also known as ELV Directive (European Union, 2000). ELV Directive aims at the limitation of the production of ELV waste and the minimization of the impact that the process of dismantling and recycling vehicles has on the environment. It provides clear quantified objectives to guarantee an increase in the reuse, recycling and recovery of ELVs and their components and improve the energy conservation in the treatment process.

The European Commission published an evaluation of the ELV Directive in 2014, its results suggested that even though the implementation and compliance with the directive is costly for the industry, the environmental and economic benefits obtained after its implementation overcome the initial investment. Furthermore, the performance of automotive and ELV recycling companies is currently perceived as more efficient, sustainable and innovative than prior to the implementation of the ELV Directive.

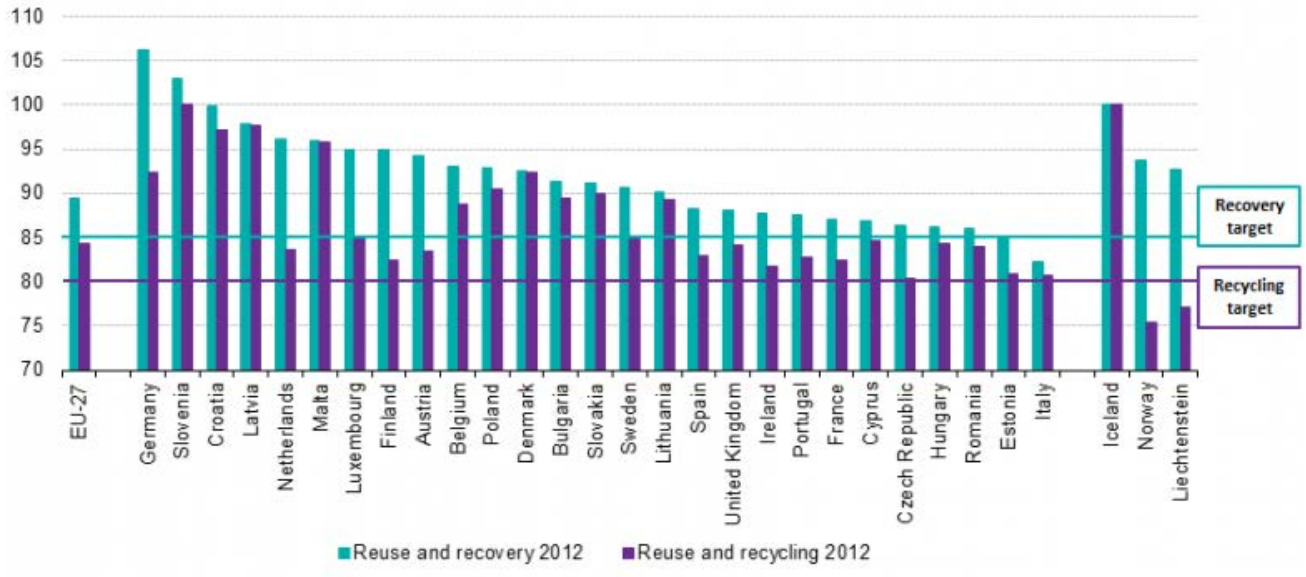


Figure 3.4. Recovery and recycling rate for end-of-life vehicles (ELV), EU-27 in 2012 (EUROSTAT, 2014)

The initial reuse and recovery was set in 2006 to a minimum of 85% by an average weight per ELV and year, and the reuse and recovery target was set to a minimum of 80% by an average weight per ELV and year. In 2015 these targets became stricter and increased to a minimum of 95% and 85% respectively. Figure 3.4 shows the recovery and recycling rates for ELV reported by EU Member States in 2012. It can be ascertain that by that year all state members but Italy, 82.3% in reuse and recovery, were in compliance with the recycling targets stablished under the framework of Directive 2000/53/EC in 2006 (EUROSTAT, 2014).

The WEEE Directive (European Union, 2012) establishes several requirements on the European Union member states such as:

- Stablish systems to encourage both, the separate collection and the return of WEEE free of charge.
- Achieve a collection rate of minimum 4 kilograms per inhabitant per year on average of WEEE from private households.

- Ensure that all WEEE collected from private households is transported to the authorised treatment sites.
- Ensure that producers stablish systems to provide with the recovery, reuse and recycling of WEEE according to the defined targets.
- Report WEEE targets on a regular basis.

In addition, Article 15 on Information for treatment facilities estates that, Member States shall ensure that the producers provide all the necessary information free of charge about the preparation for reuse, and the management of WEEE in respect of each type of new EEE placed for the first time in the Union market the latest one year after the equipment release. The provided information must include the list of all EEE components and materials, as well as the location of hazardous substances (European Union, 2012).

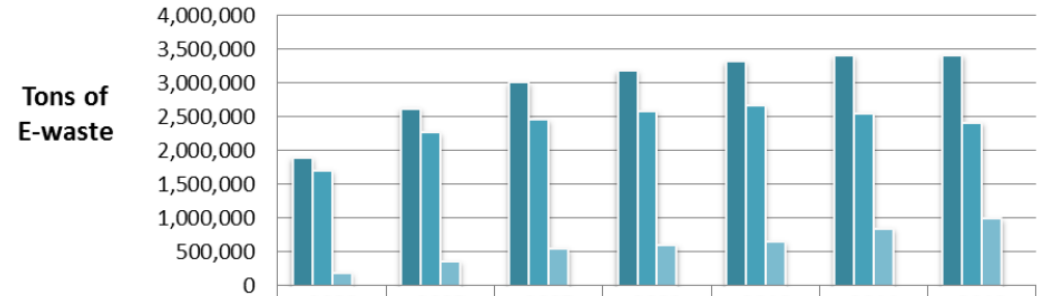
3.3.2 United States of America E-Waste Recycling Policy

There is no current U.S. Federal mandate that regulates the recycling of electronic waste, however there has been twenty five states that already have enabled new laws or improved the existing ones in order to promote and enact state-wide legislation which regulates recycling of e-waste. All those states but California (Electronic Recyclers International, Inc., 2016), prefer the "Producer Responsibility" approach, which states that the Manufacturer must assume the costs of the recycling treatment. This approach has persuade the EEE manufacturers to review and improve their e-waste return management policy across the United States. In addition, there are currently twenty states which have banned the disposal of WEEE in landfills (Electronic Recyclers International, Inc., 2016).

In 2004 California State enabled the 2003 Electronics Waste Recycling Act, also known as Senate Bills 20 and 50 (SB 20 & 50), which was amended in 2015. The main goal of SB 20 & 50 is the reduction of hazardous substances sold in California. Additionally, it requires the retailer to collect a fee Electronic Waste Recycling from customers that purchase certain EEE with cathode ray tubes (CRT), liquid crystal displays (LCD) and plasma display. Retailers are able to retain a 3 % of the collected fee and must submit the rest to the Californian State Board of Equalization (Californian State Board of Equalization, 2003).

The eCycling Leadership Initiative it is an interstate resolution done by a collaboration among consumer electronics manufacturers, retailers, collectors, recyclers, non-governmental organizations and governments at all levels, that was initiated by the Consumer Electronics Association (CEA) and

its main goals is to increase the annual amount of the EEE recycled to 0.45 Mt by 2016 while the quantity of the available collection sites is increased and the consumer awareness is improved (CEA, 2014).



	2000	2005	2007	2009	2010	2011	2012
Total e-waste generated	1,900,000	2,630,000	3,010,000	3,190,000	3,320,000	3,410,000	3,420,000
E-waste trashed	1,710,000	2,270,000	2,460,000	2,590,000	2,670,000	2,560,000	2,420,000
E-waste recycled	190,000	360,000	550,000	600,000	650,000	850,000	1,000,000
Percent Recycled	10.0%	13.7%	18.3%	18.8%	19.6%	24.9%	29.20%

Figure 3.5. WEEE generation and recycling in pounds from 2000 to 2012 in the United States (EPA, 2014)

According to a report published by the U.S. Environmental Protection Agency (EPA) in 2011, it was estimated that in 2009 an amount of 438 million of EEE were sold, 2.15 Mt were at their end-of-life and only a 25 % of them were collected for recycling (ICF International, 2011). The generation of WEEE in 2012 reached the 3.1 Mt (see Figure 3.5) of which only a 29.2 % was recycled and the rest was trashed in landfills or incinerators (United States Environmental Protection Agency (EPA), 2016). A report from the CEA states that 0.281 Mt of WEEE were recycled in the United States in 2013. This total indicates an increase of a 5.98 % compared to 2012 and a 206 % compared to 2010 (CEA, 2014).

3.3.3 Asia E-Waste Recycling Policy

- **Japan**

Japan began its efforts to improve the existing legislation on different types of waste and their recycling in 1998 by the enablement of the Home Appliance Recycling Law which entered into force in 2001. Up to that date, the increase in both, the types and the amount of WEEE, became more difficult to manage by municipalities in an environmentally efficient manner. By then WEEE was shredded to recover some recyclables and the remaining residues were disposed of in controlled

landfills due to their content of hazardous materials in shredded dust. The Home Appliance Recycling Law considers the collection of four types of household WEEE: air conditioners, refrigerators and freezers, washing machines and clothe dryers, and televisions. Consumers are required bring the e-waste to the same company in which they purchased the product and must also pay a fee to cover the end-of-life recycling and transportation costs. The retailers are in charge of the shipment of the products to the designed collection sites while manufacturers must ensure the recycling of WEEE. Japan's e-waste recycling rate is approximately 75% for products covered under the Home Appliances Recycling law, due to the fact that greater financial responsibility is placed on the consumer. (METI, 2000)

The Act on the Promotion of Effective Utilization of Resources (PEUR) manages the personal computers and other Typical WEEE since 1991, last revised in 2000. Its main purpose is to stablish a sustainable society based on the reduction, reuse and recycling. In 2006 an amendment was done to regulate the standards of judgement (Hotta, et al., 2014). This amendment states that the manufacturers must provide the composition and recycling information of EEE in order to boost the recycling and reuse of these products. The information will be provided in accordance to the Japanese Industrial Standard JIS C0950 (METI, 2015). In addition, the Small Electrical and Electronic Equipment Recycling Act manages the small electronic appliances such as mobile phones since 2013 (Hotta, et al., 2014).

- **China**

The Chinese Ministry of Information Industry (MII) published in 2006 the Management Methods for Controlling Pollution Caused by Electronic Information Products (ACPEIP) Regulation, which entered into force in 2007 and it is the equivalent to the European ROHS. China ROHS includes an ample regulatory framework for substance restrictions, pre-marked certifications, labelling and information disclosure requirements. (INGUN, 2015; Siemens PLM Software, 2010)

Figure 3.6. Example of Chinese RoHS (ACPEIP) labelling requirements (INGUN, 2015)

These requirements affect all the electronic information products (EIP) which are sold and imported in China and includes product marking requirements for the six restricted substances in the EU

RoHS for all applicable products. In order to satisfy the labelling of the Chinese RoHS requirements, all products must include labels with the presence of all restricted substances and if any toxic substance is present, it must also include the period of safe usage (see Figure 3.6). (INGUN, 2015)

3.4 Product Lifecycle Management

Presently, when a complex product is manufactured its information and the one related to its manufacturing process needs to be accessible to the different stakeholders that participate in the lifecycle of the product, e.g. partners, suppliers, customers, etc. The current tendency is to use a PLM strategy to integrate people, processes, business systems, and information in order to manage the development of the product and support its lifecycle (Lee, et al., 2008). This strategy allows the connection and communication among the different areas that manage the various stages of a product's lifecycle and ensures that the precise information is accessible to the right person at the exact moment and in the correct format (Grieves, 2006; Ameri & Dutta, 2005). As a consequence, PLM contributes to the improvement of the communication and cooperation among the diverse stakeholders involved in the lifecycle of a product (Peng & Trappey, 1998; Xu & Liu, 2003; Peltonen, et al., 1996).

Due to this, PLM has become an important concept in companies providing technologies and methodologies to manage data, information, and knowledge along the whole product life cycle. In addition, PLM is starting to be recognized as a business support for companies to become more innovative in order to meet current challenges such as product customization and traceability, growing competition, shorter product development and delivery times, globalization, tighter regulations, and legislation. Therefore, the aim of PLM is to trace and manage all the activities and flows of data and information during the product's process of development and also during the actions of maintenance and support so a new business model that integrates engineering processes and different ICT tools can be identified. (Grieves, 2006; Morris, et al., 2004; Terzi, et al., 2010; Li, et al., 2013). Some of the different business procedures are manufacturing execution systems (MES), enterprise resource planning (ERP), product data management (PDM), computer supplier management (CSM), customer relationship management (CRM), supply chain management (SPM) and environmental material aggregation and reporting system (EMARS) to name a few.

In recent years, several authors have argued about the definition of PLM using a managerial or a technological view. For example:

- PLM is an integrated, information-driven approach to all aspects of a product's life, from its design through manufacture, deployment and maintenance culminating in the product's removal from service and final disposal. PLM software suites enable accessing, updating, manipulating and reasoning about product information that is being produced in a fragmented and distributed environment. (Stackpole, 2003)
- PLM is an unified, information-driven approach constituted by people, processes/practices, and technologies to all aspects of a product's life, from its design through manufacture, deployment and maintenance that concludes in the removal of the product from service and final disposal. (Grieves, 2006)

3.4.1 Product Life Cycle Stages

The product life cycle is characterised by three phases which can be subdivided in a total of five phases that are represented in Figure 3.7 (Grieves, 2006; Terzi, et al., 2010; Kiritsis, et al., 2003; Ciceri, et al., 2009):

- **Beginning of life (BOL):** Is composed of the first stages of the existence of a product:
  - **Product Design:** is focused on the generation of ideas that can be conceptualized into a real item. This stage usually requires good communication and feedback between designers, manufacturers and final customers in order to evaluate the ideas and prototypes of a product. In addition, the development of production systems and processes are also included in this stage.
  - **Manufacturing:** is the process in which raw or previously manufactured materials are transformed into a more complex final product which is delivered to the end-user at the completion of the production process.

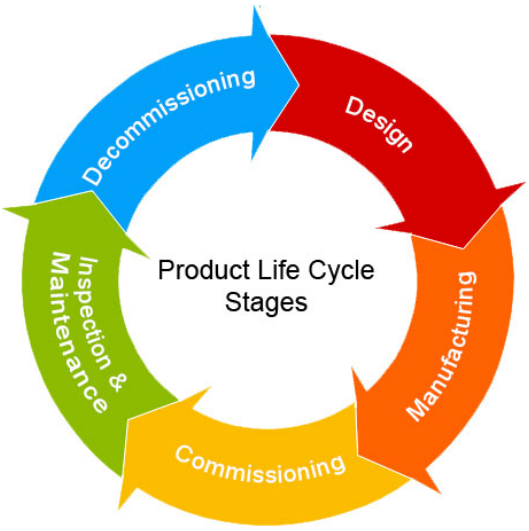


Figure 3.7. Product Life Cycle Stages

- **Middle of life (MOL):** Is composed of the stages related to the introduction and perpetuation of a product in the market:
  - **Commissioning:** in this stage of the life cycle the all the systems and components of a product are tested and verified to check if they are installed and function as required by the client. Apart from the verification and testing of the product functionality, this stage also includes the verification and correction of the documents discrepancy when required.
  - **Inspection and Maintenance:** Process in which the condition of a product is verified in order to repair or replace the damaged components to restore the product's functionality. This stage usually follows inspection and maintenance standardized procedures.
- **End of life (EOL):** Refers to the final stages of the existence of a product:
  - **Decommissioning:** In this stage the product is retired from the market at the end of its life cycle. This stage includes all the recycling stages such as collection, decontamination, disassembly, shredding, sorting, recovery, reuse and disposal.

3.4.2 Traceability

Traceability in PLM is the ability to demonstrate that the path of a product's travel through time can be followed seamlessly back to its origin (Grieves, 2006), i.e. is the set of those pre-stablished and self-sustaining procedures that provide insight into the history, location and trajectory of a product or a batch of products along their life cycle at any given time and that can be followed either to its origin or to its end-of life (see Figure 3.8).

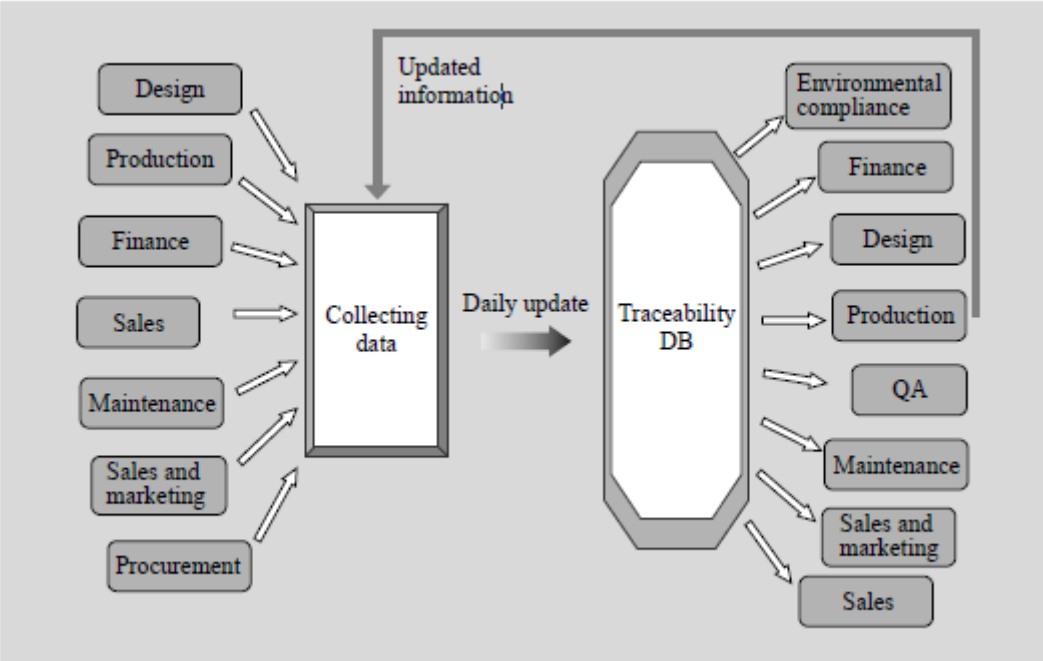


Figure 3.8. Progressed Traceability DB (Minami, et al., 2004)

There are two types of traceability:

- **Internal traceability:** Is the ability to follow the footprint left by a product along all the internal processes within a company, such as its creation, design, composition, variations, machinery, tools, temperature, batch, etc., that is, all the evidences that may affect the product during its manufacturing process.
- **External traceability:** Is to outsource the internal traceability and include additional information if necessary, as can be a packing failure, a temperature change during the manufacturing process, etc.

3.5 Augmented Reality

The first time AR was used as it is defined nowadays dates back to 1968 when Sutherland utilised a see through head mounted display (HMD) to present 3D graphics (Sutherland, 1968). However, the term “augmented reality” (AR) was not introduced until 1992, when Caudell & Mizell employed it to describe a new technology that enabled the use of a heads-up, see through display in combination with sensors to control the position of the head and workplace registration systems, in order to augment the operator’s visual field with useful and dynamic information related to the task in development (Caudell & Mizell, 1992). Since AR technology shows a 3D combination between the

real and the virtual worlds, it can also be described as the neutral ground between virtual reality (VR) and telepresence (completely real) (Milgram & Kishino, 1994; Milgram, et al., 1994).

Azuma (1997) defined AR as a variation of VR that allows the superposition of objects upon the real world instead of a total immersion in the virtual environment. The AR is not limited to specific technologies and its main characteristic is that it allows the 3D interactive combination of the real and virtual worlds in real time. This definition was widened in a follow up survey to include all senses (Azuma, et al., 2001).

Figure 3.9. Maintenance task on a Rolls Royce DART 510 engine (Gallagher, 2015)

As a consequence of the reduced time-to-markets, the shorter product life cycles and the smaller production counts, an operator encounters a continuous increase in the number of different generation systems that he has to be able to handle. AR is a technology that integrates data (images, video and text) of virtual objects into images of the real world. These images can be obtained from a camera or, if a see through HMD is employed, the user's direct view on the world can be augmented (see Figure 3.9) (Reiners, et al., 1999). A similar definition can be found in more recent sources as can be Kipper & Rampolla (2012) who state that AR uses the digital or computer generated information such as images, videos, audio, touch or haptic sensations, to superpose them upon a real-time environment.

4 Literature Review

As stated in chapter 3 the recycling process of electrical and electronic equipment and vehicles has become a topic of concern, not only from the perspective of the waste treatment but also from the recovery and reuse of materials point of view.

In order to obtain a more sustainable manufacturing process the organizations are forced to integrate systems able to deal with the enormous amount of information efficiently. The subsequent paragraphs will present the argument of how PLM and AR can support the circular economy.

4.1 PLM, Circular Economy and Green Manufacturing

As was explained in paragraph 3.2, the main purpose of the circular economy is to reduce or eliminate the use and generation of hazardous materials during all the stages of a product's lifecycle and promote green manufacturing which allows the reduction of energy and material consumption by the reuse, recovery, and treatment of product components at the product's end-of-life. Furthermore, green manufacturing not only helps the environment to recover and improves the safety of the people involved in any stage of the product lifecycle, but also has a beneficial effect in the supply chain and the overall success of a company (Barreto, et al., 2007; 2010). The implementation of sustainable manufacturing can lead to a reduction in costs and lead time, and an increment of the quality of the product (Noci, 1995; Azzone & Noci, 1998). However, green manufacturing poses a challenge for organizations, since it requires the integration of people, processes, technology, information and knowledge, along the whole life cycle of a product (Huang, et al., 2004; Sulaiman, 2000; Stark, 2011). Likewise, if a company wants to comply with the environmental legislations and implement green manufacturing, the process demands the company to adjust their entire operational processes; from product planning to logistics (Azzone & Noci, 1998). In addition, Barreto et al. (2007; 2010) mention how difficult is to attain success in the implementation of green manufacturing when the optimization of a product is accomplished by the individual improvement of its parts. They state that it is indispensable a holistic approach in which the whole lifecycle of a product is optimally design from its planning to its reuse, recycling or disposal. Kimura & Kato (2002) believe that if a sound policy for product maintenance and development during usage is implemented, and all the operations along the life cycle are completely structured, as a consequence all the revamping and commissioning activities will also be justified and manageable.

Stark (2011) indicated that PLM is a holistic approach, Grieves (2006) defined it as a "cradle to grave" concept and Stackpole (2003) described PLM as an integrated, information-driven approach which can be used to improve the life of a product. Other authors highlight that the essence of PLM is formed by the collection, allocation and management of the information and knowledge related to all the product lifecycle stages, i.e. from its design to its decommissioning. The efficiency attained with PLM helps the manufacturing companies to manage the lifecycle information of a product, which constitutes PLM as a key component in the implementation of circular economy that exchanges all the information related to a product for energy, wasted time and material across the entire organization and in the supply chain. (Grieves, 2006; Morris, et al., 2004; Terzi, et al., 2010; Li, et al., 2013)

Since a PLM system can contain and administer information related to all the stages of a product's lifecycle; it not only supports the assessment of a particular product or any of its versions to control the compliance with the environmental laws currently in force, but also allows the identification of the different processes that may have an adverse environmental repercussion (Minami, et al., 2004), the traceability of the product's data back and forth along its lifecycle, and the aggregation of manufacturing experience (Li, et al., 2013). Furthermore, the traceability of products in PLM systems can contribute to the improvement of the application environment, and the increment of outputs and quality of product development (Ameri & Dutta, 2005; Shiau & Li, 2007). Moreover, Sung & Park (2007) indicate that, in terms of productivity and competitiveness, PLM systems can provide with several advantages such as enhanced design and manufacture accuracy and improved project management among others. Additionally, Hewlett-Packard Co. (1994) affirms that PLM can lessen the production costs, diminish the time-to-market and enhance the quality of the process.

In order to achieve adequate results an efficient PLM system shall count with five core functionalities (Sung & Park, 2007; Liu & Xu, 2001; Xu & Liu, 2003; Li, et al., 2013). The first functionality is data vault and document management which provides services for the storage and retrieval of product information, as can be secure data storage, access control, metadata management or product data classification, linkage and relationship, etc. The second is engineering change management which controls the associated procedures for the administration of product data and provides a mechanism to operate the business with information. For example: workflow administration, routing control and messaging and notification preserve, task or product electronic sign-off check, etc. The third, BOM or product structure management, manages the bill of material, product configuration, associated versions and design variations, for instance product structure

creation, maintenance, search and access authorizations. The fourth functionality, parts management, provides information on standard components and facilitates the re-use of designs, e.g. parts creation, maintenance, classification, search and access authorizations. And the fifth functionality is system management which provides some utility tasks such as user authorization management, system security control, data communication, database management and project tracking. The authors defend that a sustainable product lifecycle management can be implemented by the preparation of all the activities related to the lifecycle of a product. Furthermore, the social and economic repercussions can be regulated as a consequence of the control of the resource utilization achieved by the definition of the necessary sustainable checklists.

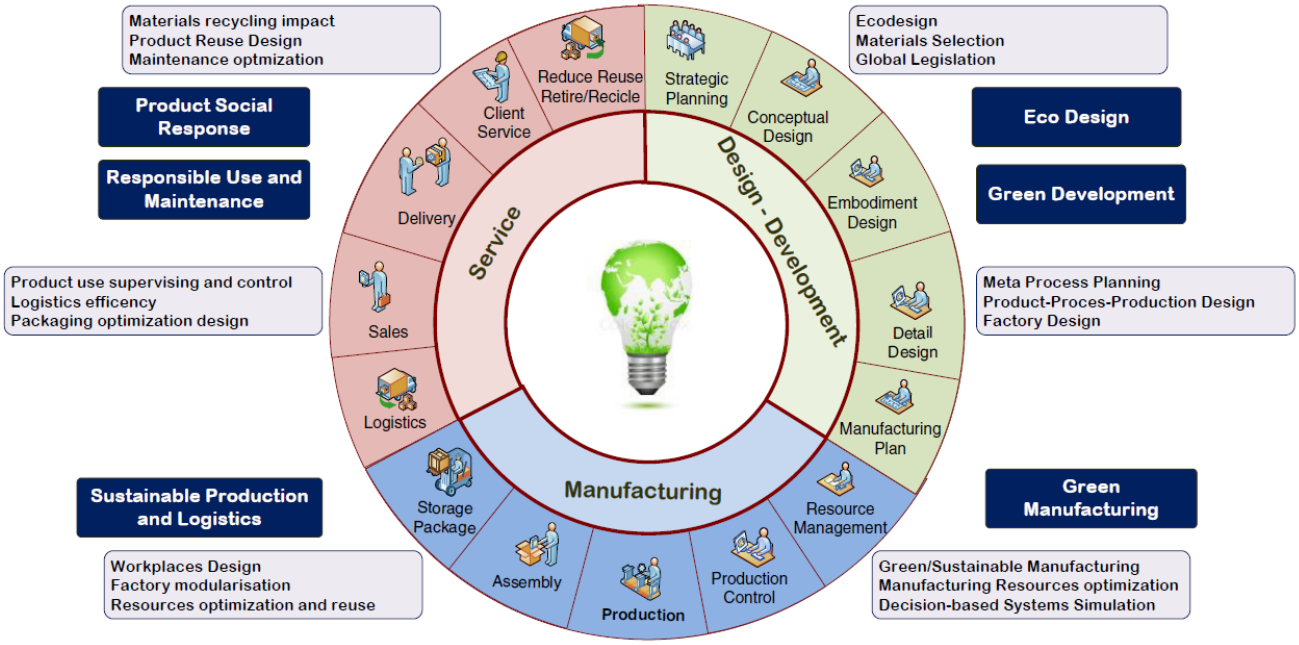


Figure 4.1. Green product lifecycle framework (Vila, et al., 2015)

Vila et al. (2015) states that the necessary competences for the green manufacturing enterprise related to the product's life cycle are: eco design (ED) which promotes a product design in which all the environmental repercussions will be considered and evaluated, from both the recycling point of view and from the perspective of any of the product lifecycle stages; in green development (GD) the manufacturing plans can be developed in order to minimise the environmental consequences of the use of manufacturing technologies; green (sustainable) manufacturing (GM) aims at the creation of manufacturing processes that minimise environmental impact and conserve energy and nature resources at the same time that ensure the safety of employees and population; sustainable production and logistics (SPL) focusses on the contribution that mass production and outsourcing bring to sustainable PLM; responsible use and maintenance (RUM) outlines intelligent control which

helps to supervise the product's use and maintenance so there is a feedback to ED activities; and product social response (PSR) in which products are recycled, reduced, reused and retired in order to close the lifecycle.

Barreto et al. (2007; 2010), Ciceri et al. (2009), and Vila et al. (2015) define sustainability in the context of PLM as the optimization of all the product life cycle stages and present some of the relations between PLM and green manufacturing. The combination of the five core functionalities with the necessary competences for a green manufacturing enterprise create a sustainable PLM framework which can be used to manage the information throughout the five stages (see paragraph 3.4.1) of the product lifecycle and can be represented as shown in Figure 4.1. For example, in the design stage the product specifications and requirements are defined, stored and made accessible for all other stages. In addition, the information core could become and archive for all environmental laws and issues. During the manufacturing stage PLM can provide with the information related to existing equipment, manufacturing procedures and facility lay-outs, this data can be used to evaluate and select the facilities that are more appropriate for the implementation of green manufacturing. Throughout commissioning and inspection and maintenance stages PLM can administer information related to warranties, product performance, and other information that would make the maintenance stage more competent. In the commissioning stage, PLM can provide with information regarding the product design and material composition, together with methods of disposal and recycling treatments. (Barreto, et al., 2007; 2010; Ciceri, et al., 2009; Vila, et al., 2015)

4.2 AR Application to the Product Life Cycle in Manufactory

Augmented Reality systems have experienced a significant improvement since their first appearance when Ivan Sutherland (1968) made the first prototype for head mounted display nick-named Sword of Damocles due to its size and weight. AR technology has progressed rapidly ever since, the following paragraphs provide a review of the different AR applications and their relation to the different stages of the product's life cycle such as assembly, maintenance, revamping and decommissioning.

4.2.1 AR Application for Assembly Instruction and Training

The utilization of AR for industrial assembly support has been a pursued area for a long time. One of the first studies was done by Claudell and Mizell (1992) who demonstrated that AR based systems could guide Boeing operators in the assembly of airplanes. The study developed a system in which a

heads-up see through display was used to guide the technician through the assembly process of wire bundles that were a part of the electrical system of an aircraft. The instructions were given through rendered wireframe diagrams or text instructions situated in front of the worker and close to the assembly part. A video camera mounted on the AR display recognized markings located on the work piece and sent the information to the processing system which calculated the position of the operator and returned to the display the information previously assigned to the markings (Caudell & Mizell, 1992; Sims, 1994). The proposed system proved to aid the operators to develop their tasks faster. When the operator's performance with the AR system was tested and compared to the conventional approach, the results showed an improved performance in the assembly of wire bundles (Curtis, et al., 1998). A similar study was developed in order to use AR to provide support information for the spaceframe assembly in the construction area. In this case, the tracking was executed by a passive inertial system which was connected to an active optical system that used blinking LEDs (Webster, et al., 1996). An AR prototype system for the assembly of door locks on cars was developed by Reiners et al. (1999). The demonstrator was a functional application which presented the information on a see through display that was used to provide the users with the necessary information to correctly assemble the door lock into a car. Haniff et al. (1999) and Boud et al. (1999) investigated whether the use of an AR interface to allow the direct manipulation of different pump components could expedite the learning of the assembly sequence faster than with traditional methods. The study found that the performance times can be improved since the AR technology allows the interaction with real objects. Similar results were found by Baird & Barfield (1999), when they evaluated the effectiveness of AR compared to traditional methods in the assembly process of a computer mother board. The results of the study showed that there was an improvement in the performance when the participants used AR technology: the assembly times were reduced and the number of errors decreased. Likewise, Tang et al. (2003) performed a comparative study of the effectiveness of the use of AR instructions in object assembly against more traditional methods. The results of the study proved that AR systems improve the performance and minimise an 82% the error rate due to mental workload on assembly tasks. These results appoint AR as a more competent didactic method than the conventional ones. In similar studies, Zauner et al. (2003) and Wiedenmaier et al. (2003) obtained results which proved that AR was more appropriate than a paper manual but needed more time for the assembly that the tutored task. Robertson et al. (2008) studied the effects that four different AR configurations had when a graphical context, which was always visible in the field of view, was utilised for the assembly of Lego blocks. The obtained results indicated that the most efficient configuration was the fully registered AR, i.e. a heads-up see through display which provides context aware information. In addition, the study also showed that when the information was placed next to

the target location and lined up or horizontally with it, the participants' performance improved in terms of assembly time and committed errors. In a similar study for Lego assembly, Hou et al. (2013) implemented an animated AR system to test activities that were usually guided by paper based manuals. Their experiment results yielded shorter assembly completion times, less assembly errors and lower task load. Furthermore, it was also shown a reduction on the learning curve together with an increase in the performance of the working memory of the subjects. Syberfeldt et al. (2014) studied the potential of the use of AR for the improvement of the operator's capability in the industrial shop-floor. The results of the experiments showed a significant difference in the time needed for the successful development of the task which was dependent on the ability of the user to adapt to the use of the AR system. In a follow up study Syberfeldt et al. (2015) investigated the acceptance of the user when AR was used for the development of the same experiment. The obtained results concluded that the acceptance of the use of AR increased with the difficulty of the task, and when the virtues of the system were emphasized from the user's perspective.

4.2.2 AR Application to Inspection and Maintenance

Many AR solutions have targeted maintenance tasks in the manufacturing industry. One of the first AR applications in this field was described by Feiner et al. (1993) who developed a knowledge based AR for maintenance assistance (KARMA) system. The system used a head mounted see through display to guide the user through the maintenance and repair of a printer. In their approach, Klinker et al. (1997) proposed the display of the identification and function of each of the parts of an engine. In their study, the authors considered the display of visual augmentation to train workers the maintenance task and emphasize the need for object interaction. Neuman & Majoros (1998) realised a study on the relation between AR and its interaction with human abilities in order to benefit the maintenance tasks. The proposed system provided support to the maintenance operator during the testing of the circuitry of an aircraft with promising results. Goose et al. (2004) presented a personal digital assistant (PDA)-based AR integrating speech (PARIS) application for the inspection of a water distribution system. The results proved that the combination of a simple interaction and the integration of the AR architecture supplied constant real time information that was beneficial for the user. Regembrecht et al. (2005) suggested a new approach that developed a new system that employed AR as a user interface (UI) which could be used to guide an astronaut through the maintenance process of an air filter inside the international space station. Due to the strict specifications required for the test in space, this experiment was earth-located and obtained positive results. Henderson & Feiner (1999) developed a system that used a head mounted display that

interacted with an AR system through the use of a wrist worn controller. The proposed solution showed its usefulness within a cramped environment, it limited the movement of the head and provided a faster repair time.

4.2.3 Augmented Revamping and Decommissioning

Once a product reaches the end of its life cycle, it must be decommissioned in a process prepared in advance to limit the exposition to hazardous materials and guarantee the safety and health of the personnel involved in the recycling process (European Union, 2012).

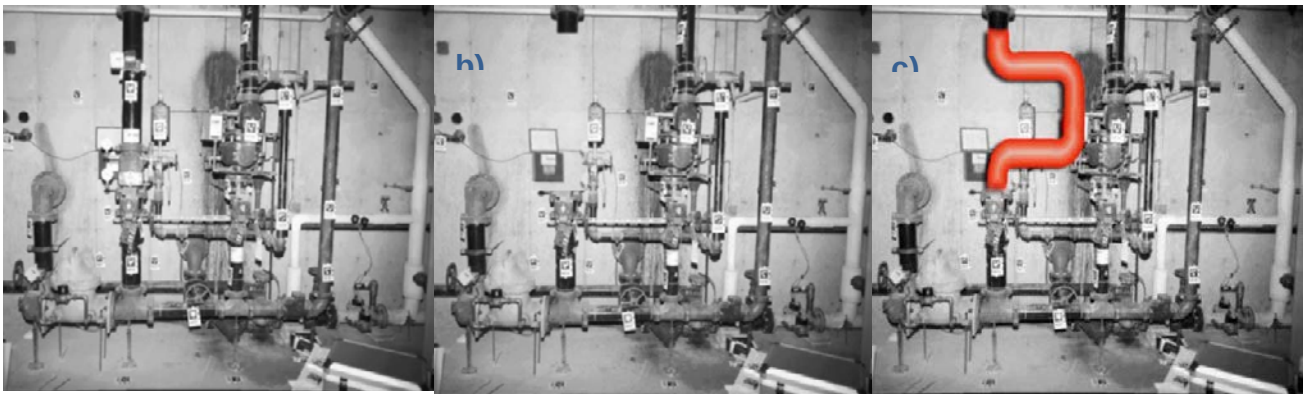


Figure 4.2. Diminished reality in a revamp procedure. a) Original, b) Diminished, c) Augmented (Zokai, et al., 2003)

Zokai et al. (2003) study the possible application of AR to revamp or decommissioning procedures in industrial plants. The proposed method employs diminished reality techniques that allow maintenance planners the removal of objects from a scene, e.g. valves, pipelines, pumps, etc. In order to do this, they access a DB which contains the current CAD images and the real object is then eliminated from the scene and/or replaced by a new CAD object (see Figure 4.2). This process allows the inspection of the process in order to asses if there could be any collide with any object not present in the CAD data. In addition, Ishii et al. (2008) proposed an AR supported method to aid the decommissioning of the ion tower of a nuclear power plant. The security and safety requirements in this kind of procedure demand the process to be planned and its feasibility verified before the real dismantling occurs. Moreover, the decommissioning procedure and the support CAD files have to be constantly updated to reflect the current state of the plant. The authors claim that the system proposed guarantees that the cutting lines are situated in place, monitors the procedure, and also documents the progress of the process. The evaluation of the feasibility of the study done by Shimoda (2007), proved that this kind of implementation provides a more comprehensive

information and is easier to use than paper documentation. However it would be needed a bigger and lighter display if the method is to be implemented in real decommissioning situations.

4.3 Augmented Reality, Product Lifecycle Management and Circular Economy.

In order to stay competitive in an ever changing market, companies these days must not only be able to comply with the continuous fluctuations of the market, but they should also be able to adapt faster to those changes than their competitors. Furthermore, producers and manufactures are compelled to manage and calibrate their supply chains to avoid unnecessary delays and failures. In addition, companies need to take advantage of their digital information assets, which involve a tremendous amount of data related to the entire life cycle of all their products.

As stated in paragraphs 3.3.1 and 3.4.2, it is important the development of an accurate traceability system that facilitates all factions the access to information about a product such as specifications, included substances, FEM analysis, simulation results, bill of materials (BOM), bill of process (BOP), etc. Likewise, information regarding the disposal and recycling process of every EEE and ELV must be available for the recycling companies at all times. Additionally, a report about the sustainability of a product or a manufacturing supply chain must be provided instantly when requested by a government agency or a non-profit organization. Furthermore, WEEE, RoSH and ELV Directives state that manufacturers must ensure that the number of a product corresponds to the serial number, so when a customer disposes of EEE or ELV and the recycling company dismantles it, all the information about which components must be collected, and whether they are to be recycled or reused must be available either through a web site or on a CD. Moreover, when a product does not comply with the current regulations, detailed information concerning its components and treatment must be provided through the same sources. Finally, once the recycling process of the technological product has been done, the end-of-life of the serial number needs to be registered. (European Union, 2000; 2011; 2012)

When AR technology is intended to be used in this context, one of the main challenges is the reuse of the existing product information which should be retrieved, as automatically as possible, from the PLM system in a standardized format compatible for the AR display. Heilala et al. (2008) reviewed some of the initial standardized formats and proposed to take advantage of their interoperability to enable the exchange of information in the manufacturing industry. Sääski et al. (2008) designed a methodology and a system for implementing an AR-PLM interface to aid the assembly work in

industry. The approach superposed 3D CAD designs over the real environment. In order to achieve a good communication protocol, the authors adopted the definitions given by the standard ISO 10303-203:1994 on product data representation and exchange (also known as STEP). The preliminary test of the system was done with a 3D puzzle and obtained promising results. In a follow up study by Salonen et al. (2009), the main focus was to test the assembly on a real industry case, the accessory power unit of a tractor, with positive results. In a posterior study, Fiorentino et al. (2010) propose to link PLM to AR to improve current industrial CAD drawing with the use of tangible digital master (TaDiMa). The purpose of the study is not to only add information to the CAD model but also to interact with it on a tangible interface in which the users can navigate and make notations to the CAD model. The conclusions of the study proved that a marker based tracking permits a simple integration of AR in the industrial life cycle process. The obtained results revealed a significant reduction of paper material for drawings and a significant increase on the speed of the revision phases. However, the marking base tracking system was found to be unstable for manufacturing environments, and the tested display had resolution problems and a limited field of view.

4.4 Review Summary

According to the present literature review, AR has been proven to be an useful user interface for different application domains such as assembly, maintenance, revamping and decommissioning. In addition the integration between AR and PLM has shown promising results (Heilala, et al., 2008; Sääski, et al., 2008; Salonen, et al., 2009; Fiorentino, et al., 2010). One main reason for the use of AR is the ability to locate the information in situ, which allows a hands-free interaction with the product (Robertson, et al., 2008). The information can be overlaid and registered on the workspace in a meaningful way, which makes AR a competent instructional method (Tang, et al., 2003). It has been proven that AR assisted assembly is and effective method to assist assembly tasks (Tang, et al., 2003; Wiedenmaier, et al., 2003; Baird & Barfield, 1999). Some experiments were conducted in order to compare the performance of an AR based system with more traditional methods when utilised to aid a user through an assembly or maintenance task. The results of these experiments proved that the AR solutions were more effective than the conventional methods and improved the performance of the operators (Caudell & Mizell, 1992; Sims, 1994; Curtis, et al., 1998; Regenbrecht, et al., 2005). In addition, the operators had faster assembly times and less errors (Baird & Barfield, 1999; Robertson, et al., 2008), the work load was reduced (Tang, et al., 2003; Wiedenmaier, et al., 2003). Furthermore, users that employed AR solutions to train an assembly task had a shorter learning curve (Haniff, et al., 1999; Boud, et al., 1999) and an increase in their performance memory

(Hou, et al., 2013). Furthermore, AR solutions have shown promising results when applied to revamping and decommissioning (Zokai, et al., 2003; Ishii, et al., 2008) and have been proven to provide a more comprehensive information and to be easier to use for these tasks than paper-based instructions (Shimoda, et al., 2007).

The main limitations of AR solutions for assembly, maintenance, revamping or decommission are to determine when, what and where to display the information on the AR display. There are limitations in the current calibration, display resolution, tracking techniques and registration algorithms (Curtis, et al., 1998; Yuan, et al., 2008). The phenomenon known as attention tunnelling could be a cause for the reduction of the user's performance in cases where the amount of information provided through the AR display overpowers the user's attention and causes distraction from the real environment (Tang, et al., 2003).

According to the aforementioned reasons, the circular economy and the technological recycling industry can strongly benefit from the implementation of an interface between AR and PLM for the purpose of recycling EEE and ELV.

This project will present an AR-PLM system architecture for the assistance of the recycling work by visual information superimposed on the physical assembly parts that must be recycled. AR methods are specifically well suited for complex, short manufacturing series or in a customized production factory environment since each EEE and ELV within the same class may have a slightly different configuration. The traditional approach is to use blueprints and instruction manuals with guiding pictures which describe the content of each operation. As technological products become smaller, the need for guidance to the worker with all available tools becomes increasingly important. The AR-PLM interface can reduce disassembly times, accelerate learning of the recycling task and provide with more quality assurance to the recycling process.

Table 4.1 to Table 4.7 show a summary of the papers and articles that have been used to develop the literature review. The sources have been divided into two groups, AR and PLM.

Table 4.1. List of AR papers and their contents. Part I

Publication	Research Scope	PLM	Method	Limitations	Conclussions
Baird & Barfield (1999)	AR effectiveness for assembly	No	Computer mother board assembly	↓ Resolution and poor image projection on see-through displays	Improved performance with AR technology: the assembly times were reduced and the number of errors decreased.
Boud et al. (1999)	VR and AR as training tools for assembly	No	Water pump assembly	Lack of haptick feedback in VR. Problem with registration between real and virtual objects.	The performance times can be improved since the AR technology allows the interaction with real objects
Caudell & Mizell (1992)	AR for Aircraft assembly support	No	Wiring assembly & maintenance	↓ Resolution, calibration and tracking	Increased efficiency
Feiner (1993)	AR for maintenance and repair	No	Laser printer maintenance	Poor registration and manual focus and calibration. No visible line support.	AR can be used for maintenance tasks although as its development is at an early stage, the system needs some improvements.
Fiorentino et al. (2010)	Tangible AR interface implementation	Yes	Augmented user,	Display resolution, limited field of view, bad tracking	A marker based tracking permits a simple integration of AR in the industrial life cycle process. Significant
Goose et al. (2004)	VRML-based AR framework for industrial service and maintenance	No	Development of PARIS system	Problems with marker tracking in some areas, mobile collaborative fault diagnosis	The combination of a simple interaction and the integration of the AR architecture supplies constant real time information that is beneficial for the user.
Heilala et al. (2008)	Review of smart manufacturing	Yes	Interoperability as infraestructure to share information	N/A	Proposes to take advantage of their interoperability to enable the exchange of information in the manufacturing industry.
Henderson & Feiner (1999)	AR for inspection and maintenance	No	Interaction of AR system with wrist worn controller	The system limited the movement of the head.	The system proved its usefulness within a cramped environment and provided a faster repair time

Table 4.2. List of AR papers and their contents. Part II

Publication	Research Scope	PLM	Method	Limitations	Conclussions
Hou et al. (2013)	AR to cognetively facilitate the assembly process	No	AR animation technology	Speculative analysis	The most efficient configuration is the fully registered AR. When the information is placed next to the target location and lined up or horizontally with it, the performance improved in terms of assembly time and committed errors.
Ishii et al. (2008)	AR supported method to aid the decommissioning of the ion tower of a nuclear power plant.	No	Reference support to cut lines and restraint parts and reccord support for the progress of dismantling	A larger display with lighter weight is required for the implementation in real decommissioning	the system proposed guarantees that the cutting lines are situated in place, monitors the procedure, and also documents the progress of the process.
Neuman & Majoros (1998)	AR for maintenance tasks	No	Creation of a mock up scenario based on AR media	Incomplete data, application not tested with real users.	The experiment was conducted with possitive results, but needs improvements.
Regembrecht et al. (2005)	AR for maintenance of the ISS	No	Analysis of 10 AR projects applied in the automotive and aerospace industry	The experiment was earth-located, tracking problems, HMD too heavy, incomplete wireless coverage.	The experiment was conducted with possitive results
Reiners et al. (1999)	AR for assembly of door locks	No	Development of a new tracking algorithm	Tracking delay	The experiment was conducted with possitive results, although needs to be used by experienced users.
Robertson et al. (2008)	Effect of AR when graphical context is provided for assembly tasks	No	Comparison study among different AR setups	Context provided outside of the working area.	Registered AR outperforms both non-registered AR and graphics displayed on a HMD

Table 4.3. List of AR papers and their contents. Part III

Publication	Research Scope	PLM	Method	Limitations	Conclussions
Salonen et al. (2009)	Animated AR for assembly task	No	Comparison of AR against traditional methods	The order of parts to be assembled, assembly tools and guides need to be specified.	Shorter assembly completion times, less assembly errors and lower task load. In addition to a reduction on the learning curve together with an increase in the performance of the working memory of the subjects.
Shimoda (2007)	Feasibility of AR for decommissioning of the ion tower of a nuclear power plant.	No	Comparison of AR against traditional methods	Display too small in size and heavy to carry.	Implementation provides a more comprehensive information and is easier to use than paper documentation.
Syberfeldt et al. (2014)	Potential of AR for the improvement of the operator's assembly capabilities	No	Comparison of AR against traditional methods	No real see-through display. Problems with tracking system and poor resolution.	Improved performance with AR technology: the assembly times were reduced
Syberfeldt et al. (2015)	User aceptance of AR for assembly tasks	No	Comparison of AR against traditional methods	No real see-through display. Problems with tracking system and poor resolution.	The acceptance of the use of AR increased with the difficulty of the task, and when the virtues of the system were emphasized from the user's perspective.
Sääski et al. (2008)	Metodology for implementing an AR-PLM interface for assembly work	Yes	Usage of ISO 10303 (STEP)	Lack of accuracy, and incomplete system integration.	Smooth information retrieval from PDM is achieved.
Tang et al. (2003)	Effectiveness of AR for object assembly	No	Comparison of AR against traditional methods	The attention tunnelling phenomenon could be a cause for the reduction of the user's performance.	AR is a competent instructional method. AR is and effective method to assist assembly tasks. Error reduction of 82%.
Wiedenmaier et al. (2003)	AR for assembly processes	No	Comparison of AR against traditional methods	The system does not support intuitive assembly. Problems with tracking.	AR assisted assembly is and effective method to assist assembly tasks that reduces the workload

Table 4.4. List of AR papers and their contents. Part IV

Publication	Research Scope	PLM	Method	Limitations	Conclussions
Zauner et al. (2003)	MR for assembly of hierarchical structures	No	Comparison of AR against traditional methods	At least one hand ocupied with the AR display. Detection system limited to flat surfaces	AR is more effective than a paper manual but needs more time for the assembly that the tutored task.
Zokai et al. (2003)	AR to revamp or decommissioning procedures in industrial plants.	No	Use of diminished reality techniques	Posibility of error propagation. Performance problems with non-planar surfaces.	The algorithm uses the paraperspective projection model and has flexibility to recover a crude or fine detailed background. It is applied successfully to images from industrial scenes.

Table 4.5. List of PLM papers and their contents. Part I

Publication	Research Scope	Method	Limitations	Conclusions
Ameri & Dutta (2005)	The role of PLM in supporting knowledge-intensive processes throughout product lifecycle	Review	Research issues in PLM: Knowledge representation, capture, generation and dissemination.	PLM is a knowledge management system which supports the entire product value chain. It closes different knowledge loops throughout product lifecycle by enabling reuse of lifecycle knowledge in the development phase.
Barreto et al. (2007)	Green Manufacturing, adoption of environmental standards and influences to PLM	Study of six companies	Data from published material only & limited group of study	PLM can support green manufacturing procedures by tightly integrating such practices into the lifecycle of a product and yielding environmentally friendly productions.
Barreto et al. (2010)	PLM in support of Green Manufacturing	Study of six companies	Data from published material only & limited group of study	PLM can support green manufacturing procedures by tightly integrating such practices into the lifecycle of a product and yielding environmentally friendly productions.
Barreto et al. (2010)	Why companies practice green manufacturing, what practices are engaged, and how PLM supports environmental-friendly initiatives.	Study of six companies	Data from published material only	PLM can support green manufacturing procedures by tightly integrating such practices into the lifecycle of a product and yielding environmentally friendly productions.
Ciceri et al. (2009)	PLM as an important foundation for achieving a sustainable paradigm for life, development, engineering, manufacturing, use and disposal of products.	Review	Closing product lifecycle information loops requires involvement of product users. PLM reference approach to sustainability is still lacking	Sustainability can be achieved through optimization of the use of resources along the product lifecycle, while retaining quality of products and services. PLM represents a very important approach for achieving a more sustainable paradigm of work and life, a more sustainable product development, manufacturing, use and dismissa
Glavič & Lukman (2007)	Sustainability terms and definitions. Hierarchical classification.	Survey	Enphasis given to the environmental engineering field	Sustainable systems introduce interconnections between environmental protection, economic performance and societal welfare, guided by a political will, and ethical and ecological imperatives.
Hewlett-Packard (1994)	Manage the product's lifecycle electronic information	DMS, CAD/CAM, ISO 9000, STEP, CALS	Slow process of implementation in HP	Obtention of the right information at the right time and in the right form all the way through the product lifecycle. Lower production costs, reduced time to market, and improved product and process quality. Building and maintaining project sponsorship and managing organizational change
Huang et al. (2004)	Web-based PDM framework	PDM +Java + J2EE Tomcat middleware, CAD/CAM	No CAD or ERP integration	J2EE three-tier web-based PDM provides flexibility, security and allows WWW communication. Developed solely with Java so it is independent of OS

Table 4.6. List of PLM papers and their contents. Part II

Publication	Research Scope	Method	Limitations	Conclusions
Kimura & Kato (2002)	Method for PLM support by comprehensive product life cycle simulation	Proposal of model for product usage	Simulated for short-life technologically immature products	There resuslts demonstrate the possibility for better life cycle scenarios with a certain unusual life cycle management method. The results can be generalized for several types of durable consumer goods.
Kiritsis et al. (2003)	PLM data structure for recording maintainability data for use in design for maintainability	PEID, RFID, ONS, ID@URI notation, WWAI, DFX, EPC code, PML based on XML	Focused on MOL & EOL. System definition attributes depend on the design and the main knowledge for maintenance personel	This system allows all actors to track, manage and control product information at any phase of its lifecycle, at any time and any place in the world. The proposed approach can also be used in design for reliability or design for safety and extended to the EOL phase of a product's life cycle.
Lee et al. (2008)	Evolution of PLM. Review of its characteristics and benefits.	Case of study: aviation maintenance, repair and overhaul	Study limited to two companies.	PLM shows impresive results in aircraft design although its full potential in aviation MRO activities has not been reached. PLM provides realiable and accurate feedback among partners in a extended enterprise and also among the diferent stages of a product's life cycle.
Li et al. (2013)	Audit principles and rules for PLM/PDM implementation	Method analysis, experts interview	Interviews to experts from one manufacturing sector only. Validated in one company only, cannot be generalised	General auditing points and rules for PLM/PDM systems are defined. Their implementation prevents losses due to wrong data, disordered structures, inconsistent data and improper management of product data.
Liu & Xu (2001)	Web-based PDM	Review of web-based PDM systems	N/A	Web-based PDM becomes more adaptative and flexible
Minami et al. (2004)	Eco PLM for traceability of EEE	Implementation in Hitachi	Hazardous substances not contemplated	Immediate solution for environmental laws compliance.
Shiau & Li (2007)	PDM implementation to conceptual design phase. Closed-loop PDM	PDM, DCR & DCN Workflow system	Upload of information & approval workflow procedure manually in PDM system	The application of PDM to the conceptual design phase enables early involvement of the strategy in concurrent engineering during new product design
Stackpole (2003)	Quality-based performances on small firms	State of the art review	N/A	DCF approach can be implemente successfully in firm-related environments when precision of investment is the main objective.
Sulaiman (2000)	PDM key factor to quick and efficient product development and manufacturing	Review	Lack of coordination and product documentation, and delay in data sincronization	More systematic processes, reduces costs and time-to market, and allows the identification of the different prats, components, and documents. Enables a smooth manufacturing process.

Table 4.7. List of PLM papers and their contents. Part III

Publication	Research Scope	Method	Limitations	Conclusions
Sung & Park (2007)	Design of component based PDM system for global manufacturing support	Use of J2EE architecture, JSP servlets and EJB components	The system does not cover all business areas.	The system is able to handle heavy load with good performance and high stability. After implementation in real company its productivity and competitiveness increased, reduced design hours, eliminated design inconsistency, reduced time to market and parts in inventory.
Terzi et al. (2010)	PLM state of the art	Review	Focused on BOL & MOL phases	Effective PLM can contribute to the achivement of sustainability. PLM is expected to support value creation through enablers in technical, environmental, social and economical areas.
Vila et al. (2015)	Green PLM	Framework definition	N/A	The process provides the necessary tools to achieve Green PLM and increases the efficiency in product development and sustainable lifecycle management
Xu & Liu (2003)	Web-enabled PDM in collaborative desing environment system definition	CORBA, CGI, Active X, VRML, STEP and OODBMS	Partial implementation of the system. The use of Visual Basic forces the storage of run time libraries on the client tier	The proposed system enables various collaborative designs activities. It offers increased accesibility and user friendlyness.

5 WEEE Disassembly and Recycling Process

This section includes a general definition of the recycling process for WEEE and ELV, and includes several examples of real cases studied in previous researches.

5.1 WEEE and ELV Recycling Process

A recycling process is a sequence of actions that are directed towards the recovery of reusable components and recyclable materials. The process is usually composed of three main stages: collection, processing and purchasing recycling products. As can be seen in the general WEEE recycling process schema represented in Figure 5.1, the three main stages of the recycling process form a continuous loop. Each of the stages can be comprised of one or more phases which can be classified according to their function in the recycling process.

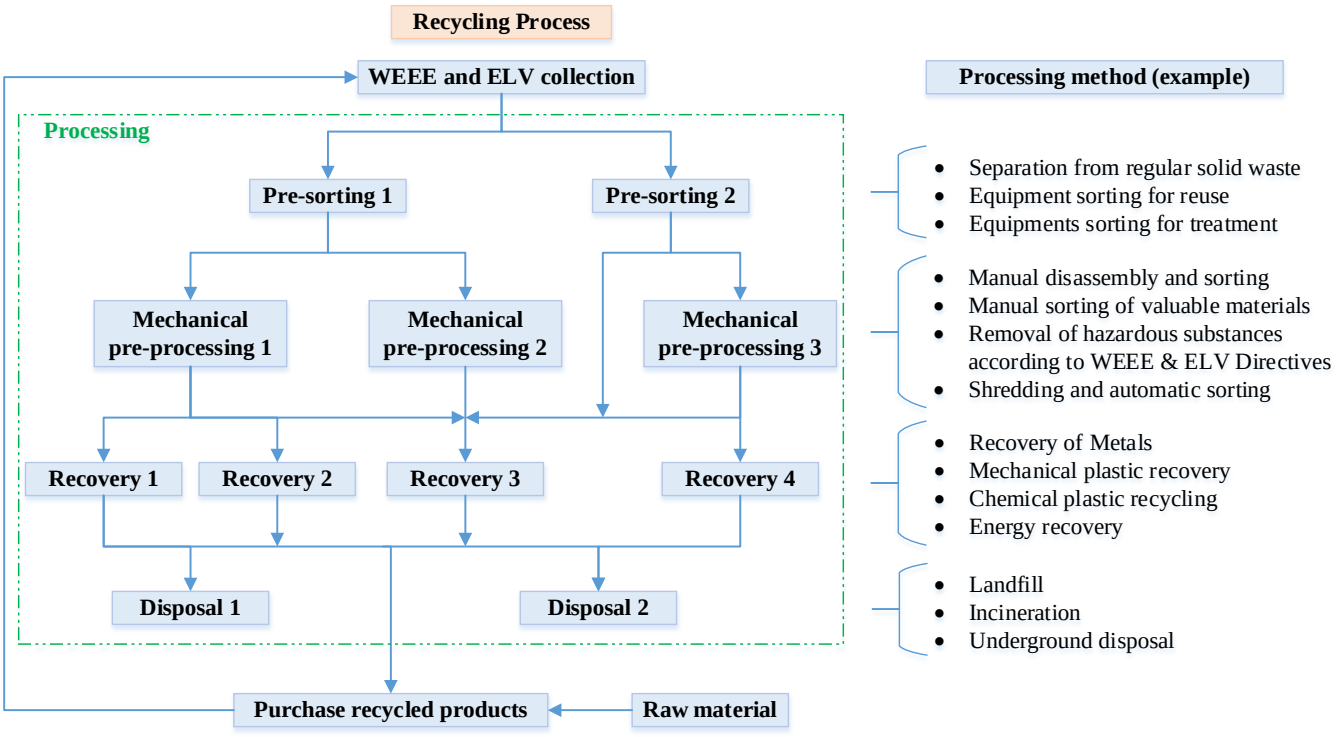


Figure 5.1. WEEE general recycling process diagram (Chancerel & Rotter, 2009; United States Environmental Protection Agency (EPA), 2016)

Although in Figure 5.1, the collection stage is form by just one process, WEEE Directive contemplates two main WEEE collection streams, municipal solid waste and separate collection (European Union, 2012), while ELV Directive considers only the collection of vehicles and parts through authorised treatment facilities (ATF) which ensure the vehicles are decontaminated before

they are treated. Once the ELV and WEEE is has been received and decontaminated in the ATF it must be treated in a process which varies on the ATF and the product; some of the possible stages in the treatment process are listed in Figure 5.1. The last step completes the recycling loop, and it is formed by the reintroduction of the recycled products into the market.

5.2 Example of Treatment Processes for WEEE and ELV

The disassembly process for WEEE and ELV may vary depending on the ATF and the company. The next paragraphs present several examples obtained from previous cases of study.

5.2.1 Waste Mobile Phone Treatment Process

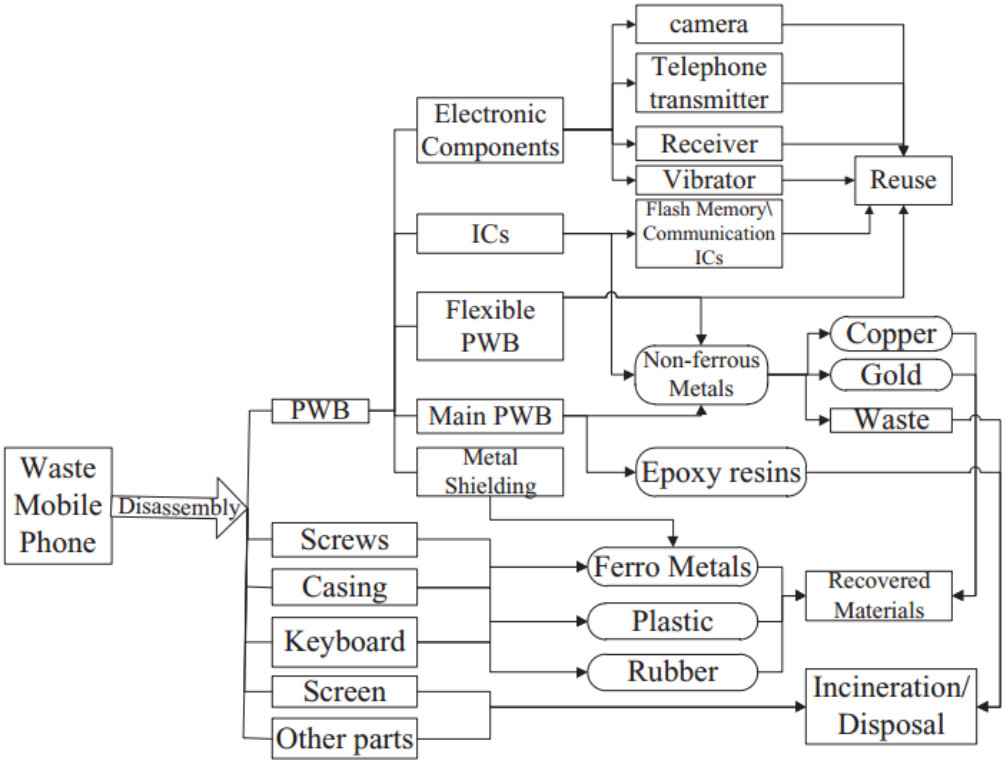


Figure 5.2. Recovery and remanufacturing of waste mobile phone WEEE (Lu, et al., 2014)

5.2.2 ELV Treatment Process

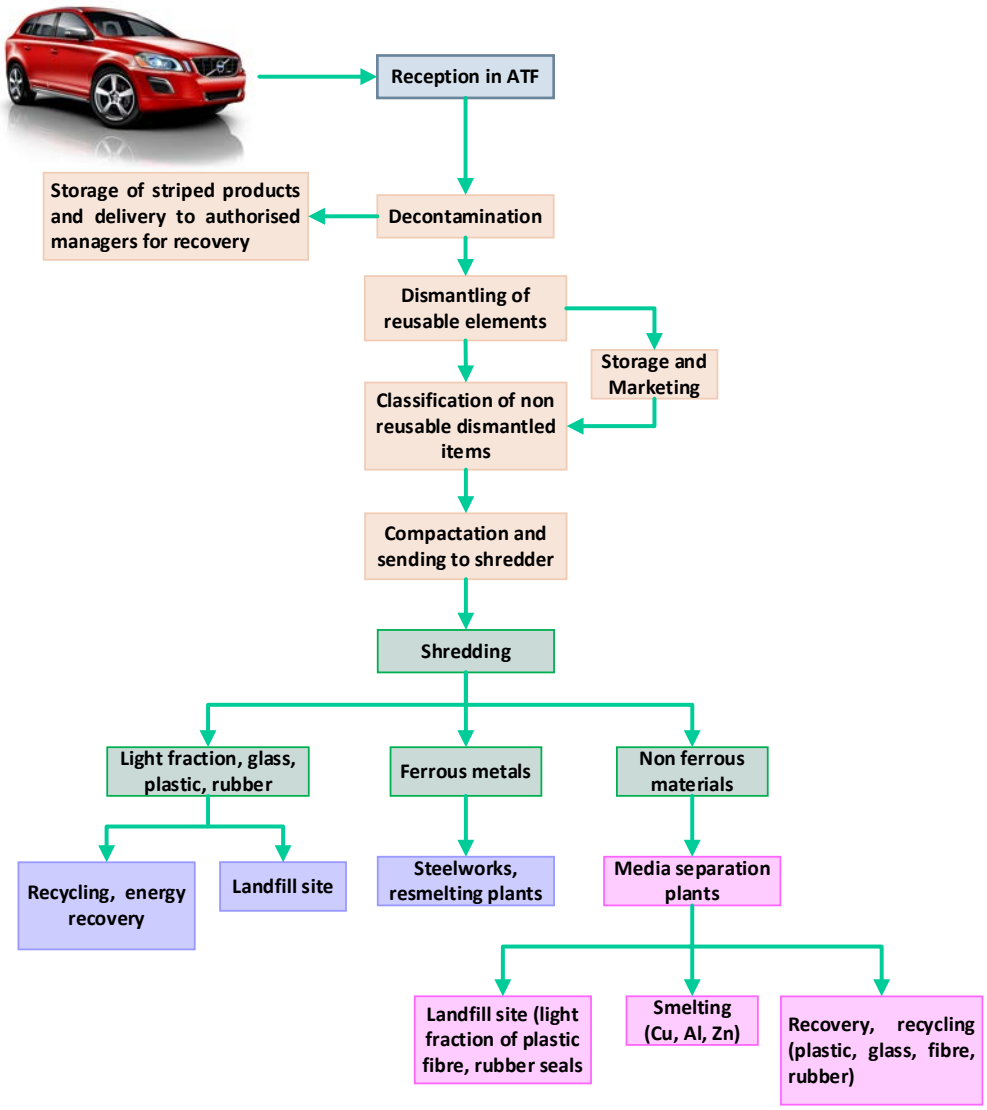


Figure 5.3. ELV treatment process (García, 2011)

5.2.3 Waste LCD TV Treatment Process

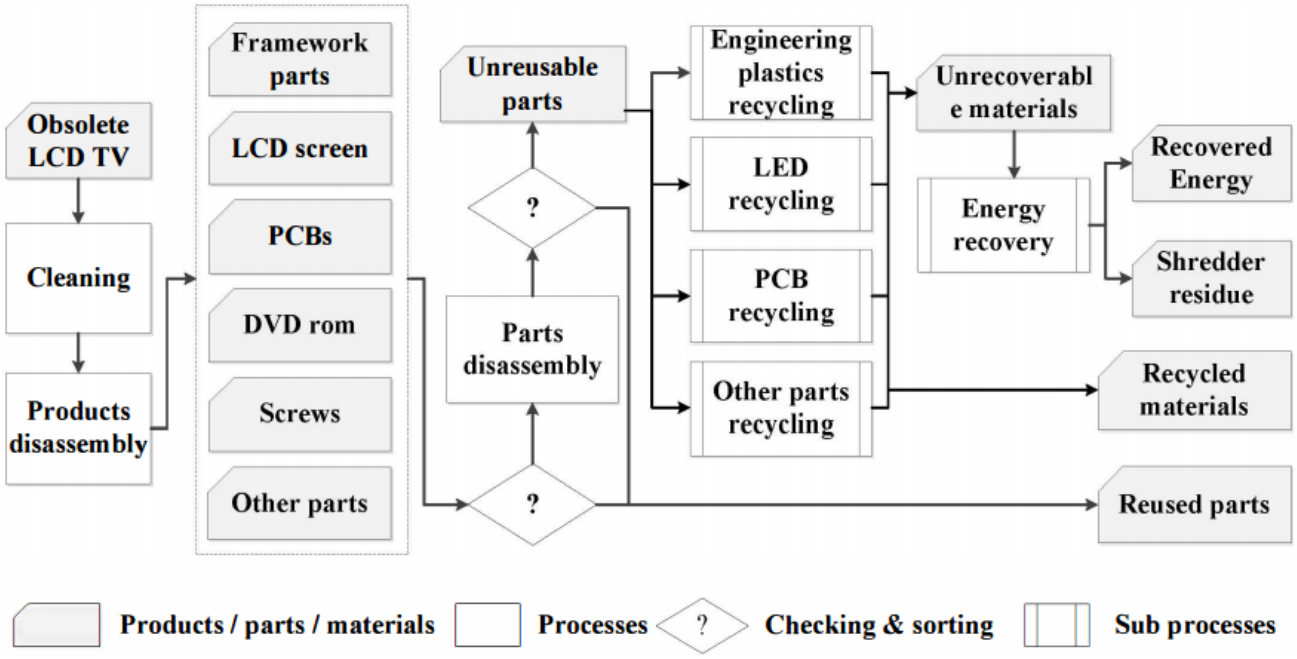


Figure 5.4. Recovery and remanufacturing of LCD TV WEEE (Li, et al., 2012)

Figure 5.2, Figure 5.3 and Figure 5.4 show the recycling process of a mobile phone, a vehicle and a LCD TV. As can be seen in the examples, when the technological products are received by the ATF they are cleaned and/or decontaminated from hazardous components and dismantled. Then the reusable components are retired and sent to the proper storage facility that will manage their reuse. Ferrous metals, non-ferrous metals, plastic, glass, fibres and rubber are recovered and treated so they can be use in the remanufacturing process. Materials that can be incinerated to recover energy are sent to the correct treatment facility and the rest are disposed of in landfill sites.

In In this scenario, the proposed framework of integration between AR and PLM described in this thesis could provide with an information system able to manage the recycling information related to the technological products. This includes the identification of the components of the product including features such as component revision, variants, etc, and their qualification as elements in new products. The system could also supply the type of treatment that should be applied to the specific component, the instructions for the correct dismantling process and the tools that should be used and where, to name a few examples. The instructions supplied could help the operator to make the correct recycling decision, which could reduce the number of errors committed during the treatment and optimise the performance of the operator while the efficiency of the process is improved.

6 PLM Architecture and AR Integration

In order to describe the structure of a system and the connectivity of its parts, it is necessary to define the fundamental components and their relations. The subsequent paragraphs will present the definition of the framework for the integration of AR and PLM.

6.1 AR-PLM Integration Framework Requirements

In order to achieve a successful integration between AR and PLM, the system should count with some basic pre-requisites needed in order to correctly define and customise the framework of integration. The pre-requisites for the AR-PLM integration are: a product information embedded device (PEID), a PEID reader to extract the product's information, a tool to write the code, a code compiler, the middleware of a PLM system which contains the product life cycle information, and an AR device to display the requested data.

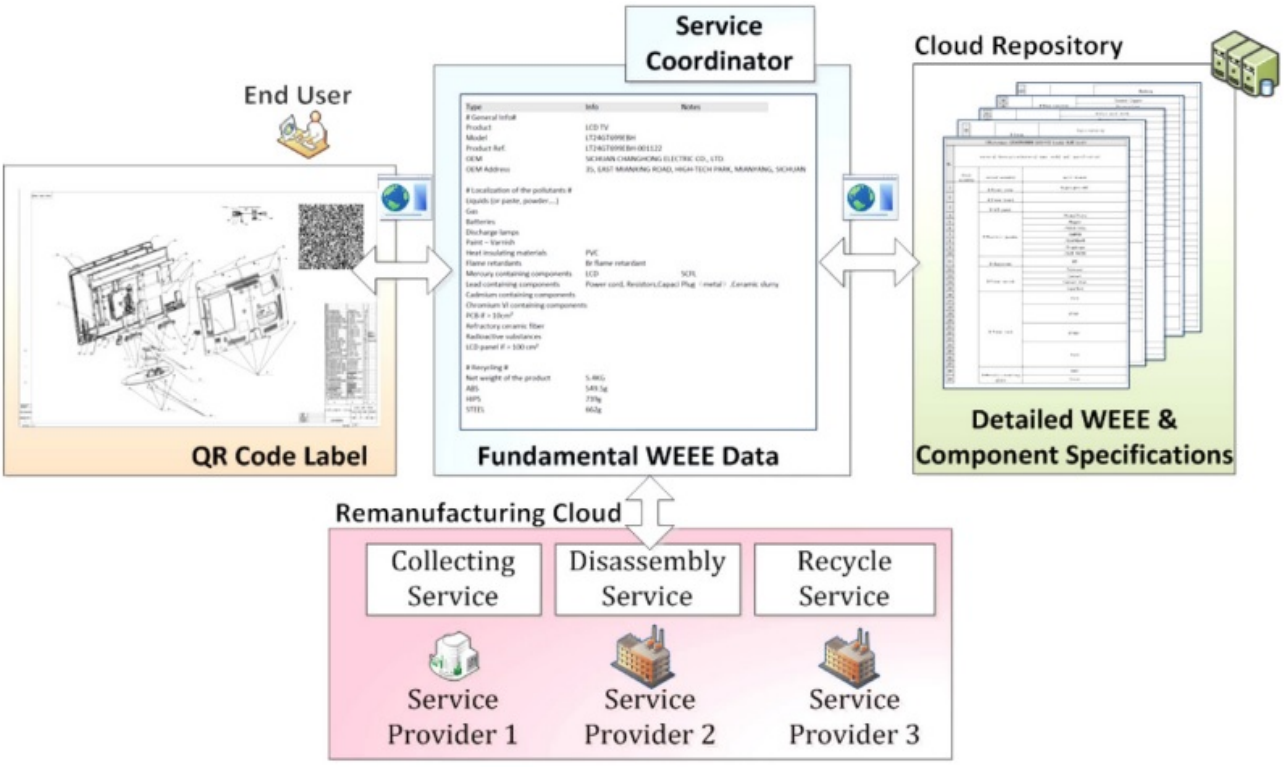


Figure 6.1. QR code-based product tracking mechanism (L., et al., 2014)

Figure 6.1 shows a possible schema of the framework of integration between AR and PLM. The basic operational process of the AR-PLM interface works as follows: a product's PEID is read with the AR device in order to obtain the product ID. Once the ID has been obtained, a request of

information about the decommissioning process of the product is sent to the middleware of the PLM system, which in turn, searches for the desired information and returns it to the AR device. Finally, the AR system receives the necessary information for the correct treatment of the technological product and superposes it upon the real world so the user can follow the supplied instructions for the recycling process.

6.2 Product Identifiers and PEID Architecture

Nowadays an immense number of EEE products and vehicles are produced in the manufacturing industry and sold in the market. These technological products may have many variants and revisions, each of which is typically an "instance" that belongs to a "series" which has been manufactured in batches. These products are often formed by a profuse number of items whose quantity varies with each product and/or variant. A product may be composed of many assemblies which in turn are constituted by several sub-assemblies which at the same time are comprised of a myriad of parts or components. The manufacturing process is commonly done in two ways: the whole product is produced by the same company, or its different constituents are produced by different companies and later assembled by the main manufacturer. This means that every product needs special unambiguous identifiers which properly define the product, variant, series, batch, assembly, component, etc., so their information can be correctly tracked from the beginning to the end of their lifecycles.

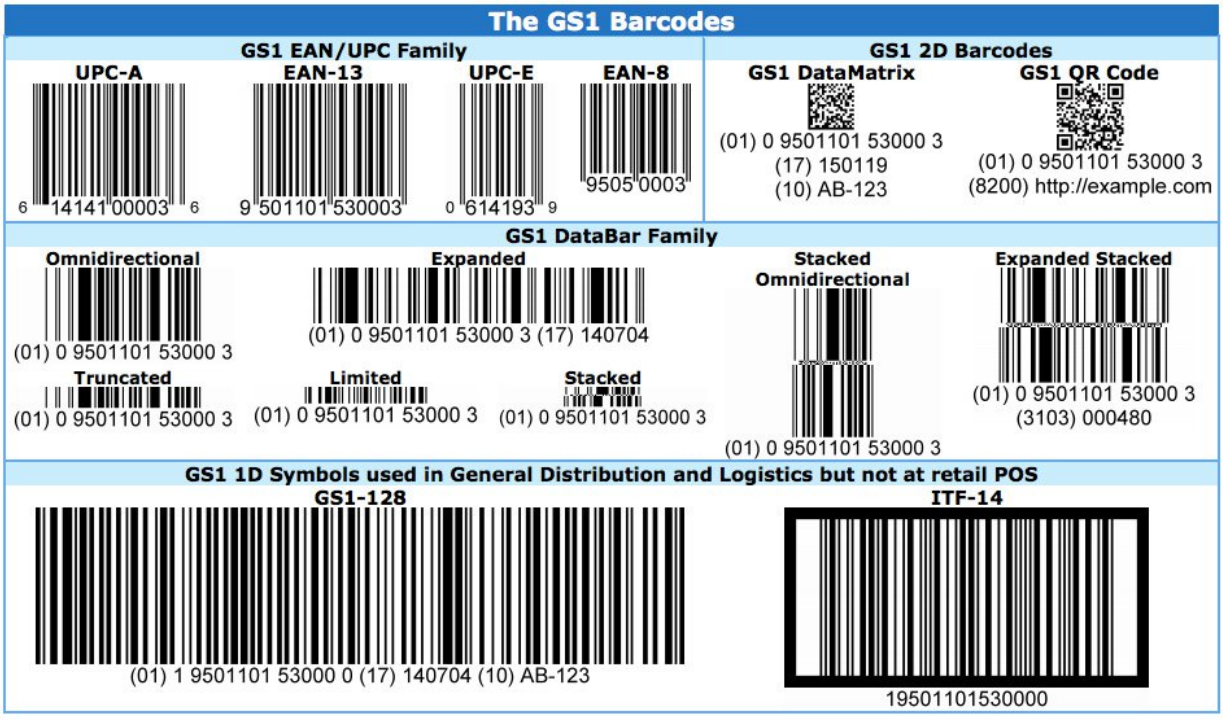


Figure 6.2. Example of GS1 standardised barcodes (IndiaFilings, 2015)

In order to unify the identification of products and parts, there are some standards that can be used such as the global trade item number (GTIN), global location number (GLN), international mobile equipment identity (IMEI), CAS registry numbers (CAS#s) and unique identifier or unique key, among others (Stark, 2011). Figure 6.2 and Figure 6.3 show examples of standardised barcodes and product identification labelling.

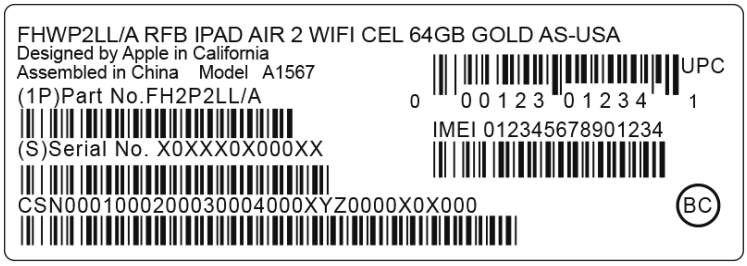


Figure 6.3. Example of product identification label (Truesmart.com, 2016)

The development of product identification technologies allow the embedment of the information in a product through PEID which enable retrieval of a product's lifecycle information in real time at any time and place. This way the information is easily accessible, controllable and traceable throughout the whole product life cycle.

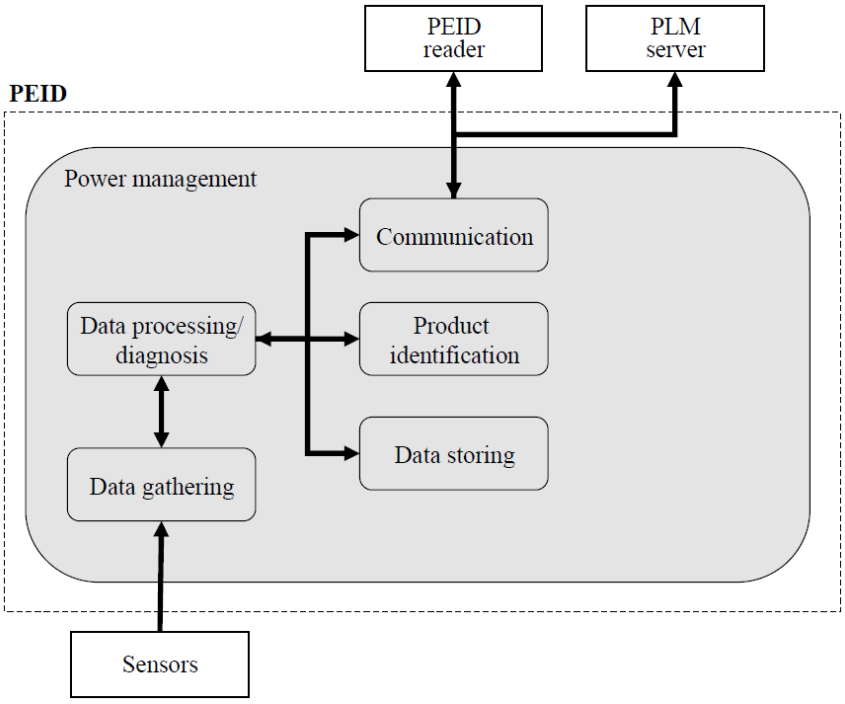


Figure 6.4. General architecture of an active PEID (Kiritsis, 2007)

There are several types of PEID which can be divided into two main groups: PEID that only contains a globally unique product ID (GUPI), it may or may not include data storage capabilities but have no

computation capabilities, for example: quick response (QR) codes, barcodes, passive radio-frequency identification (RFID) tags, etc. And PEID that is able to identify itself (see Figure 6.4) when it is near a sensor, have computation power and wireless network connectivity or internet connection, e.g. vehicle electronic control units (ECUs), active RFID, universal plug and play (UPnP), on-board computers, etc. (Kiritsis, et al., 2003; Kiritsis, 2007)

Kiritsis (2007) defines two types of communication protocols between PEIDs and other systems, short and long distance. In a short distance protocol of communication a PEID interacts with a PEID reader or another PEID through Bluetooth technologies or short range radio waves, e.g. communication between RFID tags and on-board computers, or barcodes and a barcode scanner. A long distance communication a PEID exchanges information with a remote PLM server, a PLM business app connects with a decision support app, etc., for example the communication between a personal digital assistant (PDA) and a general packet radio service (GPRS).

6.3 AR System Architecture

There is a myriad of AR displays which can be classified into three different types: hand held display, spatial and head mounted display, see Figure 6.5.

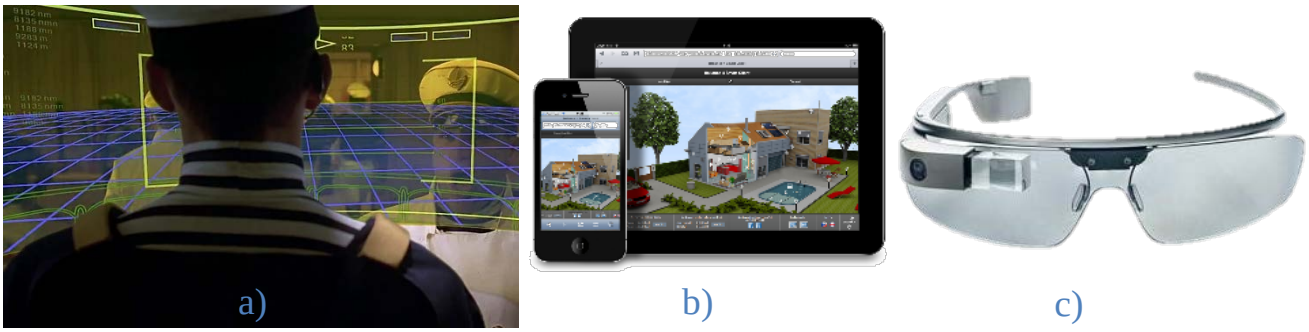


Figure 6.5. Example of AR displays, a) spatial, b) hand-held and c) head mounted

Hand held displays are those that in order to be used need to be held with one or both hands. Spatial displays can be divided into video, optical and projective displays. In this case the desired information is projected directly over the physical objects and no special eyewear is needed. And head mounted displays are those displays that are worn on the head and allow hands free movement.

6.4 PLM System Architecture

In general, a PLM system is formed by a basic assortment of functions and features that may include a data vault, a series of utility functions and sets of user-directed functions. From all these functions it is believed that the foundation of the PLM system is constituted by the data vault, which encloses all the data of the product life cycle and controls the connection to externally generated data (Xu & Liu, 2003). In most cases, the data vault stores two main types of data: application generated data and metadata, which are the product information generated from applications such as AutoCAD, CATIA, Creo Pro/engineer, etc., and data about PLM controlled information such as information models, application logic, analytic models, etc. In paragraph 4.1, it was stated that an efficient PLM system should count with five core functionalities in which data can be stored, managed and retrieved. The user-directed functionalities are data vault and document management, engineering change management, product structure management, parts management and system management; and the utility functions are design communication, product data transfer and distribution, product data translation and conversion, product data viewing and visualisation and system administration. Since both, the user-directed and the utility functions, are connected to the network framework, the modification of the procedure in this section enables the PLM to operate in any computing environment (Xu & Liu, 2003).

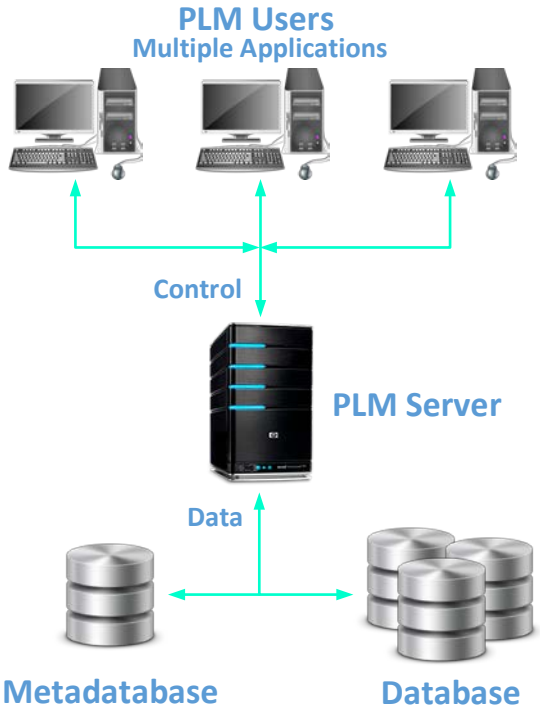


Figure 6.6. General PLM system architecture

In a general PLM system architecture (see Figure 6.6), multiple end users which may utilise different applications are allowed access to the main PLM server. Then the PLM server uses metadata obtained from the metadatabase to manage the search and acquisition of the required product data and information stored in the data vault. A PLM system typically includes more than one database which can be located in diverse sections of the corporation, and usually uses a local area network (LAN) or Intranet to interconnect all the individual components of the system.

As the manufacturing industry evolves, there is an increasing need for PLM systems that provide a collaborative network environment to which engineers, manufacturers and teams of designers are allowed to access, operate and share information related to the same project from different areas and geographical locations situated all over the world. It is for this reason that the implementation of a web-based system architecture in PLM has become a common solution. The number of difficulties faced by non-web PLM systems in a collaborative environment are highly reduced if a web-enabled PLM system is used instead (Liu & Xu, 2001).

Table 6.1. Commercial PLM systems (PLM Technology Guide, 2008)

Commercial PLM systems				
Vendor	Systems Integrator	Product	Product type	Client Focus
Aras Corp	Innovator	Aras Innovator	PLM Software (Open Source)	Small-Medium
Autodesk	PLM 360			
Dassault Systemes	Enovia	Enovia MatrixOne Enovia SmarTeam	PLM Software	Medium-Large Small-Medium
Oracle	Agile	Agile 9 Agile e6 Agile Advantage	PLM Software	Medium-Large Medium-Large Small-Medium
PTC	Windchill	Windchill Windchill On-Demand	PLM Software PLM On-Demand	Medium-Large Small-Medium
Siemens PLM Software	Teamcenter	Teamcenter Unified Teamcenter Engineering Teamcenter Enterprise Teamcenter Express	PLM Software	Medium-Large Medium-Large Medium-Large Small-Medium
Arena Solutions		Arena	PLM On-Demand	Small-Medium
Ingenuus Software		BPMPlus	PLM Software	Small-Medium
Professional Systems Associates Inc.		CMPRO	PLM Software	Small-Medium
SAP		SAP PLM	PLM Software	Medium-Large
Sopheon		Accolade	PLM Software	Small-Medium

There is a multitude of commercialised PLM products and systems integrators in the market which cover different market demands. Table 6.1 show a list of some of the companies that develop commercial PLM systems and the type of software or service that it is provided. These systems currently available have different client focuses, specialties and platform environments and all of them have enabled web-based services.

6.5 Integration Issues

One of the main aspects of PLM is its logical model data access as a consequence of the engineering and business data integration. Just as there are multiple types of procedures and product information, there is also a myriad of application systems which can be integrated within the scope of PLM. This great potential for the integration of all kind of resources in the supply chain is what designates PLM systems as components of the modern manufacturing systems (MMS) (Gao, et al., 2009).

As stated in chapter 4, the integration of AR and PLM to decommission EEE and ELV at their end-of-life, assist the operators in the reception of the information on what needs to be done, in the precise moment that it is required, and provides accurate, up-to-date information essential to develop the process. However, CIMdata (2006) and Bergsjö & Malmqvist (2008) state the integration of different fields requires the consideration of several elements that pose a challenge, such as the level of integration required between the systems, the definition of the master source of information, and the management of the processes related to both domains. In addition, Crnkovic et al. (2003) affirms that the difference of visions, assumptions and fundamental technologies between PLM and other domains is the main cause for the challenge of integration. Persson-Dahlqvist (2005) indicates that in order to achieve a successful integration the processes, tools and technologies, and the stakeholder's culture and within the domains should be taken into account. Svensson (2003) explains that the legacy in PLM environments consists of a series of application and databases that increase with the integration of new systems whenever there is a demand. However, this type of integration increases the risk of data duplicity and wasted time when the same information is sent to several systems. Burr et al. (2005) states that there are different levels for system integration and recommends to types of integration: Best-in-class and All-in-one. In the first option, first-rate software from each domain is integrated on a higher level. In the second one, the applications are directly connected to the master system. Bergsjö & Malmqvist (2008) and Hallin et al. (2004) defends that the integration should be considered not only on the system level but also in the information level. System level integration unifies information systems either by the connection of the interfaces or by product data

import/export procedures. On the other hand, information level integration unifies the information systems on a data level which has a common information core.

6.6 Level of AR-PLM Integration

The level of integration required between AR and PLM is a key factor whose level of complexity depends on the type of environment used to transfer (unidirectional, bidirectional) the information (Bergsjö & Malmqvist, 2008). In a unidirectional network environment the information flow consist in a packet exchange in two directions which are considered to be different: A→B and B→A (reply), i.e. the information flows in one direction at a time and maintains one side's information confidential while the data on the other side is accessed, e.g. product information requests from AR to PLM. A bidirectional network environment is more complex than the unidirectional transfer. In this case, the information flows in both directions at the same time, A↔B, i.e. the user interface allows the access to information from both, A and B, domains; for example, change requests and notices. A bidirectional transfer environment offers more functionality and associated benefits than the unidirectional transfer but should be used where most value to the process is provided. In addition, the information between both domains needs to be consistent in both transfer network environments. This requires the maintenance of the information integrity, states, replication, defined attributes and metadata (CIMdata, 2006).

6.7 PLM Tiered Architecture

The architecture of a PLM system has evolved along time from the one-tiered architecture of the client/server (silos) model to the current web services model which typically counts with a three-tiered architecture, or most recently, a n-tiered architecture (see Figure 6.7). One-tiered architecture was widely extended in 1980s and consisted of a self-contained massive system which was composed of a graphic user interface (GUI), a processing logic and a database. This systems provide a high performance and are easy to design but are not integrated into a network, which means that they cannot be used for web-based applications. The two-tiered system architecture took over in the mid-1990s and was formed by a network that connected the client tier with the server tier. In this case, a web browser realises a request to a web server, which processes the request and returns the desired response. The user interface and the logic processing are located in the client tier while the data vault is located in the server tier. Moreover, this model allows the division of tasks between both layers, uses secure algorithms, and has a clean modular design which improves the scalability,

allows a quicker processing time when compared to a one-tiered system and appoints this architecture as a more suitable option for a collaborative system environment. However, since the layers of the system are not separated, the whole application must be updated at the same time, which makes them difficult to update. In addition, the client tier can be customise in such a way that enables the modification of rules and settings while the rest of the functions are being preserved, which becomes a hindrance as the system increases its complexity and hinders the implementation of a collaborative system environment due to the lack of absolute independence of both tiers.

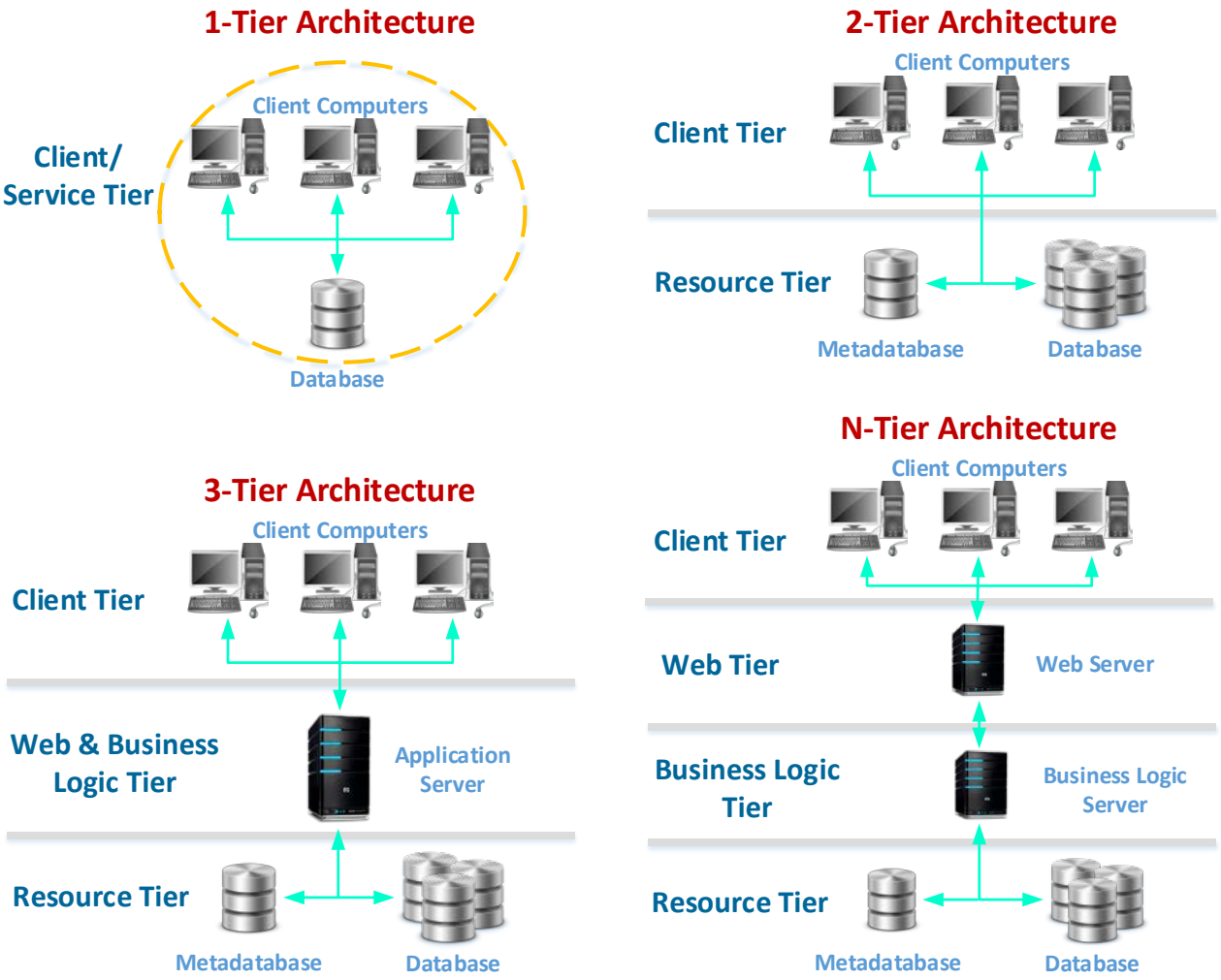


Figure 6.7. Examples of PLM tiered architecture

The three-layered architecture appeared around 1996 when the client tier was divided into two independent parts, the user interface and the logic processing. The consequence of the addition of this new tier resulted in a reduction of the processing and response of the user's requests. The division into client, middleware (web) and resource tiers provides a concurrent efficient data access via linear transactions and supplies a balance between flexibility and complexity which makes it more suitable for medium scale deployments. A n-tiered architecture is a more powerful complex

system able to provide a myriad of services to many clients at the same time. In this type of distributed application an unlimited number of programs run simultaneously on different host computers, in which the exchange information that arrives from different sources and databases is done through standard web services transport protocols that may use more than one type of communication protocol to interact with each other. In this kind of setting, there is need for a common distributed object architecture such as CORBA, EJB, JSP, DCOM, etc. to manage and integrate all operations and applications. Both three-tier and n-tier architecture allow the use of multiple machines at each tier to increase the scalability and failure tolerance of the system. (Liu & Xu, 2001; Xu & Liu, 2003; Liu, et al., 2005; Liu, et al., 2005; Singh, V., 2008; Georgiev, et al., 2007)

6.8 Middleware Software

In pursuance of the integration of any kind of application with a PLM system, the new application should be able to connect and communicate with the resource tier over a communication network. Middleware is a computer software that assists an application in its interaction or communication with other applications, software packages, networks, hardware, operating systems, etc., i.e. is the part of a PLM system in charge of the interoperability (Beal, 2016; Lawton, 2016). PLM systems may use more than one middleware application in order to facilitate the interoperability among the different operating systems and components that may form part of the system. There are several types of middleware available for the integration of external applications with a PLM system, and the selection of one type or another will depend on the type of integration or customization that needs to be done and the architecture of the PLM system since the communication protocol may vary from one tiered architecture to another.

An application programming interface (API) is a set of routines, classes, protocols and structures obtained from libraries and operating systems which is generally used to simplify and provide support for the interaction between two applications or programs. For example, a client, server or database oriented middleware will usually supply an API to access the client tier, the server tier or the resource tier respectively. In addition, the PLM software may require the exclusive use of code extensions that use published and documented APIs which have been authorised by the developer. Common used API standardised programming languages are C++, Java, C# (.NET), etc. (Bergsjö & Malmqvist, 2008; Rachuri, et al., 2008)

6.9 Service Oriented Architecture (SOA)

A service oriented architecture (SOA) is an approach in which APIs provide services to other components of the system through a communication protocol, e.g. network. Although SOA is commonly deployed through web services, its principles are independent of any vendor, product and technology, so its architecture is independent from the rigid server and client architecture of a multi-tier environment. SOA allows the creation of highly scalable information systems and provides interoperability and reusability which facilitates the interaction between different applications which can be both, self-owned and third party applications.

Figure 6.8. SOA runtime deployment n-tiered PLM system (Chesney, 2012)

A SOA deployment may vary with the type of tiered PLM system architecture, Figure 6.8 shows and example of a SOA deployment in a n-tiered system. SOA uses java services that provide libraries and interfaces that allow the development of applications in different programming languages such as C++, Java, C# (.NET), etc. Typically, SOA Java services utilise hypertext transfer protocol secure (HTTP/S) based communication protocols, as can be the simple object access protocol (SOAP) and the web services description language (WSDL), to provide a communication network between the different tiers of the PLM system. HTTP/S communication protocols use extensible markup language (XML) standardize documents to interact with the service provider and the service customer. XML based protocols support the exchange of information with other information exchange standards like the electronic data interchange (EDI), standard for exchange of product model data (STEP), product data markup language (PDML) or business process modelling language (BPML) among others, which contributes to the interoperability, reuse, scalability and flexibility required in both, local and global deployments of PLM systems. (Bergsjö & Malmqvist, 2008; Rachuri, et al., 2008; Siemens PLM Software, 2013; Hallin, et al., 2004)

The server tier of a PLM system supports different run time environments such as Microsoft .NET, IBM Web Sphere, Java 2 Enterprise Edition (J2EE), etc. SOA requests in the web application server can be done through standard HTTP POST, SOAP-style requests, etc. The request is then normalize to the business logic language of implementation, e.g. C++. (Bergsjö & Malmqvist, 2008; Rachuri, et al., 2008; Siemens PLM Software, 2013; Huang, et al., 2004)

The resource tier server and its databases can be access through the API provided by the database oriented middleware. There are different types of database API of which, open database connectivity (ODBC) and Java database connectivity (JDBC) are two examples. ODBC is language independent and allows the access to both types of databases, relational and non-relational. Drivers are used to translate the middleware into the database management system (DBMS) specific commands. On the other hand, JDBC also allows the access to the database made by client programs but is a specific API created for Java language, consequently it is language dependent. Java applets can also count with a JDBC driver to enable their communication with a relational database.

7 AR-PLM Interface Components

In order to demonstrate and implement the ideas described in chapters 5 and 6, a demonstrator has been developed as a proof of concept. This chapter defines the framework for the integration of AR technology with Siemens Teamcenter (PLM software integrator) for the purpose of recycling an EEE or ELV product.

7.1 AR System Architecture

The main purpose of this project is the definition of a framework for the integration between AR and PLM with the purpose of recycling EEE and vehicles. It is logic to assume that in order to properly develop the recycling process the information related to the decommissioning process must be superposed upon the real world by a device which permits hands-free movement, i.e. free use of both hands at the same time. Therefore, the selected AR display has been a see-through HMD developed by Penny model C Wear Interactive Glasses Extended 20, see Figure 7.1.

Figure 7.1 Penny C Wear Interactive Glasses Extended 20

C Wear Extended 20 fully enables the display and projection of graphics upon the real world with a projection display system (PDS) that supports a resolution of 853x480 pixels. The focal plane is placed beyond 2.1 m, at this distance the size of the perceived image is 178 cm, and the field of view (FOV) is 40ºx25º. The glasses need an external system to work, e.g. computer, tablets, cell phones, etc., the internal and external communication to this host is done through HDMI and micro USB 2.0 by the spider embedded system (SES) which also enables the interaction with the graphics via a motion tracking system (MTS) and a command button system (CBS). Both MTS and CBS use a plug and play human interface device (HID) so the glasses can be directly connected to the external host as a mouse. (Penny AB, 2015)

However, this device is not an AR system but a mixed reality (MR) system due to the absence of external sensors for real time interaction with the real world, e.g. a camera, this means that

modifications are needed for its use as AR system. The proposed solution is the connection of an external webcam to the computer host, and the posterior integration of the webcam as an external sensor for the AR-PLM interface. The implementation of the webcam into the AR system is done with a software platform called Eclipse which is formed by a set of open software tools used to develop rich client applications. The software was originally developed by IBM in 2001 but it is currently managed by the Eclipse Foundation and has become one of the most widely used Java IDE (The Eclipse Foundation, 2016). The implementation of the webcam in Eclipse is done with the addition of the open source software library webcam-capture developed by Bartosz Firyn (SarXos) (Firyn, 2015). This solution enables the interaction with the PEIDs and a video-based digital view of the real world that supports the superposition of virtual objects in real time upon the real world is achieved.

7.2 Product Identification, PEID and Transfer Protocol

This paragraph includes the information related the product identification, a description of the case of study implemented in the demonstrator and the definition of the recycling instructions, the type of PEID used for the construction of the prototype, and the communication protocols and level of integration between the PEIDs and the AR and PLM systems.

7.2.1 Teamcenter Product Identifiers

Teamcenter manages a product's life cycle information by the use of fundamental data objects called business objects, for example items, item revisions, documents, parts and designs, see Figure 7.3.

Items are structures used to represent a product, part or component, and is the parent type for many objects which include, document part and design types. It can also include other types of data objects, even other items and folders. The quantity of data objects owned by an item remains the same despite of the user's workspace being used. Item revisions are data objects used to manage the changes done to the items. A document represents all the revisions of a real-world document and provides support for products, processes, standards, etc.; the typical format for documents depends on the application used to create it. Examples of documents formats are DOC, XML, XLS, TXT, etc. A part is a business object in the product structure that might have one or more CAD designs associated with it. It is managed by the releasing system and its revisions and releases are separated from the associated design. A design is a CAD design solution that implements a business part which

might be implemented by more than one CAD design. Likewise, CAD designs can represent more than one part. (Siemens PLM Software, 2013)

When a product is created, it is commonly composed of many pieces of information that describe the item or are related to the item and its revision. Typically, the interrelationship between data objects, items and item revisions is automatically defined by relations when certain objects are added or created in an item or item revision. The automatically generated relation types are revision, item master and item revision master, BOM view and BOM view revision, alternate ID, TC_validation, JTSESSION and 3DMarkup and IMAN_Drawing. In addition, the user-specified relation types also relate to objects and item revisions when the new object is introduced into an item or an item revision by the use of paste and paste special menu commands. (Siemens PLM Software, 2013)

7.2.2 Case of Study: Berg Go-Kart

In order to make the AR-Teamcenter framework of integration and the decommissioning process as real as possible, a real product present in Teamcenter database was used as case of study. The product of choice was a Go-Kart developed by Berg Company (see Figure 7.2), this product was already created and implemented in Teamcenter in previous studies, and its resemblance to a real automobile and the complexity of its BOM appointed it as a good choice for the simulation of the decommissioning process of an ELV.

Figure 7.2. Berg Go-Kart

Figure 7.3 shows part of the BOM of the Go-Kart project together with some of the relation types of the project. The types of the business objects present in the BOM include: items ( ), item revisions ( ), documents ( ), images ( ) parts ( ) and designs ( ).It can be seen that all objects

related to an item have the same item ID independently of the type of business object. As can be appreciated in the image, the format for business object identification is formed by six numbers which are unique identifiers within the PLM system. Revisions use the ID of the item they belong to plus a backslash followed by a capital letter that varies with the item revision number.

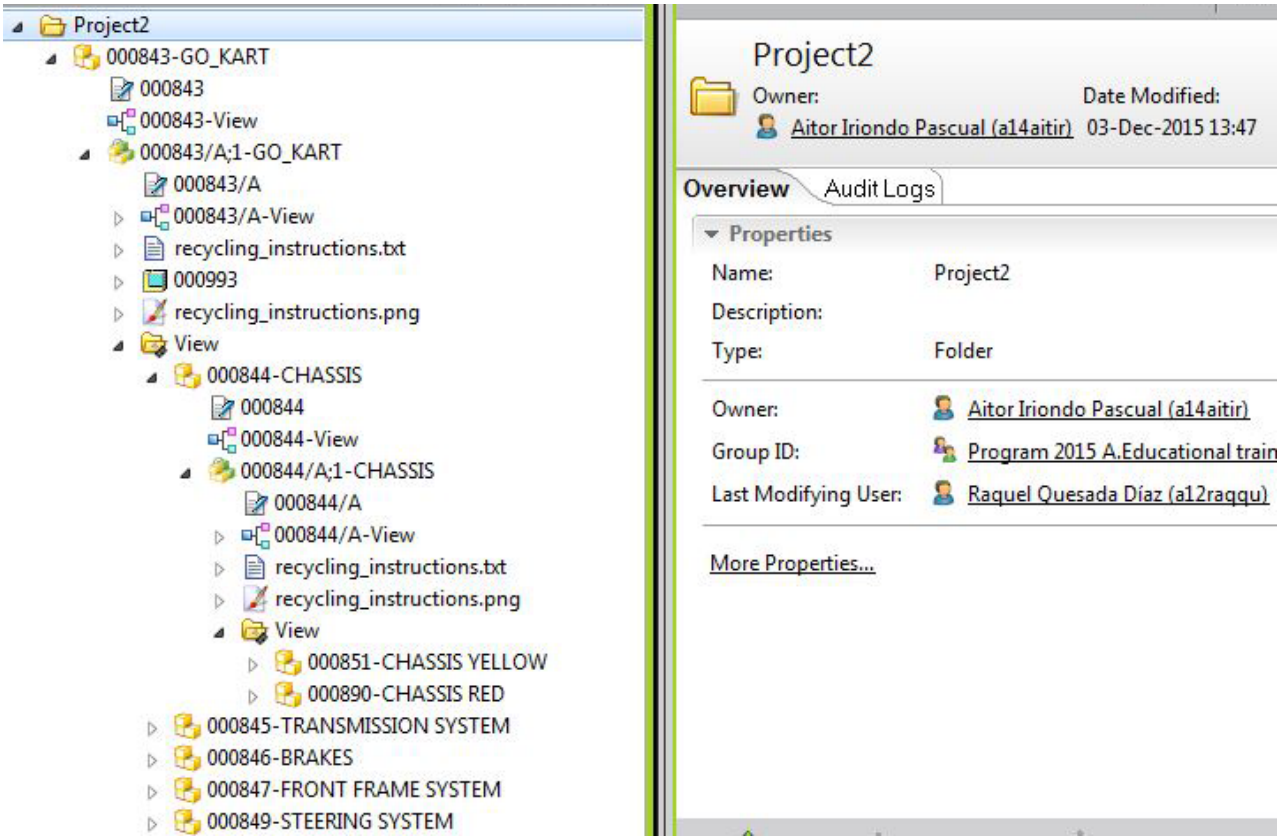


Figure 7.3 Bill of materials (BOM) of product

In order to fulfil the main purpose of the present thesis, two files where created and aggregated to each of the parts of the product, this also includes the main item of the BOM. The first file created had TXT format but XML is also supported, and includes all the information that was considered necessary for the recycling purpose of the Go-Kart product.

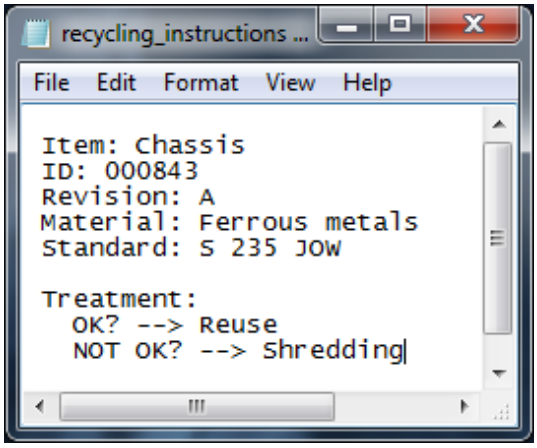


Figure 7.4. Example of a recycling instruction

Figure 7.4 shows an example of the recycling instructions that can be conveyed to the operator during the recycling process. This example was adapted to the available information in Teamcenter, in real products this file may vary depending on the type of EEE or ELV, it might also include information about the manufacturer, model, series, chemical composition, etc. for more information see Figure 7.5 and paragraph 6.2.

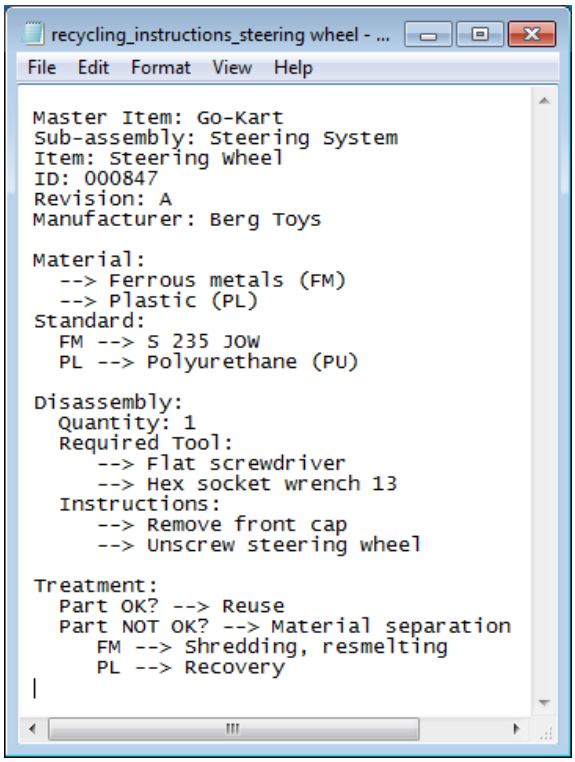


Figure 7.5. Recycling instruction with disassembly process

Go-Kart parts include at least one CAD file created with PTC Creo 2.0, the files were used to obtain the 3D images that have been used to complement the text instructions for the recycling process, for more information see Figure 7.6 and Figure 7.13.

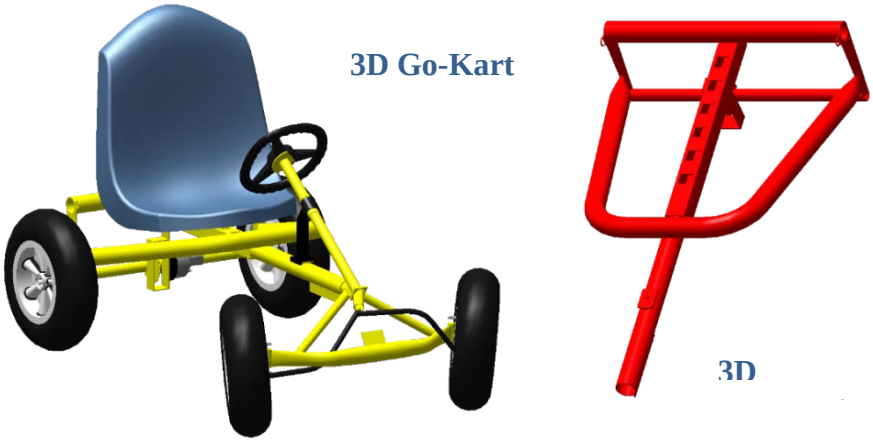


Figure 7.6. 3D CAD models of the Go-Kart and a part of its chassis

For implementation purposes and due to the fact that the image must be superposed upon the real world, the chosen file format for the file was PNG, since it allows the absence of background. In addition, for coding purposes and to due to the fact that every product might be composed of many parts, it has been decided that both, TXT and PNG, files will have the same name for all BOM items, which are recycling_instructions.txt and recycling_instructions.png respectively.

As was explained in paragraph 4.2.1. a good location of the information on the display could improve the performance of the user. Robertson (2008) considered that the performance of the operator increases when the information is displayed next to the target location and lined up horizontally with it, although this was not determinant for all cases. For this reason, and after several attempts with different data locations on the display, it has been decided that the selected location for the information is the left side of the display. Furthermore, in order to allow the standardization of the information displayed on the glasses, the width of the images was set to 300 pixels maximum for all images, although the height can be variable, and located in the coordinates (20, 20) of the screen. Likewise the text was located in coordinates (331, 0).

7.2.3 PEID

From all of the types of PEID explained in paragraph 6.2, it has been decided that only QR codes and barcodes will be used for the purpose of this project, for an example see Figure 7.7.

Figure 7.7 Example of QR code and barcodes format for the demonstrator

The product information that has been embedded in the PEIDs is the item ID, which has been used to identify the parts and send the requests of information related to the recycling process of a product. As was explained in paragraph 7.2.1 there are different kinds of information and relations included in the item ID. So it has been selected three types of item ID formats: item ID only (123456), item revision (123456/A) and item ID plus the last modification date and hour (123456/2015-02-17#12:32), see Figure 7.7 for an example with barcodes and paragraph 7.7 for the code used to manage the item ID to the search for the item.

7.3 Communication Protocol and Level of Integration between AR and Teamcenter

The selection of barcodes and qrcodes as PEID devices and a webcam integrated in an AR system as a code scanner implies that there will not be direct interaction between the PEID and the PLM system. Consequently, it will be used an unidirectional short distance transfer protocol between the PEID and the AR system.

On the other hand, there will be an exchange of information between the PEID reader (AR device) and a remote PLM server, so in this case, it will be used a long distance communication protocol. In addition, if within the domain of this project a user needs to obtain the decommissioning information of a technological product in order to start its treatment process, the user does not need to modify the product information in any kind of way. This means that the recycling information of a product is generated and maintained within the PLM domain so, in this case, it will be sufficient the use of an unidirectional network environment for the integration between AR and PLM.

If in future works there is need to request any change or action which can occur in both, PLM and AR areas, the network environment may require the change from unidirectional to bidirectional data

transfer integration. The implementation of a bidirectional network will require the synchronization of the information exchanged between both domains in order to avoid the divergence and invalidity of the obtained data.

7.4 Teamcenter System Architecture

Siemens Teamcenter® is one of the most widely used PLM systems around the world with the purpose to manage the development of the product and support its entire lifecycle. It was originally developed in the 1980s by EDS PLM Solutions, a company which later became Siemens PLM Software. (Siemens PLM Software, 2016; CIMdata, 2010)

Teamcenter system architecture has been developed in tiers, of which the highest tier is situated closest to the user while the lowest tier is located closest to the database and the metadatabase. Figure 7.8 shows the two possible tiered architectures that can be used to deploy the client and service processes: two-tiered and four-tiered.

In this project, the selected system architecture is four-tiered and consists of a client tier, a web tier, a business logic tier and a resource tier. As can be seen in Figure 7.9, Teamcenter four-tiered architecture framework has been represented in different colours. The client tier, in blue, is installed on the users' computers and comprises the rich and thin clients. It hosts the client applications and secure file caches, and provides the user interface.

The web tier, in orange, is a Java based application which can run in J2EE or .NET application servers such as JBoss, Weblogic, etc. The web tier is in charge of the communications between the client and enterprise tiers. It routes the client requests to the business logic, serves static content and processes the login requests.

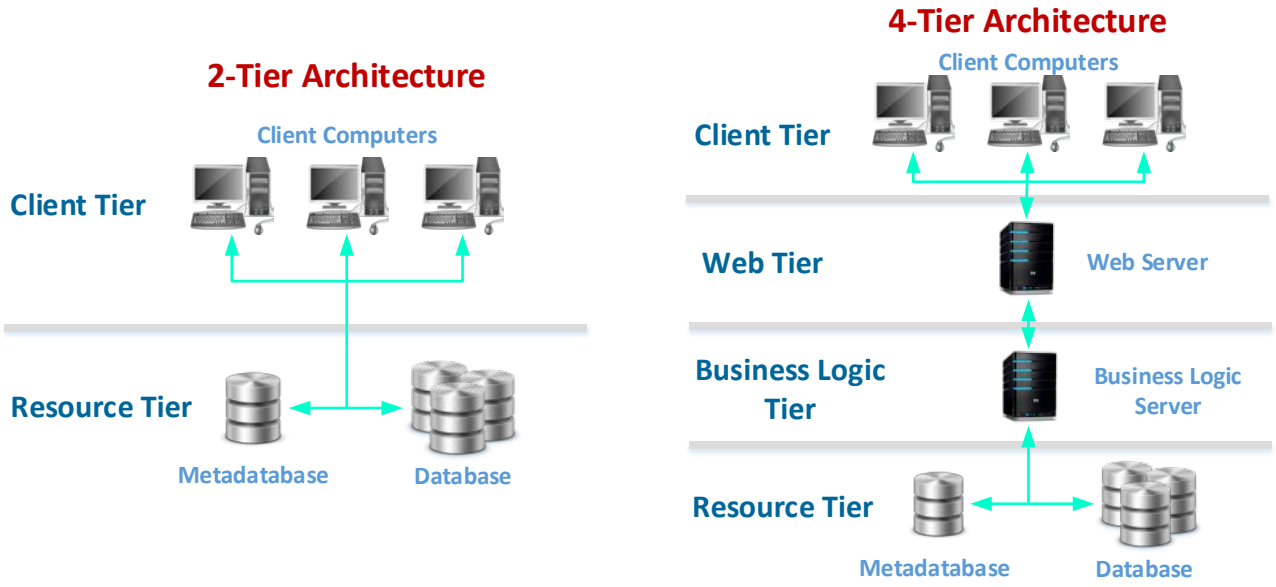


Figure 7.8. Teamcenter system architectures available

The enterprise tier, in green, retrieves and stores data in the database. It integrates a server manager which controls the configurable pool managers which run Teamcenter C++ server processes. This tier hosts the business logic, applies security rules and serves dynamic content. The customization of the server side of Teamcenter must be done in this tier via integration toolkit (ITK) programming.

The resource tier, in pink, contains the database server, metadatabase server, and a file management system. The files for the rich and thin clients, and the Teamcenter lifecycle visualization are uploaded and downloaded by the Teamcenter file management system (FMS). In addition, it also administers a file storage, distribution, catching, and access system.

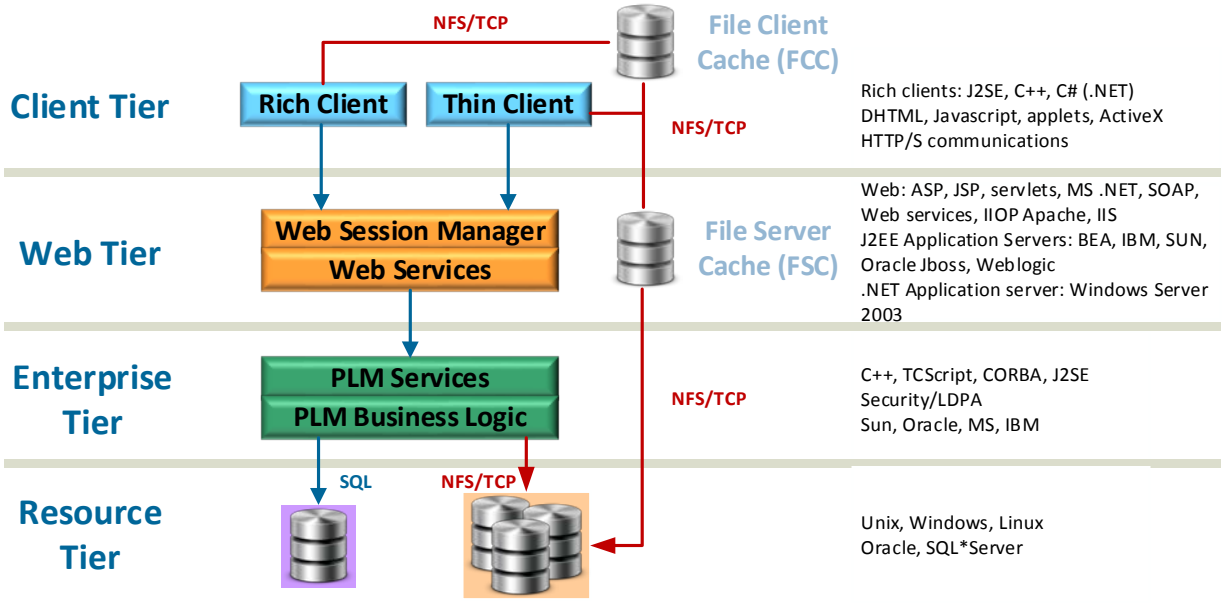


Figure 7.9. Teamcenter reference implementation architecture (Singh, V., 2008)

The access to the resource tier from the client tier depends on the type of information requested. If the requested information is metadata, the request is sent to the web tier which, in turn, sends it to the enterprise tier to be processed and sent to the DBMS. If the access to a file is needed, the request is sent via secure transfer directly to the file database. The information in the metadatabase can be accessed via SQL, this information is stored in a relational database. Teamcenter tables should be read or modified only by the persistent object manager (POM), but other applications can create and access their own tables.

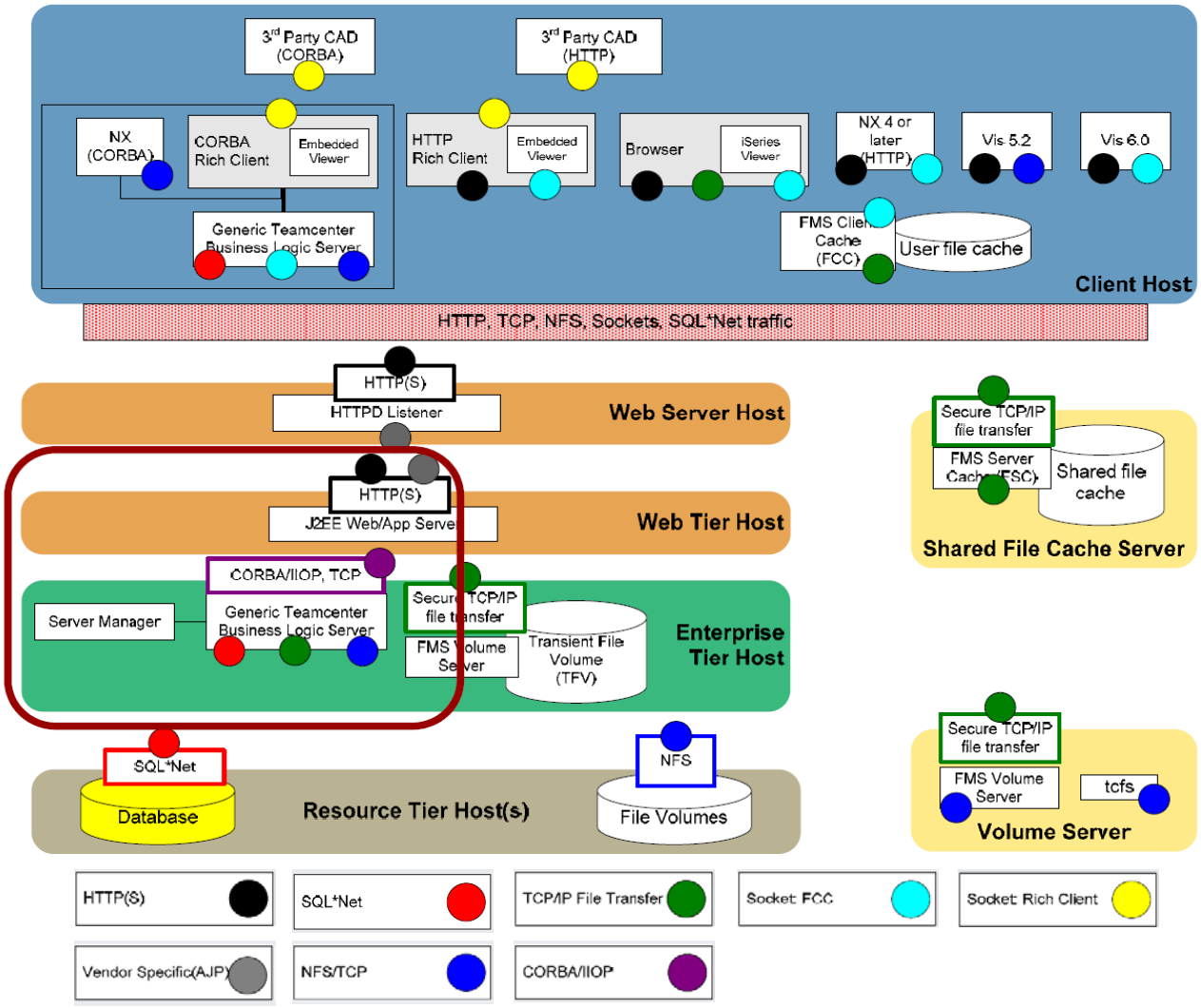


Figure 7.10. Teamcenter J2EE four-tiered architecture (Singh, V., 2008)

The type of four tiered architecture chosen for the purpose of this thesis and the different technologies available are shown in Figure 7.10. It has been selected a J2EE system which is a Java platform developed by Sun Microsystems that simplifies enterprise applications and defines a standard for their multi-tier development. In order to simplify the enterprise applications J2EE platform bases them on standardised modular components, provides a set of services for those components and automatically manages the behaviour of the application. J2EE counts with many features such as JDBC API to access the database, CORBA technology to interact with other enterprise resources, and fully supports Enterprise JavaBeams (EJBs) components, Java Servlets API, JavaServer Pages (JSPs). It also uses J2EE application servers, HTTP web servers and proxies, and XML technology. (Huang, et al., 2004). This technology would have been different if a system with .NET technology stack would have been chosen, the part of the system included in the red

rectangle shown in Figure 7.10 is where the change of technology would occur, the rest remains the same.

7.5 Teamcenter Customization and Application Integration

Teamcenter allows different types of customizations and provides several methods that can be used to achieve the desired result. The system allows the modification of both the rich and the thin clients. This customization enables the appearance of the changes enterprise wide, which means that changes will appear in all clients, and includes server side changes. In addition, it can be selected the appearance of the changes only in one of the clients, i.e. only on the rich client or the thin client. The system behaviour can be changed by the integration of external applications. The definition of the objects and rules used in Teamcenter can be done by the creation of new data models. (Siemens PLM Software, 2013)

These customizations can be classified in two main groups which depend on the part of the architecture used for the implementation. Client side customization is done with Java programming language and the use of JFC and Swing classes, and several wizards available; the rich client is customized via Java language while the thin client is modified through JavaScript. This customization enables the integration of new applications such as toolbars, menu bars, etc., and the creation of forms, POM classes, etc., in short, it allows the modification of the appearance of the client user interface and the behaviour of the client. Server side customization can be done via Teamcenter Services, ITK and C++ programming language. (Siemens PLM Software, 2013; 2006)

There are three main methods that can be used for Teamcenter customization and integration of new applications: the business modeller integrated development environment (BMIDE), Teamcenter Services and ITK. BMIDE is a tool that permits the configuration and extension of the data model and enables the definition and modification of the objects, rules, list of values (LOVs) and properties used by Teamcenter. Teamcenter Services is a server side customization method which uses standard SOA to provide a framework. Libraries are provided for Java, C++, C# (.NET) and for WSDL interfaces compliant with the WS-I Basic Profile. The use of this libraries depends on the type of deployment of the system. For example, C# libraries are available in four-tiered (HTTP) deployments, while the other two libraries support both two-tiered (IIOP) and four-tiered architectures. HTTP is the most common transport in WSDL which can be used in both, J2EE and .NET web tiers. ITK interface should be used for coding within the Teamcenter server and the development of new interfaces for Teamcenter Services. It is a low-level interface which commonly

requires more calls than Teamcenter Services in order to execute an action. (Siemens PLM Software, 2006; 2013)

7.6 Teamcenter Service Oriented Architecture

Siemens PLM software recommends the use of Teamcenter Services when the integration of an external application is required. If more functionality is needed but is not exposed with this method, a new high level service method can be coded with ITK and exposed as Teamcenter Services interfaces via the Teamcenter Service toolkit (Siemens PLM Software, 2013). In addition, one of the issues in Teamcenter environment is related to the access of the database (DB) in order to retrieve the information for every component involved in the disassembly of an EEE product. Presently, access manager module in Teamcenter environment does not support the direct access to the DB for all users, this must be done through the use of the SOA which provides with the necessary tools for the development of task-specific clients, utilities and system integrations through Teamcenter server. The Teamcenter Services are accessed through coarse-grained interfaces that minimize the amount of network traffic needed to accomplish any specific task. Consequently, the selected method of customization utilised in this project is Teamcenter Services with SOA framework. Figure 7.11 shows Teamcenter four-tiered architecture framework and shows SOA layers in yellow: client, web services and application.

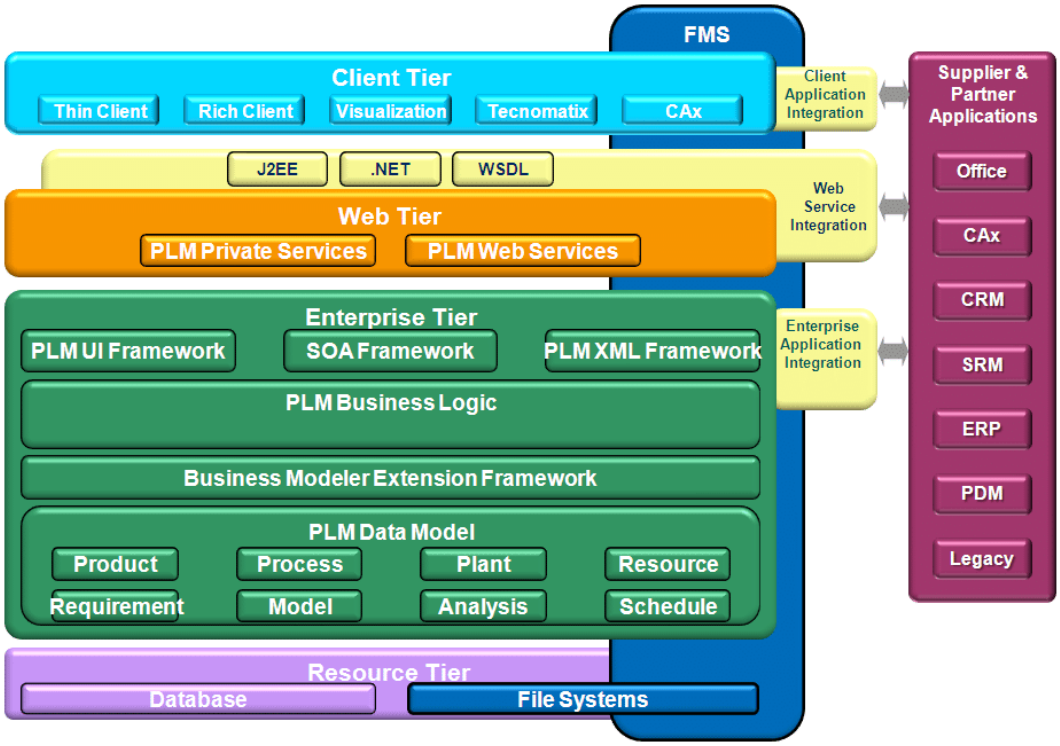


Figure 7.11. Teamcenter architecture framework

In order to use Teamcenter Services and SOA there must be access to a Teamcenter 2007 (or later) server, it is needed Java JDK 7 for Java development, Eclipse must be installed to run Java applications and Microsoft Visual Studio Service Pack 1 in order to run C++ or C# (.NET) applications. If Teamcenter Services WSDL needs to be used, Teamcenter Services WSDL/SOAP support module must be installed through the web application manager. In addition, it must be ensured that Teamcenter Services are enable for use and started, and the client must be configured to use Teamcenter Service client libraries. There is no configuration requirement for the Teamcenter server. (Siemens PLM Software, 2013)

In order to connect a client application to the server, the application must know how to connect to Teamcenter server. In a four-tiered architecture with direct HTTP communication, the connection class constructor needs to know the protocol and hostPath arguments to configure the client application for its connection to the server. The definition of the Web tier address must have **http//:host:port/app-name** syntax. Where **host** specifies the network name of the host machine for the web tier, **port** specifies the port number configured for communications on the web tier machine. For example: WebLogic default port is 7001, or Apache Tomcat uses 8080, etc., and **app-name** specifies the application name in which the Teamcenter web tier services are deployed under. Default value is tc, but it must be checked since it may have changed when the server was installed or configured.

7.7 Application Framework for the Integration between AR and Teamcenter

The selected application development environment is Java and therefore, Eclipse software development kit (SDR) has been used for the development of the application that enables the integration between the AR device and Teamcenter PLM software, see Figure 7.12.

As shown in Figure 7.12, the project in Java has been called **TeamcenterAR** and consists of one folder named **scr** that contains all the necessary Java scripts for the correct implementation of the AR-PLM platform. In addition, Table 7.1 shows the Main Code framework for the integration between AR and Teamcenter and includes a brief explanation of the main functionalities that each of the functions that form the program has.

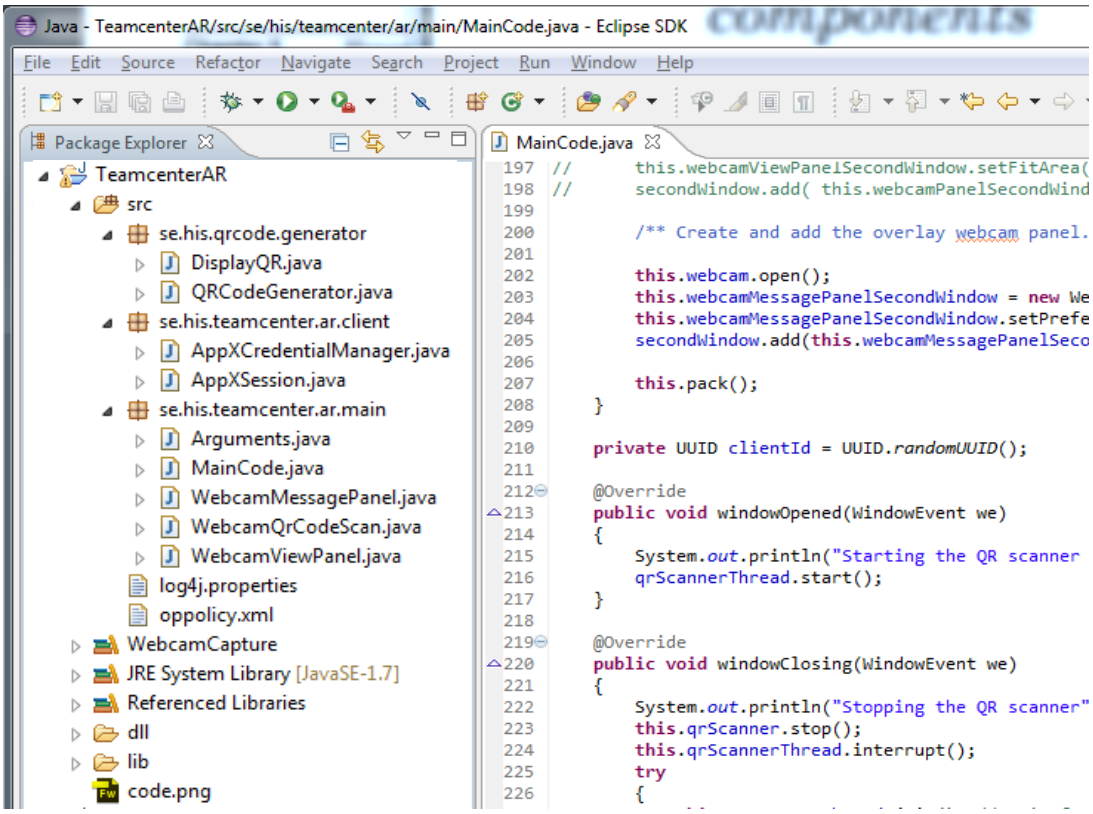


Figure 7.12 Eclipse interface with the necessary Java packages created for the AR-PLM integration (The Eclipse Foundation, 2016)

The scripts have been divided into three packages with regard of their function: the first package, **se.his.qr.code.generator,** contains the necessary scripts for the correct generation of qrcodes and their display on screen. This first package is not really necessary due to the existence of free generators online able to supply both, qrcodes and barcodes.

The second package, **se.his.teamcenter.ar.client,** contains two scripts that enable the creation of a session in Teamcenter, realises the authentication of the user and connects to the server**.**

The third package, **se.his.teamcenter.ar.main,** is composed of the main java application for the implementation (**MainCode.java**), an application to retrieve the host, username and password for the creation of the session and the authentication (**Arguments.java**), the PEID reader (**WebcamQrCodeScan.java**), and the codes for the integration of the webcam, the video feed and the superposition of the text upon the real world (**WebcamViewPanel** and **WebcamMessagePanel**).

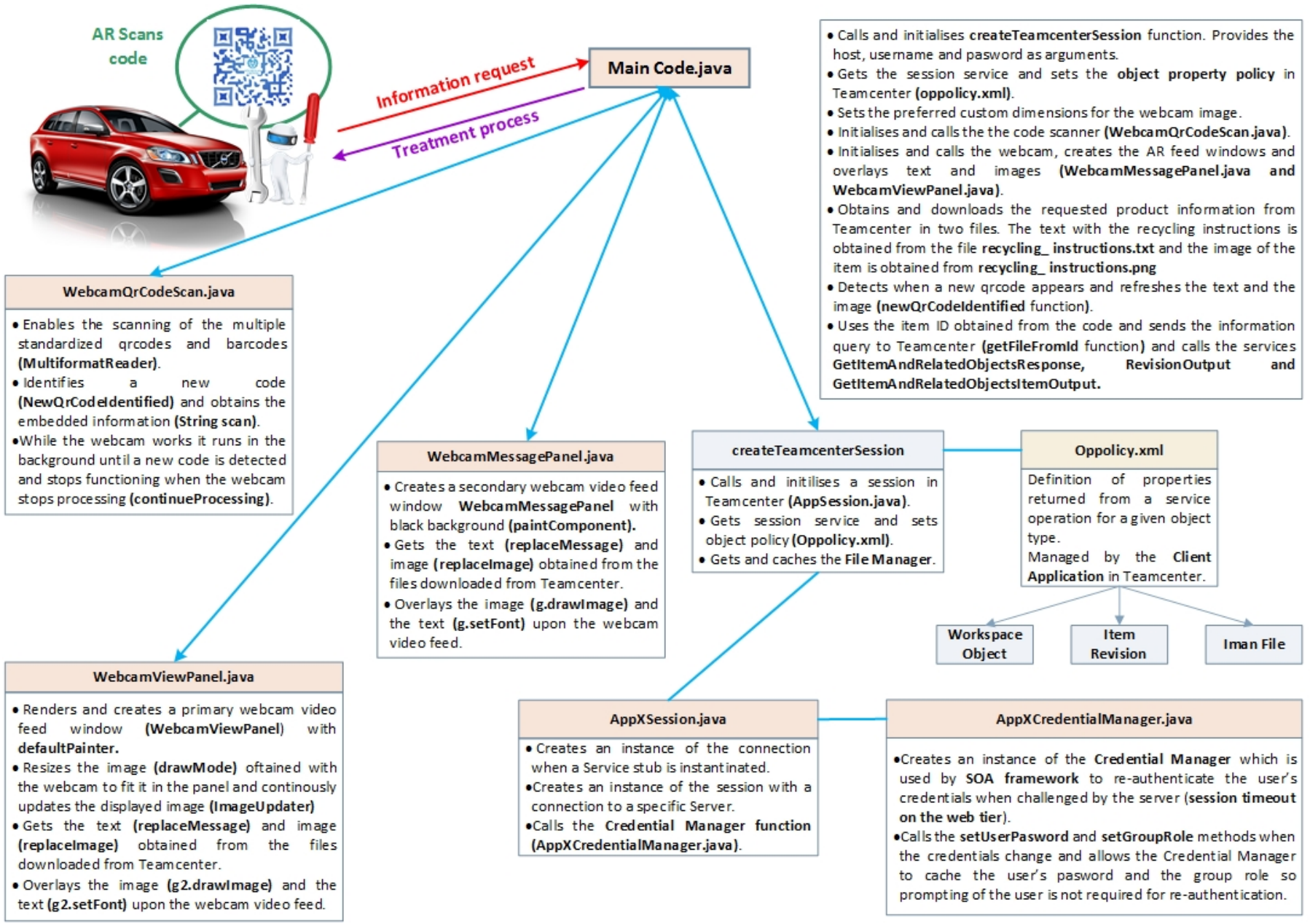


Table 7.1. Main Code framework for the integration between AR and Teamcenter

In addition, there are three folders with the necessary JAR libraries for the correct functioning of the interface (**WebcamCapture**, **JRE System Library** and **Referenced Libraries**). Moreover, it has also been included two additional folders which contain the required DLL files for Teamcenter and the Java Client. Furthermore, there are two XML files both of which are important for the correct functioning of the SOA client and the interface: **log4j.properties.xml** sets the logging for the entire web application. There are two predefined Appenders, **TheConsole** and **AppLogFile** which write the log messages on Eclipse SDK console with a specified pattern, and log messages on a log file with a specified pattern and other attributes. **Oppolicy.xml** which is the object property policy, defines which files will be returned from a service operation for a given object type. The business logic of Teamcenter determines the business objects that will be returned, while the object property policy decides which properties are returned on each business logic. This policy is applied to all service operations and the client application controls which policy is to be used at any given time and how much data is to be returned.

Likewise, it has been implemented a method in the **MainCode.java** file that reads the item ID obtained by the PEID reader and sends the query that will request the text and image files with the recycling information of the item. The code works as follows:

```
public File getFileFromId(String id, String filename) throws Exception
   {
   Reads the string previously obtained from the QR code and sends the query for the text
   and image files.
   Gets the data management service
   Prepares the search criteria
   Calls the service method a client ID is needed (it does not have to be an UUID)

   String[] idParts = id.split("/"); // Sets a separator for the ID string

   Gets all items associated with the ID stored in the QR code
   Processes/retrieves all revisions associated to a found item
   Processes/retrieves all data sets associated to a found item

   infos[0].datasetInfo.filter.name = filename; // This filter only retrieves data sets
   with this particular name, i.e. recycling_instructions.txt and
   recycling_instructions.png

   /** Call the services */

   GetItemAndRelatedObjectsResponse response = dmService.getItemAndRelatedObjects(infos);

   If there is no item master with the obtained Item ID throws and error
   If there is more than one item master with the same Item ID throws an error

   GetItemAndRelatedObjectsItemOutput itemOutput = response.output[0];
   RevisionOutput revOutput = null;

      if( Si there is a string of information after the separator “/” )
```

```
        {
           for( Every item revision and its last modification date  )
              {
              if( If there is an item revision equal to the one being searched)
              {
                 Item found;
                 break;
              }
              if(If there is a last modification date equal to the one searched) {

                 Item found;
              }
           }
           if( There is no item)
           {
              Throw an error “Item not found for that item ID”;
           }
        }
        else
        {
           The Item ID only has ID but no revision or modification date;
        }

        /** It is assumed that only one item corresponds with the ID and revision */

        if (Only one item with Item ID)
        {
           return item;
        }
        else if(More than one item with Item ID)
        {
           throw new Exception("The item revision has ambiguous data");
        }

        Dataset dataset = revOutput.datasetOutput[0].dataset; //Gets the dataset

        Assumes there is only one reference contained in the data set (the actual file)
        return Obtains the dataset;
        }
}
```

A substantial part of the **MainCode.java** and **WebcamViewPanel** files were developed by the experienced programmer Marcus Barstorp based on the identified requirements, the rest of the code was developed or modified by the author. In addition, the **Arguments.java** file was also developed by Marcus Barstorp and **WebcamQrCodeScan.java** file was developed by the author. As stated in paragraph 7.1, **WebcamViewPanel.java**, has been developed by Bartosz Firyn (SarXos) and modifications to the original code were done by Marcus Barstorp and the author. Moreover, **AppXSession.java** and **AppXCredentialManager.java** were fully developed by Siemens PLM Inc. For more information related to the authoring, the framework of integration and the relation between the different components of the interface see Table 7.1 and Appendix C: to Appendix K:

7.8 Results

Figure 7.13 presents the outcome of the construction of the prototype to prove the validity of the theoretical concepts used to define the general framework of integration in Chapter 6, and the practical case of study and proceedings for the integration between the C Wear Extended 20 glasses and Teamcenter PLM Software described in this chapter. When the integrated webcam detects a QR code or a barcode, a request is sent to Teamcenter which, if there is an Item with the provided Item ID, returns the information presented in Figure 7.13, otherwise the systems throws and error. Both of the screenshots presented in Figure 7.13, have been obtained from a duplicate of the AR user interface created for this purpose. The QR codes located on top of some of the Go-Kart parts are the target image for the AR functionality.

Figure 7.13. Proof of concept evaluation and display of TXT and PNG information

The results obtained from the implementation of a demonstrator to prove the concept for the described scenario, show that the framework of integration between the C Wear Glasses and Teamcenter is fully functional. Likewise, it is also proved that an AR system can be successfully integrated with a PLM system in order to enable the display of dynamic information content for decommissioning purposes when EEE and vehicles reach the end of their life cycle. Table 7.2 shows the framework for the integration between AR and PLM, and how the product information flows during the whole decommissioning process. The schema represents an iterative cycle in which, when a product is manufactured and distributed, the information regarding its maintenance and collection at the end of its life is sent to the global product information service so it can be retrieved by other stakeholders. For example, the customer that owns a product and wants to know how to maintain it or dispose of it. The decommissioning treatment information is directly provided to the PLM system of the ATF. When WEEE and ELV arrive to the ATF the information embedded in the PEID device

is retrieved by the AR system and a query is sent to the PLM system, which in turn searches for the product information and returns it to the AR device which supplies it to the operator. The operator applies the specified treatment to the technological product and recovered and reusable items are sent to the remanufacturing companies to their use as components in new products. The information regarding the recycling treatment, maintenance and collection of these new EEE and vehicles is then newly introduced into the PLM system of the ATF and the global product information service. In addition, the information about which WEEE and ELV was treated and how its treatment was managed is introduced in the PLM system of the ATF in order to use it to comply with the Directives and regulations in force and sent a report through the EMARS system to the governmental authority in charge of the report collection.

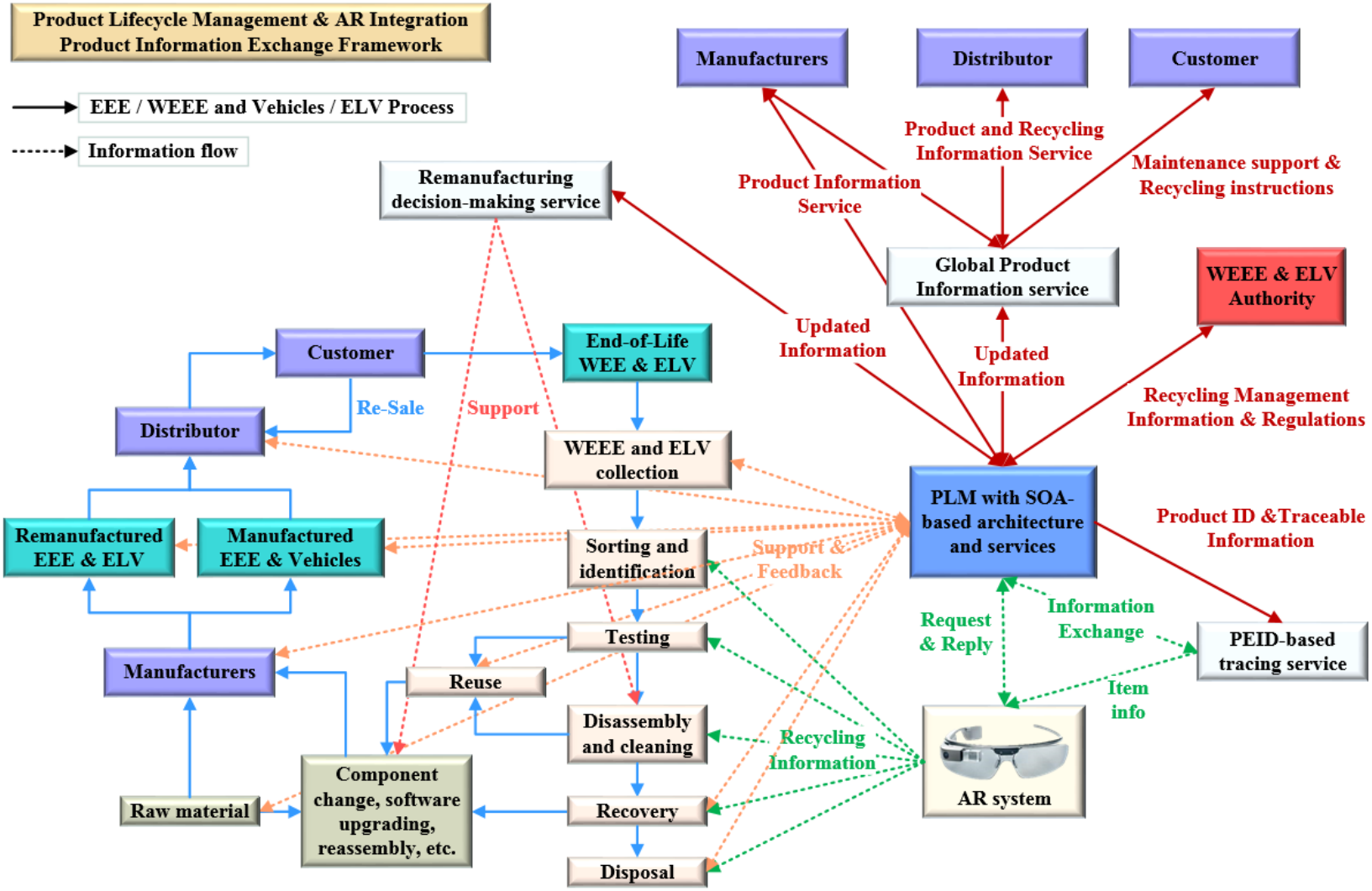


Table 7.2. Framework of integration between AR and PLM and flow of information exchange

8 Conclusions

Previous studies on PLM have focused on its potential of integration with other systems such as CAD, ERP, CSM or MES, and technologies as SOA, J2EE, WSDL or SOAP, among others. However, there has been little focus of study on the feasibility of the integration between AR and PLM systems and its potential as a support system for the Circular Economy. Likewise, previous studies on AR in the industrial manufacturing context have focussed on the development of a framework of integration with the purpose of product assembly, operator training, task learning, machine or installation maintenance, etc., but have paid little attention to the decommissioning process of technological products such as EEE and vehicles. The present thesis argues that the successful integration between AR and PLM for the decommissioning process of WEEE and ELV is possible.

8.1 Summary

The main purpose of the present thesis was to aggregate knowledge about the relation between PLM, AR and circular economy by the definition of a framework that allows the use of AR in a conventional PLM system with the purpose of recycling a technological product at the end of its life cycle. The proposed method of integration is considered to constitute both an innovation and a possible improvement if compared with the current approach. It is believed that the development of a method that addresses the issue of integration between AR and PLM could provide with a secure, efficient management of stored data related to various products and their properties related to the recycling process at the end-of-life of the product. The result of this approach would be an AR-PLM system architecture which assists the circular economy's recycling process by the use of visual information superimposed on the physical technological equipment.

Each EEE and ELV product has a different background and requirements that should be considered for the design of the integration framework. PLM applications provide generic features, while addressing unique PLM requirements. The customization of the application kernel to the context of AR and sustainability involves the possible adaptation of PLM's interface with the addition of new features which will meet the necessary requirements and specifications for the successful integration of AR and PLM.

First, and extensive literature review has been done and data collection techniques have been used in order to obtain the required information about the main fields related to the scope of this project, i.e. AR, PLM, circular economy and the decommissioning of ELV and EEE.

Secondly an analysis and an evaluation of the collected data is done so it can be determined if it is possible to define and develop an architectural framework for integration between AR and PLM for decommissioning purposes. The results obtained from the literature review were positive so the theoretical framework for the general implementation between AR and PLM was introduced and the requirements of the system were defined.

Once the definition of the theoretical framework of integration between AR and PLM was done, the requirements for the implementation of the proposed framework in Siemens Teamcenter PLM Software were defined and a demonstrator was built as proof of concept in order to test the validity of the defined framework of integration. This framework was implemented between C Wear Extended 20 glasses and Teamcenter, and a practical scenario in which the demonstrator was used for the decommissioning process of a Go-Kart was set to prove the validity of the concept in the described scenario.

8.2 Conclusions

The obtained results from the implementation of the prototype showed that the framework of integration between the C Wear Glasses and Teamcenter is fully functional. Also, it is also proven that an AR system can be successfully integrated with a PLM system in order to enable the display of dynamic information content for decommissioning purposes when EEE and vehicles reach the end of their life cycle.

Although the implementation of the demonstrator has proven the validity of the concepts described in the present thesis, the system is far from perfect and needs some improvements.

First of all, it must be noted that the provided software solution has been developed on a prototype level which represents an initial approach to the proposed framework of integration, and therefore, it cannot be considered failure proof.

Secondly, it is important to keep in mind that neither the general framework of integration nor the framework of integration used in the prototype, have been subjected to any kind of assessment or evaluation in terms of feasibility, capability, potential of utilization, acceptability, etc. The prototype

has been implemented only as a proof of concept in order to verify the validity of the proposed method of integration, i.e. it is a laboratory product which has not met reality yet and therefore, it cannot be considered as an early version of the product; its only purpose is to prove the proposed concept.

It is also relevant to note that since C Wear Extended 20 is a MR device, the provided solution to the lack of external sensors supplies a video-based video feed in real time from the webcam which is superposed upon the real world. This is a detriment for the user since the webcam video feed is seen at the same time as the real world and it is extremely confusing. In order to solve this problem, the video feed has been rendered as an overlaid window with black background and enough transparency to allow the superposition of the required product information upon the real world without interfering with the correct development of the task at hand.

In addition, it is necessary to point out that some setbacks have been observed in the use of the available C Wear Extended 20 glasses. The AR device was deteriorated due to its use in previous researches so it could not be adjusted in size to the user's face as the screws were thread stripped. The mirror used in the PDS to reflect the image projected through the lens was worn out and improperly fastened to the holder so only the central part of the image could be seen properly but the rest appeared blurred. Furthermore, the device has to be physically connected to a host device through HDMI and micro USB, which reduces the manoeuvrability of the user. If these considerations are added to a projected image resolution of only 853x480 pixels, the result is a heavy deteriorated image difficult to read due to the distortion provoked by the incorrect fastening of the mirror in the PDS system and the lack of size adjustment, and therefore, this configuration is considered to be highly inefficient and physically demanding.

However, it is important to keep in mind that Penny's C Wear Interactive Glasses is a product which is at an early stage of its development and it is considered that, when this hardware is introduced as a commercial product in the future, it will count with a better screen resolution and improved technical specifications. In fact, the next version of the C Wear Interactive Glasses, which will be named C Wear 30 and be released at the end of 2016, is a proper AR system which counts a mobile computer based on an Intel compute stick. The glasses will integrate full Wi-Fi connection and Bluetooth 4 support. In addition, unlike C Wear Interactive Glasses Extended 20, it will count with a true see-through retina projection with 40º of view and hands free head tracking. All of these new features will increase the functionality and mobility of the AR system, which is highly reduced in the current

system. Consequently, the experience of the user and viability of the implementation in a real manufacturing context will increase significantly.

Although it has been used a real product for the implementation of the demonstrator in order to execute a decommissioning process as real as possible, the definition of the Go-Kart only contemplates the 3D CAD design in the BOM, so the rest of the information related to its life cycle is incomplete. Hence, the recycling instructions provided by the PLM system are only a draft that can be highly improved if all the five stages of the product life cycle are contemplated.

Moreover, this implementation utilises a product whose parts are defined at an early stage of the design process, which means than none of the parts has been approved and released for manufacturing yet, and therefore, they can still be modified. This proved to be a hindrance since it was not possible to use a definite release date as part of the item ID embedded in the PEID. In its stead, it was included the date of the last modification which forces the renewal and update of the information embedded in the PEID every time there is a modification in any of the items included in the BOM. Furthermore, the PEID must be withdrawn and replaced by new ones each time a change in the Go-Kart or any of its components and files is done.

Additionally, when the user needs to obtain information related to a product, it must be ensured that there is only one PEID present in the image, otherwise the system is unable to obtain the information embedded in the PEIDs. Furthermore, it was observed a significant difference in the performance of the PEID reader when it was tested with the type of PEIDs chosen for the study, which included a variety of standardised QR codes and barcodes. While all types of QR codes were read almost instantly and independently of the level of customization, it was observed a significant increase in the time needed by the system to read the information embedded in some the barcodes.

Compliance with EU Directives requires all stakeholders involved in the process of decommissioning of WEEE and ELV to send a report through EMARS with the collection, recovery, reuse, recycling, etc. quotas that has been achieved. This environmental material aggregation and reporting system can be integrated into the PLM system. The combination of integration between AR, PLM and EMARS, may require a change in the level of integration between AR and PLM as it might involve requests of changes of actions which can occur in all areas (AR, PLM, EMARS). This means that the network environment may need to be changed from unidirectional to bidirectional data transfer integration.

Despite these remaining issues to be resolved it is believed that both product lifecycle management, the circular economy and the recycling processes could benefit from the proposed concept.

8.3 Future Work

Even though it has been proved that the successful integration between AR and PLM for the decommissioning process of WEEE and ELV is possible. There are still some improvements and further studies that can be done in future work.

The data used in this project related to the decommissioning process of WEEE and ELV has been obtained from published material only due to the lack of response from companies of the technology and recycling sectors. Consequently, the next desired step in the near future is to interview key personal from companies of the technology and recycling sectors in order to obtain additional details.

In addition, the International Dismantling Information System (IDIS) is a system developed by the IDIS2 Consortium which currently includes all the recycling information regarding 983 different models and 2104 variants from 71 car brands. So it will also be desired the attainment of a copy of IDIS Software so it could be used to realise some experiments with real cases of study.

Moreover, Teamcenter offers a framework for product sustainability and design for environment initiatives that allows the traceability of a product down to the substance level, the limitation in the use of hazardous materials and substances at the same time that the compliance with Directives WEEE, RoSH, REACH, etc. is ensured, as well as the report through EMARS. The utilization of this software extension could be useful to define a real decommissioning case and posterior study of the performance of the system.

It is also intended the repetition and further development of the study with the new version of the C Wear Glasses as well as with other hardware solutions for AR.

Planned future work also includes the realization of experiments that will involve the development of different decommissioning tasks with several levels of difficulty. It is desirable the inclusion of some tasks that include the use of tools for some of the operations involved, since this will introduce additional challenges and approximate the task to a real decommissioning process.

Since the present thesis has focused the framework of implementation between AR and PLM in a four-tiered architecture with J2EE run time environment and technology stacks that include J2EE web tier application, servlets, JSPs, EJBs, JCA, CORBA, J2EE application servers, HTTP web

servers and proxies. It would be desirable the study, development and comparison of the implementation of AR and PLM when a different run time environment that uses another technology stack, for example Microsoft .NET, IBM Web Sphere, etc.

In the development of the prototype, it was decided the creation and inclusion of the new datasets in Teamcenter as files with TXT and PNG formats, but this is not the only method that can be used. It can be also use CAD files, animations, etc. Teamcenter also allows the creation of new data sets via BMIDE that can take the form of new tables and labels in the rich client interface to name an example. It is then proposed the study of the other possibilities and their viability of implementation within the scope of the integration between AR and Teamcenter.

It would be highly beneficial the inclusion of a study regarding the ergonomic aspects of the use of AR wearable technologies in the operational process of recycling WEEE and ELV. For example, the study of hand and head movements, the additional hazardous situations provoked by the use of docked technologies, or the repercussion that the prolonged use of a head mounted display may have for the eyes, or if there exists any secondary effect provoked by this relatively new technology. It will also be interesting the inclusion of a comparative study of body coordination with and without the use of the equipment.

9 References

Electronic Recyclers International, Inc., 2016. *U.S. Landfill Ban.* [Online] Available at: http://electronicrecyclers.com/sustainability/us-landfill-ban [Accessed 20 February 2016].

Afisi, O. T., 2009. *The problem of induction and Karl Popper's hypothetico-deductive methodology: A critical evaluation.*. [Online] Available at: http://philsci-archive.pitt.edu/id/eprint/4514 [Accessed 10 March 2016].

Ameri, F. & Dutta, D., 2005. Product lifecycle management: closing the knowledge loops. *Computer-Aided Design and Applications,* 2(5), pp. 577-590.

Azuma, R., 1997. *A survey of augmented reality,* Malibu, California: Hughes Research Laboratories.

Azuma, R. et al., 2001. Recent advances in augmented reality. *Computer Graphic Applications, IEEE,* 21(6), pp. 34-47.

Azzone, G. & Noci, G., 1998. Identifying effective PMSs for the deployments of "green" manufacturing strategies. *International Journal of Operations & Production Management,* 18(4), pp. 308-335.

Baird, K. M. & Barfield, W., 1999. Evaluating the effectiveness of augmented reality displays for a manual assembly task. *Virtual Reality,* Volume 4, pp. 250-259.

Baldé, C., Wang, F., Kuehr, R. & Huisman, J., 2015. *The global e-waste monitor - 2014,* Bonn: United Nations University, IAS-SCYCLE.

Barreto, L. V., Anderson, H., Anglin, A. & Tomovic, C. L., 2007. Product lifecycle management in support of green manufacturing. In: S. o. M. Engineers, ed. *Proceedings of ICCPR2007: International Conference on Comprehensive Product Realization 2007.* Beijin: s.n., pp. 294-305.

Barreto, L. V., Anderson, H. C. & Anglin, A., 2010. Product lifecycle management in support of green manufacturing: addressing the challenges of global climate change. *International Journal of Manufacturing Technology and Management,* 19(3-4), pp. 294-305.

Beal, V., 2016. *Webopedia. Middleware.* [Online] Available at: http://www.webopedia.com/TERM/M/middleware.html [Accessed 13 May 2016].

Bergsjö, D. Ć. A. & Malmqvist, J., 2008. Implementing a service oriented PLM architecture using PLM services 2.0. In: D. Marjanovic, M. Storga, N. Pavkovic & N. Bojcetic, eds. *Proceedings DESIGN 2008, the 10th International Design Conference.* Dubrovnik: The Design Society, pp. 1-10.

Boud, A., Haniff, D., Baber, C. & Steiner, S., 1999. Virtual reality and augmented reality as a training tool for assembly tasks. In: *Proceedings of the 1999 International Conference on Information Visualisation.* London: IEEE, pp. 32-36.

Burr, H. et al., 2005. CAx/engineering data management integration: Enabler for methodical benefits in the design process. *Journal of Engineering Design,* 16(4), pp. 385-398.

Californian State Board of Equalization, 2003. *California electronic waste recycling Act of 2003,* California State: Californian State Board of Equalization.

Caudell, T. P. & Mizell, D. W., 1992. Augmented Reality: An Application of Heads-Up Display Technology to Manual Manufacturing Processes. *Proceedings of the Twenty-Fifth Hawaii International Conference on System Sciences (HICSS),* Volume II, pp. 659-669.

CEA, 2014. *Third Annual Report of the eCycling Leadership Initiative,* USA: Consumer Electronics Association.

Chancerel, P. & Rotter, S., 2009. Recycling-oriented characterization of small waste electrical and electronic equipment. *Waste Management,* Volume 29, pp. 2336-2352.

Chesney, S., 2012. Practical application of service oriented architecture. In: *2012 Global Product Data Interoperability Summit (GPDIS).* Chandler, Arizona: Siemens PLM Software.

Ciceri, N. D., Garetti, M. & Terzi, S., 2009. Product lifecycle management approach for sustainability. In: R. Roy & E. Shehab, eds. *Proceedings of the 19th CIRP Design Conference - Competitive Design.* Bedfordshire: Cranfield University Press, pp. 147-154.

CIMdata, 2006. *PLM and ERP Integration: Business Efficiency and value,* Ann Arbor, Michigan, USA: CIMdata, Inc..

CIMdata, 2010. *Teamcenter "unified". Siemens PLM Software's next generation PLM platform,* Ann Arbor, Michigan: CIMdata, Inc..

Crnkovic, I., Asklund, U. & Persson-Dahlqvist, A., 2003. *Implementing and Integrating Product Data Management and Software Configuration Management.* 1st ed. Norwood, Massashusetts: Artech House, Inc..

Curtis, D., Mizell, D. W., Gruenbaum, P. & Janin, A., 1998. Several devils in the details: making an AR application work in the airplane factory. In: *Proceedings of the International Workshop on Augmented Reality (IWAR'98).* San Francisco, California: AK Peters, pp. 47-60.

Electronic Recyclers International, Inc., 2016. *U.S. Legislation.* [Online] Available at: http://electronicrecyclers.com/sustainability/us-legislation [Accessed 20 February 2016].

Ellen MacArthur Foundation, 2012. *Towards the circular economy.* [Online] Available at: http://www.ellenmacarthurfoundation.org/publications [Accessed 28 January 2016].

EPA, 2014. *Municipal solid waste generation, recycling and disposal in the United States 2012,* United States: United States Environmental Protection Agency.

European Commission, 2016. *European Commission Environment.* [Online] Available at: http://ec.europa.eu/environment/waste/index.htm [Accessed 3 February 2016].

European Union, 2000. Directive 2000/53/EC of the European Parliament and of the Council of 18 September 2000 on end-of life vehicles (ELV). *Official Journal of the European Union,* pp. 1-22.

European Union, 2002. Directive 2002/96/EC of the European Parlament and of the Council of 27 January 2003 on waste electrical and electronic equipment (WEEE). *Oficial Journal of the European Union,* pp. 24-39.

European Union, 2003. Directive 2002/95/EC of the European Parlament and of the Council of 27 January 2003 on the restriction of the use of certain substances in electrical and electronic equipment (RoSH). *Official Journal of the European Union,* pp. 19-23.

European Union, 2008. Directive 2008/98/EC of the European Parliament and of the Council of 19 November 2008 on waste and repealing certain Directives (EEA relevance). *Official Journal of the European Union*, pp. 3-30.

European Union, 2011. Directive 2011/65/EU of the European Parlament and of the Council of 8 June 2011 on the restriction of the use of certain substances in electrical and electronic equipment (RoSH Recast). *Official Journal of the European Union*, pp. 88-110.

European Union, 2012. Directive 2012/19/EU of the European Parlament and of the Council of 4 July 2012 on waste electrical and electronic equipment (WEEE). *Oficial Journal of the European Union*, pp. 88-110.

EUROSTAT, 2014. *End-of-life vehicle statistics.* [Online] Available at: http://ec.europa.eu/eurostat/statistics-explained/index.php/End-of-life_vehicle_statistics#Coverage_and_targets
[Accessed 15 February 2016].

EUROSTAT, 2016. *Waste statistics - electrical and electronic equipment.* [Online] Available at: http://ec.europa.eu/eurostat/statistics-explained/index.php/Waste_statistics_-_electrical_and_electronic_equipment
[Accessed 10 February 2016].

Feiner, S., MacIntyre, B. & Sellgmann, D., 1993. Knowledge-based augmented reality. *Communications of the ACM*, 36(7), pp. 53-72.

Fiorentino, M., Monno, G. & Uva, A. E., 2010. Tangible interfaces for augmented engineering data management. In: S. Maad, ed. *Augmented Reality*. Croatia: INTECH, pp. 113-128.

Firyn, B. (., 2015. *SarXos: Webcam-capture.* [Online] Available at: https://github.com/sarxos/webcam-capture
[Accessed 3 March 2016].

Gallagher, S., 2015. *Augmented reality gets to work-and gets past the "glassholes".* [Online] Available at: http://arstechnica.com/information-technology/2015/03/augmented-reality-gets-to-work-and-gets-past-the-glassholes/
[Accessed 8 March 2016].

Gao, L., Gao, J., Fan, Y. & Zheng, J., 2009. Research on PLM System Architecture Model and Process. In: *2009 WASE International Conference on Information Engineering, 2009 (Volume 1).* Taiyuan, Shanxi: IEEE Computer Society, pp. 491-495.

García, L. P., 2011. The treatment of end-of-life vehicles in Apain and the trend overtime. *Seguridad y Medio Ambiente,* 31(123).

Georgiev, I., Shikalanov, A., Georgiev, I. & Boyanov, K., 2007. N-tier Architecture for Intercontinental Tracking System based on Lightweight Semantic Web Service Processes. *Journal of Information Technologies and Control,* pp. 1-19.

Glavič, P. & Lukman, R., 2007. Review of sustainability terms and their definitions. *Journal of Cleaner Production,* Volume 15, pp. 1875-1885.

Goose, S. et al., 2004. PARIS: Fusing vision-based location tracking with standards-based 3D visualization and speech interaction on a pda. In: *International Conference on Distributed Multimedia Systems.* San Francisco, California: s.n., pp. 75-80.

Grieves, M., 2006. *Produt lifecycle management: Driving the next generation of lean thinking.* 1st ed. New York, New York: McGraw Hill.

Hage, J. T., 1999. Organizational innovation and organizational change. *Annual Review of Sociology,* 25(1), p. 597–622.

Hallin, K., Zimmerman, T. & Malmqvist, J., 2004. Towards a Framework for Integrated information management in mechatronic product development. In: T. Lehtonen, A. Pulkkinen & A. Riitahuhta, eds. *Proceedings of NordDesign 2004 Conference.* Tampere: The Design Society, pp. 153-162.

Haniff, D., Boud, A. & Baber, C., 1999. Assembly training with augmented reality and virtual reality. *International Conference on human-computer interaction,* Volume 2, pp. 35-36.

Heilala, J., Helaakoski, H. & Peltomaa, I., 2008. Smart Assembly — Data and Model Driven. In: S. Ratchev & S. Koelemeijer, eds. *IFIP — International Federation for Information Processing, Volume 260, Micro-Assembly Technologies and Applications.* Boston: Springer US, pp. 371-381.

Henderson, S. & Feiner, S., 1999. Evaluating the benefits of augmented reality for task localization in maintenance of an armored personel carrier turret. *ISMAR,* pp. 1-10.

Hewlett-Packard Co., 1994. *Product Data Management at Hewlett-Packard Company.* [Online] Available at: http://www.johnstark.com/arthp.html [Accessed 25 March 2016].

Hotta, Y., Santo, A. & Tasaki, T., 2014. *EPR-based electronic home appliance recycling system under Home Appliance Recycling Act of JApan,* Japan: Institute for Global Environmental Strategies.

Hou, L. & Wang, X., 2010. Using augmented reality to cognitively facilitate product assembly process. In: S. Maad, ed. *Augmented Reality.* Croatia: InTech, pp. 99-112.

Hou, L., Wang, X., Bernold, L. & Love, P., 2013. Using augmented reality to cognitively guide assembly. *Journal of Computing in Civil Engineering,* 27(5), pp. 439-451.

Huang, M. Y., Lin, Y. & Xu, H., 2004. A framework for web-based product data management using J2EE. *The International Journal of Advanced Manufacturing Technology,* 24(11), pp. 847-852.

ICF International, 2011. *Electronics waste management in the United States through 2009,* United States: U.S. Environmental Protection Agency (EPA), Office of Resource Conservation and Recovery.

IMSA Amsterdam, 2013. *Unleashing the Power of the Circular Economy.* [Online] Available at: http://www.circle-economy.com/library/books-reports/page/2/ [Accessed 28 January 2016].

IndiaFilings, 2015. *How to Get a Barcode in India.* [Online] Available at: http://www.indiafilings.com/learn/how-to-get-barcode-in-india/ [Accessed 10 May 2016].

INGUN, 2015. *ACPEIP (China RoSH),* Konstanz: INGUN Prüfmittelbau GmbH.

Ishii, H. et al., 2008. Proposal and evaluation of a supporting method for NPP Decommissioning work by augmented reality. *The 12th World Multi-Conference on Systemics, Cybernetics and Informatics: (WMSCI 2008),* pp. 157-162.

Kimura, F. & Kato, S., 2002. Life cycle management for improving product service quality. In: K. Feldmann, ed. *Proceedings of the 9th CIRP International Seminar on Life-cycle Engineering.* Erlangen: Meisenbach Verlag Bamberg, pp. 25-32.

Kipper, G. & Rampolla, J., 2012. *Augmented reality: An emerging technologies guide to AR.* 1st ed. Rockland, Massachusetts: Syngress Media,U.S..

Kiritsis, D., 2007. PLM and product embedded information devices. *IFAC Proceedings Volumes. 8th IFAC Workshop on Intelligent Manufacturing Systems,* 40(3), pp. 8-23.

Kiritsis, D., Bufardi, A. & Xirouchakis, P., 2003. Research issues on product lifecycle management and information tracking using smart embedded systems. *Advanced Engineering Informatics,* 17(3-4), pp. 189-202.

Klinker, G. et al., 1997. Confluence of computer vision and interactive graphics for augmented reality. *Presence: Teleoperations and Virtual Environments. Special Issue on Augmented Reality,* 6(4), pp. 433-451.

L., W., V., W. X., Gao, L. & Váncza, 2014. A cloud-based approach for WEEE remanufacturing. *CRIP Annals - Manufacturing Technology,* Volume 63, pp. 409-412.

Lawton, G., 2016. *TechTarget. SearchSOA: middleware.* [Online] Available at: http://searchsoa.techtarget.com/definition/middleware [Accessed 13 May 2016].

Lee, S., Ma, G. & Vestraeten, J., 2008. Product lifecycle management in aviation maintenance, repair and overhaul. *Computers Industry,* 59(2-3), pp. 296-303.

Li, S.-H., Chen, J.-L., Yen, D. C. & Lin, Y.-H., 2013. Investigation on auditing principles and rules for PDM/PLM system implementation. *Computers in Industry,* 64(6), p. 741–753.

Liu, D. T. & Xu, X. W., 2001. A review of web-based product data management systems. *Computers in Industry,* 44(3), p. 251–262.

Liu, X., Heo, J. & Sha, L., 2005. Modeling 3-tiered Web applications. In: *Proceedings of the 13th IEEE International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS).* Atlanta, Georgia: IEEE Computer Society, pp. 307 - 310.

Liu, X., Heo, J. & Sha, L., 2005. *Modeling 3-Tiered Web Services,* Technical Report: UIUCDCS-R-2005-2580.

Li, W. D., Xia, K. & Gao, L., 2012. Sustainable information management for waste electrical and electronic equipment. In: L. Gao, et al. eds. *Proceedings of the 2012 IEEE 16th International Conference on Computer Supported Cooperative Work in Design.* Wuhan: IEEEXplore, pp. 875-881.

Lu, B. et al., 2014. Reusability based on life cycle sustainability assessment: case study on WEEE. *Procedia 21st CIRP Conference on Life Cycle Engineering,* Volume 15, pp. 472-478.

METI, 2000. *Act on the promotion of effective utilization of resources (Act No. 48 of 1991),* Japan: Ministry of Economy, Trade and Industry.

METI, 2015. *Law for the promotion of effective utilization of resources* , Japan: Recycling Promotion Division, Ministry of Economy, Trade and Industry (METI).

Milgram, P. & Kishino, F., 1994. A taxonomy of mixed reality virtual displays. *IEICE Transactions on Information and Systems,* E77-D(9), pp. 1321-1329.

Milgram, P., Takemura, H., Utsumi, A. & Kishino, F., 1994. Augmented reality: A clas of displays on the realityvirtuality continuum. *SPIE Proceedings: Telemanipulator and Telepresence Technologies,* Volume 2351, pp. 282-292.

Minami, T., Kanemaki, Y., Nemoto, H. & Tani, M., 2004. Eco & PLM project for environmental traceability of EEE. *Hitachi REview,* 53(5), pp. 243-249.

Morris, H., Lee, S., Shan, E. & Zeng, S., 2004. Information integration framework for product life-cycle management of diverse data. *Journal of Computing and Information Science in Engineering,* 4(4), pp. 352-358.

Neumann, U. & Majoros, A., 1998. Cognitive, performance, and systems issues for augmented reality applications in manufacturing and maintenance. In: *Proceedings of Virtual Reality Annual International Symposium (IEEE 1998).* Atlanta, Georgia: IEEE, pp. 4-11.

Noci, G., 1995. Accounting and non-accounting measures of quality-based performances in small firms. *International Journal of Operations & Production Management,* 15(7), pp. 78-105.

Oates, B. J., 2006. *Research Information Systems and Computing.* 1st ed. London; Thousand Oaks, California; New Delhi: SAGE Publications Ltd.

Ogungbuyi, O., Nnorom, I. C., Osibanjo, O. & Schluep, M., 2012. *E-Waste country assessment Nigeria,* s.l.: E-Waste Africa project of the Secretariat of the Basel Convention.

Peltonen, H., Pitkänen, O. & Sulonen, R., 1996. Process-based view of product data management. *Computers in Industry,* Volume 31, pp. 195-203.

Peng, T.-K. & Trappey, A. J. C., 1998. A step toward STEP-compatible engineering data management: the data models of product structure and engineering changes. *Robotics and Computer-Integrated Manufacturing,* 14(2), pp. 89-109.

Penny AB, 2015. *C Wear interactive glasses extended.* [Online] Available at: http://www.penny.se/extended.html [Accessed 11 April 2016].

Persson-Dahlqvist, A., 2005. *Product Data Management and Software Configuration Management Integration,* Västerås: Mälardalen University Press Licentiate Theses.

Piippo, P. et al., 2003. Intranet based system for a product innovation management process. *International Journal of Technology Management,* 25(6-7), pp. 631-642.

PLM Technology Guide, 2008. *PLM Systems.* [Online] Available at: http://plmtechnologyguide.com/site/?page_id=27 [Accessed 11 April 2016].

PLM Technology Guide, 2008. *PLM Vendors.* [Online] Available at: http://plmtechnologyguide.com/site/?page_id=12 [Accessed 11 April 2016].

Rachuri, S. et al., 2008. Information sharing and exchange in the context of product lifecycle management: Role of standards. *Computer-Aided Design,* 40(7), pp. 789-800.

Regenbrecht, H., G., B. & Wilke, W., 2005. Augmented reality projects in automotive and aerospace industry. *IEEE Computer Graphics and Applications,* 25(6), pp. 48-56.

Reiners, D., Stricker, D., Klinker, G. & Müller, S., 1999. Augmented reality for construction tasks: doorlock assembly. In: U. Behringer, G. Klinker & D. Mizell, eds. *IWAR '98 Proceedings of the international workshop on Augmented reality : placing artificial objects in real scenes: placing artificial objects in real scenes.* Natick, Massachusetts: A. K. Peters, Ltd., pp. 31-46.

Robertson, C. M., MacIntryre, B. & Walker, B., 2008. An evaluation of graphical context when the graphics are outside of the task areaq. In: *7th IEEE/ACM International Symposium on Mixed and Augmented Reality (ISMAR 2008).* Cambridge: IEEE Computer Society, pp. 73-76.

Salonen, T., Sääski, J., Marstio, I. & Rainio, K., 2009. Data pipeline from CAD to AR based assembly instructions. In: *Proceedings of the world Conference of Innovative Virtual reality (WINVR09).* Chaton-sur-Chaone: SME-AFM 2009, pp. 165-168.

Shiau, J. Y. & Li, X., 2007. A closed-loop design change control workflow in product data management for conceptual design phase. *International Journal of Intelligent Control and Systems,* 12(1), pp. 24-36.

Shimoda, H. et al., 2007. A feasibility study of decommissioning support method by augmented reality. *International Symposium on Symbiotic Nuclear Power Systems for 21st Century (ISSNP),* pp. 244-249.

Siemens PLM Software, 2006. *Teamcenter engineering integration toolkit (ITK).* United States: Siemens PLM Software Inc..

Siemens PLM Software, 2010. *Designing and manufacturing products to comply with international ELV, RoHS, REACH and WEEE guidelines,* United States: Siemens PLM Software Inc..

Siemens PLM Software, 2013. *Teamcenter 10.1 Client customization programmer's guide.* United States: Siemens PLM Software Inc.

Siemens PLM Software, 2013. *Teamcenter 10.1 Getting started with customization.* United States: Siemens PLM Software Inc.

Siemens PLM Software, 2013. *Teamcenter 10.1 My Teamcenter guide.* United States: Siemens PLM Software Inc..

Siemens PLM Software, 2013. *Teamcenter 10.1 Server customization guide.* United States: Siemens PLM Software Inc..

Siemens PLM Software, 2013. *Teamcenter 10.1 Services guide.* 1st ed. United States: Siemens PLM Software Inc..

Siemens PLM Software, 2016. *Teamcenter.* [Online] Available at: http://www.plm.automation.siemens.com/se_se/products/teamcenter/ [Accessed 15 May 2016].

Sims, D., 1994. New realities in aircraft design and manufacture. *IEEE Computer Graphics and Applications* , 14(2), p. 91.

Singh, V., 2008. *Teamcenter Middle Tier Overview,* s.l.: Siemens PLM Software.

Stackpole, B., 2003. There's a new app in town. *CIO MAgazine University of Michigan - PLM Development Consortium*, 15 May, 46(4), pp. 93-96.

Stark, J., 2011. *Product lifecycle management: 21st century paradigm for product realization.* 2nd ed. London: Springer-Verlag London.

Sulaiman, R., 2000. Change and delay. *Manufacturing Engineering,* 79(3), pp. 122-123.

Sung, C. & Park, S., 2007. A component-based product data management system. *The International Journal of Advanced Manufacturing Technology*, 33(5-6), pp. 614-626.

Sutherland, I., 1968. A head-mounted three-dimensional display. *Fall Joint Computer Conference, American Federation of Information Processing Societies (AFIPS),* Volume 33, pp. 757-764.

Swanson, E., 1994. Information systems innovation among organizations. *Management Science,* 40(9), p. 1069–1088.

Svensson, D., 2003. *Towards Product Structure Management in Hetrogeneous Environments,* Gothenburg: Chalmers University of Technology.

Syberfeldt, A., Danielsson, O., Holm, M. & Ekblom, T., 2014. Augmented reality at the industrial shop-floor. In: *Augmented and Virtual Reality. Volume 8853 of the series Lecture Notes in Computer Science.* Lecce: Springer, pp. 201-209.

Syberfeldt, A., Danielsson, O., Holm, M. & Wang, L., 2015. Visual assembling guidance using augmented reality. *Procedia manufacturing,* Volume 1, pp. 98-109.

Sääski, J. et al., 2008. Integration of design and assembly using augmented reality. *Micro Assembly Technologies and Applications,* Volume 260, pp. 395-404.

Tang, A., Owen, C., Biocca, F. & Mou, W., 2003. Comparative effectiveness of augmented reality in object assembly. In: *CHI '03 Proceedings of the SIGCHI Conference on Human Factors in Computing Systems.* Ft. Lauderdale, Florida: ACM New York, pp. 73-80.

Terzi, S. et al., 2010. Product lifecycle management - From its history to its new role. *International Journal of Product Lifecycle Management* , 4(4), pp. 360-389.

The Eclipse Foundation, 2016. *Eclipse.* [Online] Available at: https://eclipse.org [Accessed 13 May 2016].

Truesmart.com, 2016. *Cách kiểm tra iPad Air 2 chính hãng hay hàng dựng.* [Online] Available at: http://www.truesmart.com.vn/news/1209/cach-kiem-tra-ipad-air-2-chinh-hang-hay-hang-dung.html [Accessed 10 May 2016].

United States Environmental Protection Agency (EPA), 2016. *Recycling basics.* [Online] Available at: https://www.epa.gov/recycle/recycling-basics [Accessed 12 April 2016].

United States Environmental Protection Agency (EPA), 2016. *Sustainable Management of Electronics.* [Online] Available at: https://www.epa.gov/smm-electronics [Accessed 20 February 2016].

Webster, A. et al., 1996. Augmented reality in architectural construction, inspection and renovation. In: *Proceedings of ASCE Third Congress on Computing in Civil Engineering.* Anaheim, California: s.n., pp. 913-919.

Wiedenmaier, S., Oehme, O., Schmidt, L. & Luczak, H., 2003. Augmented reality for assembly processes design and experimental evaluation. *International Journal of Human-Computer Interaction,* 16(3), pp. 497-514.

Vila, C., Abellán-Nebot, J., Albiñana, J. & Hernández, G., 2015. An approach to sustainable product lifecycle management (Green PLM). *Procedia Engineering,* Volume 132, pp. 585-592.

Xu, X. W. & Liu, T., 2003. A web-enabled PDM system in a collaborative design environment. *Robotics and Computer-Integrated Manufacturing,* 19(4), pp. 315-328.

Yuan, M., Ong, S. & Nee, A., 2008. Assembly Guidance in Augmented Reality Environments Using a Virtual Interactive Tool. *International Journal of Production Research,* 46(7), pp. 1745-1767.

Zauner, J., Haller, M., Brandl, A. & Hartman, W., 2003. Authoring of a mixed reality assembly instructor for hierarchical structures. In: *Proceedings of the Second IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR ’03).* Tokio: IEEE Computer Society, pp. 237-246.

Zokai, S., Esteve, J., Genc, Y. & Navab, N., 2003. Multiview paraperspective projection model for diminished reality. In: *Proceedings of the Second IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR'03).* Tokio: IEEE Computer Society, pp. 217-226.

Appendix A: Categories of EEE

List of EEE categories covered by Directive 2012/19/EU along the transitional period provided for in Article 2(1)(a), according to Directive's Annex I and II (European Union, 2012):

1. Large household appliances
   Refrigerators, freezers, washing machines, cookers, dishwashers, microwaves, clothes dryers, etc.
2. Small household appliances
   Vacuum cleaners, toasters, fryers, electric knives, irons, carpet sweepers, clocks, grinders, etc.
3. IT and telecommunications equipment
   Computers, printer units, mainframes, calculators, typewriters, facsimile machines, telephones, telex, etc
4. Consumer equipment and photovoltaic panels
   Radio and television sets, video cameras, video and Hi-fi recorders, photovoltaic panels, etc.
5. Lighting equipment
   Straight and Compact fluorescent lamps, high intensity discharge lamps, low pressure sodium lamps, etc.
6. Electrical and electronic tools (with the exception of large-scale stationary industrial tools)
   Drills, saws, sewing machines, tools for mowing or other gardening activities, etc.
7. Toys, leisure and sports equipment
   Electric trains or car racing sets, hand held video game consoles, video games, coin slot machines, etc.
8. Medical devices
   Radiotherapy, cardiology and dialysis equipment, pulmonary ventilators, analysers, freezers, etc.
9. Monitoring and control instruments
   Smoke detectors, thermostats, heating regulators, etc.
10. Automatic dispensers
    Automatic dispensers for hot drinks, hot or cold bottles or cans, solid products, money, etc.

Appendix B: Properties of Hazardous Waste

List of hazardous properties covered by Directive 2008/98/EC (European Union, 2008):

- H 1 ‘Explosive’: substances and preparations which may explode under the effect of flame or which are more sensitive to shocks or friction than dinitrobenzene.
- H 2 ‘Oxidizing’: substances and preparations which exhibit highly exothermic reactions when in contact with other substances, particularly flammable substances.
- H 3-A ‘Highly flammable’
    - Liquid substances and preparations having a flash point below 21 °C (including extremely flammable liquids)
    - Substances and preparations which may become hot and finally catch fire in contact with air at ambient temperature without any application of energy.
    - Solid substances and preparations which may readily catch fire after brief contact with a source of ignition and which continue to burn or to be consumed after removal of the source of ignition.
    - Gaseous substances and preparations which are flammable in air at normal pressure
    - Substances and preparations which, in contact with water or damp air, evolve highly flammable gases in dangerous quantities.
- H 3-B ‘Flammable’: liquid substances and preparations having a flash point equal to or greater than 21 °C and less than or equal to 55 °C.
- H 4 ‘Irritant’: non-corrosive substances and preparations which, through immediate, prolonged or repeated contact with the skin or mucous membrane, can cause inflammation.
- H 5 ‘Harmful’: substances and preparations which, if they are inhaled or ingested or if they penetrate the skin, may involve limited health risks.
- H 6 ‘Toxic’: substances and preparations (including very toxic substances and preparations) which, if they are inhaled or ingested or if they penetrate the skin, may involve serious, acute or chronic health risks and even death.
- H 7 ‘Carcinogenic’: substances and preparations which, if they are inhaled or ingested or if they penetrate the skin, may induce cancer or increase its incidence.
- H 8 ‘Corrosive’: substances and preparations which may destroy living tissue on contact.
- H 9 ‘Infectious’: substances and preparations containing viable micro-organisms or their toxins which are known or reliably believed to cause disease in man or other living organisms.
- H 10 ‘Toxic for reproduction’: substances and preparations which, if they are inhaled or ingested or if they penetrate the skin, may induce non-hereditary congenital malformations or increase their incidence.
- H 11 ‘Mutagenic’: substances and preparations which, if they are inhaled or ingested or if they penetrate the skin, may induce hereditary genetic defects or increase their incidence.

- H 12 Waste which releases toxic or very toxic gases in contact with water, air or an acid.
- H 13 (*) ‘Sensitizing’: substances and preparations which, if they are inhaled or if they penetrate the skin, are capable of eliciting a reaction of hypersensitization such that on further exposure to the substance or preparation, characteristic adverse effects are produced.
- H 14 ‘Ecotoxic’: waste which presents or may present immediate or delayed risks for one or more sectors of the environment.
- H 15 Waste capable by any means, after disposal, of yielding another substance, e.g. a leachate, which possesses any of the characteristics listed above.

Notes

1. Attribution of the hazardous properties “toxic” (and “very toxic”), “harmful”, “corrosive”, “irritant”, “carcinogenic”, “toxic to reproduction”, “mutagenic” and “eco-toxic” is made on the basis of the criteria laid down by Annex VI, to Council Directive 67/548/EEC of 27 June 1967 on the approximation of laws, regulations and administrative provisions relating to the classification, packaging and labelling of dangerous substances (1).
2. Where relevant the limit values listed in Annex II and III to Directive 1999/45/EC of the European Parliament and of the Council of 31 May 1999 concerning the approximation of the laws, regulations and administrative provisions of the Member States relating to the classification, packaging and labelling of dangerous preparations (2) shall apply.

Test methods

The methods to be used are described in Annex V to Directive 67/548/EEC and in other relevant CEN-notes.

Appendix C: AR-TC MainCode.java

```
/*******************************************************************************/
/**                                                                           */
/** Copyright © 2016 Marcus Barstorp and Raquel Quesada                       */
/**                                                                           */
/** THE SOFTWARE IS PROVIDED “AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR  */
/** IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS */
/** FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR */
/** COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETER IN */
/** AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION */
/** WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.           */
/**                                                                           */
/*******************************************************************************/

package se.his.teamcenter.ar.main;

import java.awt.Dimension;
import java.awt.event.WindowEvent;
import java.awt.event.WindowListener;
import java.awt.image.BufferedImage;
import java.io.File;
import java.io.IOException;
import java.nio.file.Files;
import java.text.SimpleDateFormat;
import java.util.Arrays;
import java.util.List;
import java.util.UUID;

import javax.imageio.ImageIO;
import javax.swing.JFrame;

import se.his.teamcenter.ar.main.MainCode;
import se.his.teamcenter.ar.client.AppXSession;
import se.his.teamcenter.ar.main.WebcamMessagePanel;
import se.his.teamcenter.ar.main.WebcamViewPanel;
import se.his.teamcenter.ar.main.WebcamQrCodeScan.NewQrCodeIdentifiedListener;

import com.github.sarxos.webcam.Webcam;
import com.teamcenter.services.strong.core.DataManagementService;
import com.teamcenter.services.strong.core.SessionService;
import com.teamcenter.services.strong.core._2008_06.DataManagement.AttrInfo;
import
com.teamcenter.services.strong.core._2008_06.DataManagement.GetItemAndRelatedObjectsInfo
;
import
com.teamcenter.services.strong.core._2008_06.DataManagement.GetItemAndRelatedObjectsItem
Output;
import
com.teamcenter.services.strong.core._2008_06.DataManagement.GetItemAndRelatedObjectsResp
onse;
import com.teamcenter.services.strong.core._2008_06.DataManagement.RevisionOutput;
import com.teamcenter.soa.client.FileManagementUtility;
import com.teamcenter.soa.client.GetFileResponse;
import com.teamcenter.soa.client.model.ModelObject;
import com.teamcenter.soa.client.model.strong.Dataset;
```

```
import com.teamcenter.soa.client.model.strong.User;
import com.teamcenter.soa.common.ObjectPropertyPolicy;
import com.teamcenter.soa.exceptions.NotLoadedException;

/**************************************************************/
/** This is the main code for the AR-PLM integration interface */
/**************************************************************/
public class MainCode extends JFrame implements WindowListener,
NewQrCodeIdentifiedListener
{
   public static void main(String [] args) throws NotLoadedException, IOException
   {
      Arguments arguments = Arguments.Parse(args);
      new MainCode(arguments.host(), arguments.username(), arguments.password()); //
Declares arguments for Teamcenter (TC)
   }

   /**********************************************************/
   /** Calls and initialises createTeamcenterSession function */
   /**                                                       */
   /** @param teamcenterHost                                 */
   /** @param username                                       */
   /** @param password                                       */
   /**********************************************************/

   public MainCode(String teamcenterHost, String username, String password)
   {
      super();
      createTeamcenterSession(teamcenterHost, username, password);
      initialize();

      /** Show webcam window */

      this.setVisible(true); // Sets the first window
      secondWindow.setVisible(true); // Sets a second window
   }

   /*************************************************************************/
   /** Create a session in Teamcenter and print on screen the host and the user */
   /** The data for the log in is kept as argument in the project properties    */
   /**                                                                         */
   /** @param host                                                             */
   /** @param username                                                         */
   /** @param password                                                         */
   /*************************************************************************/

   private void createTeamcenterSession(String host, String username, String password)
   {
      System.out.println("Starting a session on " + host); // prints on screen the host
for the session
      this.session = new AppXSession (host);

      System.out.println("Logging in with username " + username); // prints on screen the
session's user
      this.user = this.session.login(username, password); // uses username and password
to log into Teamcenter

      try
      {
```

```
            System.out.println("logged in as user " + this.user.get_user_name());
        }
        catch (NotLoadedException e)
        {
            throw new RuntimeException(e);
        }

        /** Get the session service and set the object property policy in Teamcenter */

        SessionService sessionService =
SessionService.getService(AppXSession.getConnection());
        ObjectPropertyPolicy oppolicy = new ObjectPropertyPolicy();
        String path = MainCode.class.getResource("/oppolicy.xml").toString();
        try
        {
            oppolicy.load(path);
        }
        catch (IOException e)
        {
            throw new RuntimeException(e);
        }

        sessionService.setObjectPropertyPolicy(oppolicy);
        this.fileManager = new FileManagementUtility(AppXSession.getConnection()); // Get
and cache the file manager

    }

    /*****************************************************************************/
    /** Webcam configuration: initialization, dimensions, variable declaration, */
    /** function calling, windows and overlay text and image                    */
    /*****************************************************************************/

    /** Set the preferred custom dimension for the webcam image (this is hard coded for a
particular webcam) */

    public static final Dimension HD_RESOLUTION = new Dimension (1280, 720);
    private static final long serialVersionUID =1L;

    /** Variable declaration and function calling */

    private AppXSession session;
    private User user;
    private FileManagementUtility fileManager;
    private Webcam webcam;

    private Thread qrScannerThread;
    private WebcamQrCodeScan qrScanner; //Calls the function WebcamQrCodeScan

    /** First window */

    private WebcamViewPanel webcamViewPanelFirstWindow; // Deactivate this line for a
black background
// private WebcamMessagePanel webcamMessagePanelFirstWindow; // Activate this line for a
black background

    /** Second window */

    private JFrame secondWindow;
```

```
// private WebcamViewPanel webcamViewPanelSecondWindow; // Activate this line for a black background
   private WebcamMessagePanel webcamMessagePanelSecondWindow; // Deactivate these lines for a black background

   /**********************************************************************/
   /** Initialise the webcam, create windows and overlay text and images */
   /**********************************************************************/

   private void initialize()
   {
      this.setTitle("Webcam window");
      this.setResizable(true);
      this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

      this.addWindowListener(this); // Register as a window listener

      /** Setup webcam and QR scanner */

      this.webcam = Webcam.getDefault();
      this.webcam.setCustomViewSizes(new Dimension[] { HD_RESOLUTION});
      this.webcam.setViewSize(HD_RESOLUTION);

      this.qrScanner = new WebcamQrCodeScan(this.webcam); // Setup the webcam QR code scanner
      this.qrScanner.addNewQrCodeIdentifiedListener(this); // Register as a listener to QR code identification events
      this.qrScannerThread = new Thread(this.qrScanner); // Create (but do not start) the thread the scanner will run on

      /** Create and add the overlay webcam panel. It renders the webcam video feed + an overlay of text and images */

      this.webcamViewPanelFirstWindow = new WebcamViewPanel(this.webcam);
      this.webcamViewPanelFirstWindow.setPreferredSize(HD_RESOLUTION);
      this.webcamViewPanelFirstWindow.setFitArea(true);
      this.add( this.webcamViewPanelFirstWindow );

/*****************************************************************************/
/**                                                                         */
/** @modified from original by Raquel Quesada                               */
/**                                                                         */
/*****************************************************************************/

      /** Create and add the overlay webcam panel. It renders a black background window + an overlay of text and images */

//    this.webcam.open();
//    this.webcamMessagePanelFirstWindow = new MessagePanel();
//    this.webcamMessagePanelFirstWindow.setPreferredSize(HD_RESOLUTION);
//    this.add( this.webcamMessagePanelFirstWindow );

      /** Create the second window */

      secondWindow = new JFrame();
      secondWindow.setTitle("Webcam window");
      secondWindow.setResizable(true);
      secondWindow.setDefaultCloseOperation(JFrame.DO_NOTHING_ON_CLOSE);
```

```
        /** Create and add the overlay webcam panel. It renders the webcam video feed + an
    overlay of text and images */

    //    this.webcamViewPanelSecondWindow= new WebcamPanel(this.webcam);
    //    this.webcamViewPanelSecondWindow.setPreferredSize(HD_RESOLUTION);
    //    this.webcamViewPanelSecondWindow.setFitArea(true);
    //    secondWindow.add( this.webcamPanelSecondWindow );

    /*********************************************************************************/

        /** Create and add the overlay webcam panel. It renders a black background window +
    an overlay of text and images */

        this.webcam.open();
        this.webcamMessagePanelSecondWindow = new WebcamMessagePanel();
        this.webcamMessagePanelSecondWindow.setPreferredSize(HD_RESOLUTION);
        secondWindow.add(this.webcamMessagePanelSecondWindow);

        this.pack();
    }

    private UUID clientId = UUID.randomUUID();

    @Override
    public void windowOpened(WindowEvent we)
    {
        System.out.println("Starting the QR scanner thread");
        qrScannerThread.start();
    }

    @Override
    public void windowClosing(WindowEvent we)
    {
        System.out.println("Stopping the QR scanner");
        this.qrScanner.stop();
        this.qrScannerThread.interrupt();
        try
        {
            this.qrScannerThread.join(); // Wait for the QR thread to stop
        }
        catch (InterruptedException ie)
        {
            ie.printStackTrace();
        }

        System.out.println("Terminating the file manager");
        this.fileManager.term();

        System.out.println("Logging out");
        this.session.logout();
    }

    @Override
    public void windowClosed(WindowEvent we) { /* NOT USED */ }

    @Override
    public void windowIconified(WindowEvent we) {/* NOT USED */}

    @Override
```

```
    public void windowDeiconified(WindowEvent we) {/* NOT USED */}

    @Override

    public void windowActivated(WindowEvent we) {/* NOT USED */}

    @Override
    public void windowDeactivated(WindowEvent we) {/* NOT USED */}

/*****************************************************************************/
/**                                                                         **/
/** @modified from original by Raquel Quesada                               **/
/**                                                                         **/
/*****************************************************************************/

    /*************************************************************************/
    /** Refreshes the screen, the text and the image when detects a new QR code */
    /*************************************************************************/

    @Override
    public void newQrCodeIdentified(String contents) {
       System.out.println("New QR code detected\nContents:\n" + contents);
       try
       {
          List<String> messages = getInstructions(contents);
          BufferedImage image = getImage(contents);

          /** Webcam panel */
          this.webcamViewPanelFirstWindow.replaceMessages(messages);
          this.webcamViewPanelFirstWindow.replaceImage(image);
//        this.webcamViewPanelSecondWindow.replaceMessages(messages);
//        this.webcamViewPanelSecondWindow.replaceImage(image);

          /** Black window */
//        this.webcamMessagePanelFirstWindow.replaceMessages(messages);
//        this.webcamMessagePanelFirstWindow.replaceImage(image);
          this.webcamMessagePanelSecondWindow.replaceMessages(messages);
          this.webcamMessagePanelSecondWindow.replaceImage(image);
       }
       catch (Exception e) {
          String errorMessage = String.format("Error: %s", e.getMessage());
          System.out.println(errorMessage);

          /** Webcam panel */
          this.webcamViewPanelFirstWindow.replaceMessages( Arrays.asList(errorMessage) );
//        this.webcamViewPanelSecondWindow.replaceMessages(Arrays.asList(errorMessage) );

          /** Black window */
//        this.webcamMessagePanelFirstWindow.replaceMessages( Arrays.asList(errorMessage)
);
          this.webcamMessagePanelSecondWindow.replaceMessages( Arrays.asList(errorMessage)
);
       }

/*****************************************************************************/

    }

   /***************************************************************************/
```

```
   /** Download and read text file                                              */
   /** It is assumed that only one file is downloaded, so reads the file lines into    */
   /** memory and returns them                                                  */
   /*****************************************************************************/

   private List<String> getInstructions(String id) throws Exception
   {
      return Files.readAllLines(getFileFromId(id,
"recycling_instructions.txt").toPath());
   }

/*****************************************************************************/
/**                                                                          */
/** @author Raquel Quesada                                                   */
/**                                                                          */
/*****************************************************************************/

   /***************************************/
   /** Download and read image file         */
   /** It returns an image only if it exits */
   /***************************************/

   private BufferedImage getImage(String id) throws Exception
   {
      File file =getFileFromId(id, "recycling_instructions.png");
      if (file == null)
         return null;
      return ImageIO.read(file);
   }

/*****************************************************************************/

   /*****************************************************************************/
   /** Reads the ID string previously obtained from the QR code and sends the query for*/
    /**the text and image files                                                */
   /*****************************************************************************/

   public File getFileFromId(String id, String filename) throws Exception
   {
      DataManagementService dmService =
DataManagementService.getService(AppXSession.getConnection()); // Get the data
management service

      /** Prepare the search criteria */
      GetItemAndRelatedObjectsInfo[] infos = new GetItemAndRelatedObjectsInfo[1];
      infos[0] = new GetItemAndRelatedObjectsInfo();

      /** To be able to call the service method, a client ID seems to be needed (it does
not have to be a UUID) */
      infos[0].clientId = clientId.toString();
      infos[0].itemInfo.ids = new AttrInfo[]{new AttrInfo()};

      String[] idParts = id.split("/");

      /** Get items associated with the ID stored in the QR code */
      infos[0].itemInfo.ids[0] = new AttrInfo();
      infos[0].itemInfo.ids[0].name = "item_id";
      infos[0].itemInfo.ids[0].value = idParts[0];
      infos[0].itemInfo.useIdFirst = true;
```

```
      infos[0].revInfo.processing = "All"; // Process/retrieve all revisions associated
to a found item
      infos[0].datasetInfo.filter.processing = "All"; // Process/retrieve all data sets
associated to a found item

      /** This filter only retrieves data sets with this particular name */
      /** Note: It does not apply to the name of the actual file contained in the data
set */
      infos[0].datasetInfo.filter.name = filename; // recycling_instructions.txt and
recycling_instructions.png

      /**********************/
      /** Call the services */
      /**********************/

      GetItemAndRelatedObjectsResponse response =
dmService.getItemAndRelatedObjects(infos);

      if (response.output.length == 0)
        throw new Exception("No item with the ID " + id + " was found");
      else if (response.output.length > 1)
        throw new Exception("More than one item with the ID " + id + " was found");

      GetItemAndRelatedObjectsItemOutput itemOutput = response.output[0];

      /** Print some debug info */
      String name = itemOutput.item.get_object_name();
      System.out.println("Item name: " + name);
      System.out.println("Item object string: " + itemOutput.item.get_object_string());
      System.out.println("Item ID: " + itemOutput.item.get_item_id());
      System.out.println("Object type: " + itemOutput.item.get_object_type());
      System.out.println("Object string: " + itemOutput.item.get_object_string());

/*******************************************************************************/
/**                                                                           */
/** @author Raquel Quesada                                                    */
/**                                                                           */
/*******************************************************************************/

      /** Note: The revisions in the revision array are stored in descending order (the
latest revision first) */
      RevisionOutput revOutput = null;
      if( idParts.length > 1)
      {
        for( RevisionOutput revision : itemOutput.itemRevOutput )
        {
          /*********************************************/
          /** Obtain the file through the revision number */
          /**                                          */
          /** 001111/2016-04-11#17:06                  */
          /*********************************************/

          // idParts[1] is "A", "B", "C".
          if( idParts[1].equals( revision.itemRevision.get_item_revision_id() ))
          {
            revOutput = revision;
            break;
          }
```

```
            /*****************************************************/
            /** Obtain the file through the last modification date */
            /**                                                   */
            /** 001111/2016-04-11#17:06                           */
            /*****************************************************/

            SimpleDateFormat formatter = new SimpleDateFormat("YYYY-MM-dd#HH:mm");
            String creationDate = formatter.format(
revision.itemRevision.get_last_mod_date().getTime() );

            System.out.println("CREATION " + creationDate + ", " + idParts[0]);

            // idParts[1] is "2016-03-11#22:10"
            if(idParts[1].equals(creationDate)) {
               // Found it!
               revOutput = revision;
            }

         }

         if( revOutput == null)
         {
            throw new RuntimeException("A revision with the ID "+idParts[1]+" was not
found for the item with ID "+idParts[0]);
         }
      }
      else
      {
         revOutput = itemOutput.itemRevOutput[0];
      }

/*****************************************************************************************/

      /** Print some debug info */
      System.out.println("Latest revision name: "
            + revOutput.itemRevision.get_object_name());
      System.out.println("Latest revision object string: "
            + revOutput.itemRevision.get_object_string());
      System.out.println("Latest revision item ID: "
            + revOutput.itemRevision.get_item_revision_id());
      System.out.println("Latest revision item revision ID: "
            + revOutput.itemRevision.get_item_revision_id());

      /** It is assumed that only one item corresponds with the ID and revision */

      if (revOutput.datasetOutput.length == 0)
      {
         return null;
      }
      else if(revOutput.datasetOutput.length > 1)
      {
         throw new Exception("The item revision has ambiguous data");
      }

      Dataset dataset = revOutput.datasetOutput[0].dataset;

      /** Print some debug info */
      System.out.println("Dataset: " + dataset.get_object_name());
```

```
        for (String ref : dataset.get_ref_names())
        {
           System.out.println("Dataset reference: " + ref);
        }

        /** Assume there is only one reference contained in the data set (this is the
actual file) */
        ModelObject[] references = new ModelObject[] { dataset.get_ref_list()[0] };
        GetFileResponse fresp = this.fileManager.getFiles(references);

        /** Print some debug info */
        System.out.format("Downloaded %d of %d files with %d partial errors.\n",
fresp.sizeOfFiles(), references.length, fresp.sizeOfPartialErrors());
        System.out.println("Amount of files "+fresp.getFiles().length);
        System.out.println("File "+fresp.getFile(0));
        System.out.println("Files as path "+fresp.getFile(0).toPath());

        return fresp.getFile(0);
        }
}
```

Appendix D: AR-TC Arguments.java

```
/*******************************************************************************/
/**                                                                           */
/** Copyright © 2016 Marcus Barstorp                                          */
/**                                                                           */
/** THE SOFTWARE IS PROVIDED “AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR  */
/** IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS  */
/** FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR    */
/** COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETER IN  */
/** AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION   */
/** WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.            */
/**                                                                           */
/*******************************************************************************/

package se.his.teamcenter.ar.main;

import java.util.ArrayList;
import java.util.LinkedHashMap;
import java.util.List;
import java.util.Map;
import java.util.Scanner;

public class Arguments {

   private Map<String, String> options = new LinkedHashMap<>();

   public String host(){ return options.get("host"); }
   public String username(){ return options.get("username"); }
   public String password(){ return options.get("password"); }

   public static Arguments Parse(String[] args)
   {
      Arguments arguments = new Arguments();

      int current = 0;
      while( current < args.length )
      {
         String argument = args[current++];

         switch(argument)
         {
            case "-h": case "--host":
               if( current == args.length ) throw new RuntimeException("host was supplied
without a value");
               arguments.storeOption("host", args[current++]);
               break;

            case "-u": case "--username":
               if( current == args.length ) throw new RuntimeException("username was
supplied without a value");
               arguments.storeOption("username", args[current++]);
               break;
            case "-p": case "--password":
               if( current == args.length ) throw new RuntimeException("password was
supplied without a value");
```

```
            arguments.storeOption("password", args[current++]);
            break;
         default:
            throw new RuntimeException("unknown argument "+argument);
      }
   }

   List<String> mandatoryOptions = new ArrayList<>();
   if ( !arguments.options.containsKey("host") )
   {
      mandatoryOptions.add("host");
   }

   if ( !arguments.options.containsKey("username") )
   {
      mandatoryOptions.add("username");
   }

   if ( !arguments.options.containsKey("password") )
   {
      mandatoryOptions.add("password");
   }

   @SuppressWarnings("resource")
   Scanner scanner = new Scanner(System.in);

   for( String key : mandatoryOptions )
   {
      String value = System.getProperty(key, "");
      if( !value.isEmpty() )
      {
         arguments.storeOption(key,value);
         System.out.format("%s: using system property (%s)\n", key, value);
         continue;
      }

      while(!arguments.options.containsKey(key))
      {
         System.out.format("%s: ", key);
         value = scanner.nextLine();

         if( value.isEmpty() )
         {
            System.out.println("Invalid argument entered. Try again.");
         }
         else
         {
            arguments.storeOption(key,value);
            break;
         }
      }
   }

   return arguments;
}

public void storeOption(final String key, final String value)
{
   if( value == null || value.isEmpty() )
```

```
        {
           throw new RuntimeException("The option "+key+" cannot be empty");
        }
        this.options.put(key, value);
    }
}
```

Appendix E: AR-TC AppXSession.java

```
/*==========================================================================*/
/**                                                                        */
/** Copyright © 2012 Siemens Product Lifecycle Management Software Inc. */
/** All Rights Reserved.                                                   */
/**                                                                        */
/*==========================================================================*/
```

Due to Siemens copyright restrictions this code cannot be publish in a public report but can be obtained from Siemens PLM Software.
For further information:
https://www.plm.automation.siemens.com/en_us/products/teamcenter/

Description:

- Creates an instance of the connection when a Service stub is instantinated.
- Creates an instance of the session with a connection to a specific Server.
- Calls the Credential Manager function (AppXCredentialManager.java).

Appendix F: AR-TC AppXCredentialManager.java

```
/*********************************************************************/
/**                                                                 */
/** Copyright © 2012 Siemens Product Lifecycle Management Software Inc. */
/** All Rights Reserved.                                            */
/**                                                                 */
/*********************************************************************/
```

Due to Siemens copyright restrictions this code cannot be publish in a public report but can be obtained from Siemens PLM Software.
For further information:
https://www.plm.automation.siemens.com/en_us/products/teamcenter/

Description:

- The CredentialManager is used by the Teamcenter Services framework to get the server's credentials when challenged by the server. This can occur after a period of inactivity and the server has timed-out the user's session, at which time the client application will need to re-authenticate. The framework will call one of the getCredentials methods (depending on circumstances) and will send the sessionService.login service request. Upon successful completion of the login service request. The last service request (one that caused the challenge) will be resent.
- The framework will also call the setUserPassword setGroupRole methods whenever these credentials change, thus allowing this implementation of the CredentialManager to cache these values so prompting of the user is not required for re-authentication.

Appendix G: AR-TC WebcamQrCodeScan.java

```
/*****************************************************************************/
/**                                                                         **/
/** Copyright © 2016 Raquel Quesada                                         **/
/**                                                                         **/
/** THE SOFTWARE IS PROVIDED “AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR **/
/** IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS **/
/** FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR **/
/** COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETER IN **/
/** AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION **/
/** WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.         **/
/**                                                                         **/
/*****************************************************************************/

package se.his.teamcenter.ar.main;

import java.awt.image.BufferedImage;
import java.util.Set;
import java.util.concurrent.CopyOnWriteArraySet;

import com.github.sarxos.webcam.Webcam;
import com.google.zxing.BinaryBitmap;
import com.google.zxing.LuminanceSource;
import com.google.zxing.MultiFormatReader;
import com.google.zxing.NotFoundException;
import com.google.zxing.client.j2se.BufferedImageLuminanceSource;
import com.google.zxing.common.HybridBinarizer;

public class WebcamQrCodeScan implements Runnable
{
   private final Webcam webcam;
   private MultiFormatReader reader = new MultiFormatReader();
   private String oldContent = null;
   private Set<NewQrCodeIdentifiedListener> newQrCodeIdentifiedListeners = new
CopyOnWriteArraySet<>();
   private boolean continueProcessing = true;

   public interface NewQrCodeIdentifiedListener
   {
      public void newQrCodeIdentified(final String contents);
   }

   public void addNewQrCodeIdentifiedListener( NewQrCodeIdentifiedListener listener )
   {
      newQrCodeIdentifiedListeners.add(listener);
   }

   public void removeNewQrCodeIdentifiedListener( NewQrCodeIdentifiedListener listener )
   {
      newQrCodeIdentifiedListeners.remove(listener);
   }

   public WebcamQrCodeScan( Webcam webcam )
   {
      this.webcam = webcam;
   }
```

```
public void run()
{
   while( continueProcessing )
   {
      try
      {
         // Sleep for a little while (no need to process each frame from the webcam)
         Thread.sleep(100);
      }
      catch (InterruptedException e)
      {
         // If the thread was interrupted, return immediately
         return;
      }

      if( this.webcam.isOpen() )
      {
         BufferedImage image = webcam.getImage();
         if(image == null)
         {
            continue;
         }

         final String newContent = this.scan(image);
         if( newContent != null )
         {
            if( !newContent.equals( oldContent ))
            {
               oldContent = newContent;
               newQrCodeIdentified(newContent);
            }
         }
      }
   }
}

public void stop()
{
   this.continueProcessing = false;
}

private String scan(BufferedImage image)
{
   LuminanceSource source = new BufferedImageLuminanceSource(image);
   BinaryBitmap bitmap = new BinaryBitmap(new HybridBinarizer(source));

   try {
      return reader.decode(bitmap).getText();
   } catch (NotFoundException e) {
      return null;
   }
}

private void newQrCodeIdentified(final String content)
{
   for( NewQrCodeIdentifiedListener listener : newQrCodeIdentifiedListeners )
   {
      listener.newQrCodeIdentified(content);
```

```
            }
        }
    }
```

Appendix H: AR-TC WebcamViewPanel.java

```
/*************************************************************************************/
/**                                                                                 */
/** Copyright © 2012 - 2015 Bartosz Firyn and Contributors                          */
/**                                                                                 */
/** Modified by Raquel Quesada and Marcus Barstorp                                  */
/**                                                                                 */
/** Permission is hereby granted, free of charge, to any person obtaining a copy of */
/** this software and associated documentation files (“the “Software”), to deal in  */
/** in the Software without restriction, including without limitation the rights to */
/** use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of */
/** the Software, and to permit persons to whom the Software is furnished to do so, */
/** subject to the following conditions:                                            */
/** The above copyright notice and this permission notice shall be included in all  */
/** copies or substantial portions of the Software.                                 */
/**                                                                                 */
/** THE SOFTWARE IS PROVIDED “AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR      */
/** IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS */
/** FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR  */
/** COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETER IN */
/** AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION */
/** WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.                 */
/**                                                                                 */
/** https://github.com/sarxos/webcam-capture                                       */
/** https://github.com/sarxos/webcam-capture/blob/master/webcam-
    capture/src/main/java/com/github/sarxos/webcam/WebcamPanel.java                */
/**                                                                                 */
/*************************************************************************************/

package se.his.teamcenter.ar.main;

import static java.awt.RenderingHints.KEY_ANTIALIASING;
import static java.awt.RenderingHints.KEY_INTERPOLATION;
import static java.awt.RenderingHints.KEY_RENDERING;
import static java.awt.RenderingHints.VALUE_ANTIALIAS_OFF;
import static java.awt.RenderingHints.VALUE_ANTIALIAS_ON;
import static java.awt.RenderingHints.VALUE_INTERPOLATION_BILINEAR;
import static java.awt.RenderingHints.VALUE_RENDER_SPEED;

import java.awt.AlphaComposite;
import java.awt.BasicStroke;
import java.awt.Color;
import java.awt.Dimension;
import java.awt.Font;
import java.awt.FontMetrics;
import java.awt.Graphics;
import java.awt.Graphics2D;
import java.awt.GraphicsConfiguration;
import java.awt.GraphicsEnvironment;
import java.awt.RenderingHints;
import java.awt.image.BufferedImage;
import java.beans.PropertyChangeEvent;
import java.beans.PropertyChangeListener;
import java.util.Collection;
import java.util.HashMap;
import java.util.List;
```

```
import java.util.Map;
import java.util.concurrent.CopyOnWriteArrayList;
import java.util.concurrent.Executors;
import java.util.concurrent.RejectedExecutionException;
import java.util.concurrent.ScheduledExecutorService;
import java.util.concurrent.ThreadFactory;
import java.util.concurrent.TimeUnit;
import java.util.concurrent.atomic.AtomicBoolean;
import java.util.concurrent.atomic.AtomicInteger;

import javax.swing.JPanel;
import javax.swing.SwingUtilities;

import org.slf4j.Logger;
import org.slf4j.LoggerFactory;

import com.github.sarxos.webcam.Webcam;
import com.github.sarxos.webcam.WebcamEvent;
import com.github.sarxos.webcam.WebcamException;
import com.github.sarxos.webcam.WebcamExceptionHandler;
import com.github.sarxos.webcam.WebcamListener;

public class WebcamViewPanel extends JPanel implements WebcamListener,
PropertyChangeListener
{

   private static final long serialVersionUID = 1L;
   private static final Logger LOG = LoggerFactory.getLogger(WebcamViewPanel.class);

   /** Minimum FPS frequency. */
   public static final double MIN_FREQUENCY = 0.016; // 1 frame per minute

   /** Maximum FPS frequency. */
   private static final double MAX_FREQUENCY = 50; // 50 frames per second

   /** Thread factory used by execution service. */
   private static final ThreadFactory THREAD_FACTORY = new PanelThreadFactory();
   public static final Map<RenderingHints.Key, Object> DEFAULT_IMAGE_RENDERING_HINTS =
new HashMap<RenderingHints.Key, Object>();
   static
   {
      DEFAULT_IMAGE_RENDERING_HINTS.put(KEY_INTERPOLATION, VALUE_INTERPOLATION_BILINEAR);
      DEFAULT_IMAGE_RENDERING_HINTS.put(KEY_RENDERING, VALUE_RENDER_SPEED);
      DEFAULT_IMAGE_RENDERING_HINTS.put(KEY_ANTIALIASING, VALUE_ANTIALIAS_ON);
   }

   /** This runnable will do nothing more than repaint panel. */
   private final Runnable repaint = new SwingRepainter(this);

   /** Rendering hints to be used when painting image to be displayed. */
   private Map<RenderingHints.Key, Object> imageRenderingHints = new
HashMap<RenderingHints.Key, Object>(DEFAULT_IMAGE_RENDERING_HINTS);

   /** Scheduled executor acting as timer. */
   private ScheduledExecutorService executor = null;

   /**
    * The mode of how the image will be resized to fit into panel bounds. Default is
{@link DrawMode#FIT}
```

```
     *
     * @see DrawMode
     */
    private DrawMode drawMode = DrawMode.FIT;

    /** Frames requesting frequency. */
    private double frequency = 5; // FPS

    /**
     * Is frames requesting frequency limited? If true, images will be fetched in
configured time
     * intervals. If false, images will be fetched as fast as camera can serve them.
     */
    private boolean frequencyLimit = false;

    /** Display FPS? */
    private boolean frequencyDisplayed = false;

    /** Display image size? */
    private boolean imageSizeDisplayed = false;
    private boolean antialiasingEnabled = true;
    private final Webcam webcam;

    /** Repainter is used to fetch images from camera and force panel repaint when image
is ready. */
    private final ImageUpdater updater;

    /** Image currently being displayed. */
    private BufferedImage image = null;

/**********************************************************************************/
/**                                                                              */
/** @author Raquel Quesada                                                       */
/**                                                                              */
/**********************************************************************************/

    /** Image retrieved from Teamcenter. */
    private BufferedImage pngimage = null;

/**********************************************************************************/

    /** Webcam is currently starting. */
    private volatile boolean starting = false;

    /** Painting is paused. */
    private volatile boolean paused = false;

    /** Is there any problem with webcam? */
    private volatile boolean errored = false;

    /** Webcam has been started. */
    private final AtomicBoolean started = new AtomicBoolean(false);

    private final Painter defaultPainter = new DefaultPainter();

    /**
     * Painter used to draw image in panel.
     *
     * @see #setPainter(Painter)
```

```
     * @see #getPainter()
     */
    private Painter painter = defaultPainter;

    /** Preferred panel size. */
    private Dimension defaultSize = null;

    /** If debug info should be displayed. */
    private boolean displayDebugInfo = false;

    /**  Is image mirrored? */
    private boolean mirrored = false;

    private List<String> messages = new CopyOnWriteArrayList<>();
    /**
     * This enum is to control of how image will be drawn in the panel bounds.
     *
     * @author Sylwia Kauczor
     */
    public static enum DrawMode {

        /**
         * Do not resize image - paint it as it is. This will make the image to go off out
the
         * bounds if panel is smaller than image size.
         */
        NONE,

        /**
         * Will resize image to the panel bounds. This mode does not care of the image
scale, so the
         * final image may be disrupted.
         */
        FILL,

        /**
         * Will fit image into the panel bounds. This will resize the image and keep both x
and y
         * scale factor.
         */
        FIT,
    }

    /**
     * Interface of the painter used to draw image in panel.
     *
     * @author Bartosz Firyn (SarXos)
     */
    public static interface Painter {

        /**
         * Paint panel without image.
         *
         * @param panel the webcam panel to paint on
         * @param g2 the graphics 2D object used for drawing
         */
        void paintPanel(WebcamViewPanel panel, Graphics2D g2);

        /**
```

```
         * Paint webcam image in panel.
         *
         * @param panel the webcam panel to paint on
         * @param image the image from webcam
         * @param g2 the graphics 2D object used for drawing
         */
        void paintImage(WebcamViewPanel panel, BufferedImage image, Graphics2D g2);
    }

    /**
     * Default painter used to draw image in panel.
     *
     * @author Bartosz Firyn (SarXos)
     * @author Sylwia Kauczor
     */
    public class DefaultPainter implements Painter {

        /**
         * Webcam device name.
         */
        private String name = null;

        /**
         * Last repaint time, used for debug purpose.
         */
        private long lastRepaintTime = -1;

        /**
         * Buffered image resized to fit into panel drawing area.
         */
        private BufferedImage resizedImage = null;

        @Override
        public void paintPanel(WebcamViewPanel owner, Graphics2D g2) {

            assert owner != null;
            assert g2 != null;

            Object antialiasing = g2.getRenderingHint(KEY_ANTIALIASING);

            g2.setRenderingHint(KEY_ANTIALIASING, isAntialiasingEnabled() ?
VALUE_ANTIALIAS_ON : VALUE_ANTIALIAS_OFF);
            g2.setBackground(Color.BLACK);
            g2.fillRect(0, 0, getWidth(), getHeight());

            int cx = (getWidth() - 70) / 2;
            int cy = (getHeight() - 40) / 2;

            g2.setStroke(new BasicStroke(2));
            g2.setColor(Color.LIGHT_GRAY);
            g2.fillRoundRect(cx, cy, 70, 40, 10, 10);
            g2.setColor(Color.WHITE);
            g2.fillOval(cx + 5, cy + 5, 30, 30);
            g2.setColor(Color.LIGHT_GRAY);
            g2.fillOval(cx + 10, cy + 10, 20, 20);
            g2.setColor(Color.WHITE);
            g2.fillOval(cx + 12, cy + 12, 16, 16);
            g2.fillRoundRect(cx + 50, cy + 5, 15, 10, 5, 5);
            g2.fillRect(cx + 63, cy + 25, 7, 2);
```

```
      g2.fillRect(cx + 63, cy + 28, 7, 2);
      g2.fillRect(cx + 63, cy + 31, 7, 2);

      g2.setColor(Color.DARK_GRAY);
      g2.setStroke(new BasicStroke(3));
      g2.drawLine(0, 0, getWidth(), getHeight());
      g2.drawLine(0, getHeight(), getWidth(), 0);

      String str = null;

      final String strInitDevice = "Initializing device";
      final String strNoImage = "No image";
      final String strDeviceError = "Device error";

      if (!errored) {
         str = starting ? strInitDevice : strNoImage;
      } else {
         str = strDeviceError;
      }

      FontMetrics metrics = g2.getFontMetrics(getFont());
      int w = metrics.stringWidth(str);
      int h = metrics.getHeight();

      int x = (getWidth() - w) / 2;
      int y = cy - h;

      g2.setFont(getFont());
      g2.setColor(Color.WHITE);
      g2.drawString(str, x, y);

      if (name == null) {
         name = webcam.getName();
      }

      str = name;

      w = metrics.stringWidth(str);
      h = metrics.getHeight();

      g2.drawString(str, (getWidth() - w) / 2, cy - 2 * h);
      g2.setRenderingHint(KEY_ANTIALIASING, antialiasing);
   }

   @Override
   public void paintImage(WebcamViewPanel owner, BufferedImage image, Graphics2D g2) {

      assert owner != null;
      assert image != null;
      assert g2 != null;

      int pw = getWidth();
      int ph = getHeight();
      int iw = image.getWidth();
      int ih = image.getHeight();

      Object antialiasing = g2.getRenderingHint(KEY_ANTIALIASING);
      Object rendering = g2.getRenderingHint(KEY_RENDERING);
```

```
            g2.setRenderingHint(KEY_ANTIALIASING, VALUE_ANTIALIAS_OFF);
            g2.setRenderingHint(KEY_RENDERING, VALUE_RENDER_SPEED);
            g2.setBackground(Color.BLACK);
            g2.setColor(Color.BLACK);
            g2.fillRect(0, 0, pw, ph);

            // resized image position and size
            int x = 0;
            int y = 0;
            int w = 0;
            int h = 0;

            switch (drawMode) {
               case NONE:
                  w = image.getWidth();
                  h = image.getHeight();
                  break;
               case FILL:
                  w = pw;
                  h = ph;
                  break;
               case FIT:
                  double s = Math.max((double) iw / pw, (double) ih / ph);
                  double niw = iw / s;
                  double nih = ih / s;
                  double dx = (pw - niw) / 2;
                  double dy = (ph - nih) / 2;
                  w = (int) niw;
                  h = (int) nih;
                  x = (int) dx;
                  y = (int) dy;
                  break;
            }

            if (resizedImage != null) {
               resizedImage.flush();
            }

            if (w == image.getWidth() && h == image.getHeight() && !mirrored) {
               resizedImage = image;
            } else {

               GraphicsEnvironment genv = GraphicsEnvironment.getLocalGraphicsEnvironment();
               GraphicsConfiguration gc =
genv.getDefaultScreenDevice().getDefaultConfiguration();

               Graphics2D gr = null;
               try {

                  resizedImage = gc.createCompatibleImage(pw, ph);
                  gr = resizedImage.createGraphics();
                  gr.setComposite(AlphaComposite.Src);

                  for (Map.Entry<RenderingHints.Key, Object> hint :
imageRenderingHints.entrySet()) {
                     gr.setRenderingHint(hint.getKey(), hint.getValue());
                  }

                  gr.setBackground(Color.BLACK);
```

```
            gr.setColor(Color.BLACK);
            gr.fillRect(0, 0, pw, ph);

            int sx1, sx2, sy1, sy2; // source rectangle coordinates
            int dx1, dx2, dy1, dy2; // destination rectangle coordinates

            dx1 = x;
            dy1 = y;
            dx2 = x + w;
            dy2 = y + h;

            if (mirrored) {
               sx1 = iw;
               sy1 = 0;
               sx2 = 0;
               sy2 = ih;
            } else {
               sx1 = 0;
               sy1 = 0;
               sx2 = iw;
               sy2 = ih;
            }

            gr.drawImage(image, dx1, dy1, dx2, dy2, sx1, sy1, sx2, sy2, null);

         } finally {
            if (gr != null) {
               gr.dispose();
            }
         }
      }

      g2.drawImage(resizedImage, 0, 0, null);

      if (isFPSDisplayed()) {

         String str = String.format("FPS: %.1f", webcam.getFPS());

         int sx = 5;
         int sy = ph - 5;

         g2.setFont(getFont());
         g2.setColor(Color.BLACK);
         g2.drawString(str, sx + 1, sy + 1);
         g2.setColor(Color.WHITE);
         g2.drawString(str, sx, sy);
      }

      if (isImageSizeDisplayed()) {

         String res = String.format("%d\u2A2F%d px", iw, ih);

         FontMetrics metrics = g2.getFontMetrics(getFont());
         int sw = metrics.stringWidth(res);
         int sx = pw - sw - 5;
         int sy = ph - 5;

         g2.setFont(getFont());
         g2.setColor(Color.BLACK);
```

```
            g2.drawString(res, sx + 1, sy + 1);
            g2.setColor(Color.WHITE);
            g2.drawString(res, sx, sy);
         }

         if (isDisplayDebugInfo()) {

            if (lastRepaintTime < 0) {
               lastRepaintTime = System.currentTimeMillis();
            } else {

               long now = System.currentTimeMillis();
               String res = String.format("DEBUG: repaints per second: %.1f", (double)
1000 / (now - lastRepaintTime));
               lastRepaintTime = now;
               g2.setFont(getFont());
               g2.setColor(Color.BLACK);
               g2.drawString(res, 6, 16);
               g2.setColor(Color.WHITE);
               g2.drawString(res, 5, 15);
            }
         }

/*****************************************************************************************/
/**                                                                                     */
/** @modified from original by Raquel Quesada                                           */
/**                                                                                     */
/*****************************************************************************************/

         if( pngimage != null)
         {
//          g2.drawImage(pngimage, 50, 50, null );
            g2.drawImage(pngimage, 10, 10, null );
         }

         Font oldFont = getFont();
//       Font newFont = new Font(Font.MONOSPACED, Font.PLAIN, 22);
         Font newFont = new Font(Font.MONOSPACED, Font.PLAIN, 50);
         FontMetrics newFontMetrics = g2.getFontMetrics(newFont);

         g2.setFont(newFont);

         final int lineHeight = newFontMetrics.getHeight();
//       int offset = 5;
         int offset = 10;

         for( String message : messages )
         {
//          g2.setColor(Color.BLACK);
//          g2.drawString(message, 6, 18 + offset);
//          g2.setColor(Color.GREEN);
//          g2.drawString(message, 5, 17 + offset);

            g2.setColor(Color.BLACK);
            g2.drawString(message, 6, 28 + offset);
            g2.setColor(Color.YELLOW);
            g2.drawString(message, 5, 27 + offset);

            offset += lineHeight;
```

```
            }

    /*****************************************************************************************/

            g2.setFont(oldFont);

            g2.setRenderingHint(KEY_ANTIALIASING, antialiasing);
            g2.setRenderingHint(KEY_RENDERING, rendering);
        }
    }

    private static final class PanelThreadFactory implements ThreadFactory {

        private static final AtomicInteger number = new AtomicInteger(0);

        @Override
        public Thread newThread(Runnable r) {
            Thread t = new Thread(r, String.format("webcam-panel-scheduled-executor-%d",
number.incrementAndGet()));
            t.setUncaughtExceptionHandler(WebcamExceptionHandler.getInstance());
            t.setDaemon(true);
            return t;
        }
    }

    /**
     * This runnable will do nothing more than repaint panel.
     */
    private static final class SwingRepainter implements Runnable {

        private WebcamViewPanel panel = null;

        public SwingRepainter(WebcamViewPanel panel) {
            this.panel = panel;
        }

        @Override
        public void run() {
            panel.repaint();
        }
    }

    /**
     * Image updater reads images from camera and force panel to be repainted.
     *
     * @author Bartosz Firyn (SarXos)
     */
    private class ImageUpdater implements Runnable {

        /**
         * Repaint scheduler schedule panel updates.
         *
         * @author Bartosz Firyn (sarxos)
         */
        private class RepaintScheduler extends Thread {

            /**
             * Repaint scheduler schedule panel updates.
             */
```

```
        public RepaintScheduler() {
            setUncaughtExceptionHandler(WebcamExceptionHandler.getInstance());
            setName(String.format("repaint-scheduler-%s", webcam.getName()));
            setDaemon(true);
        }

        @Override
        public void run() {

            // do nothing when not running
            if (!running.get()) {
                return;
            }

            repaintPanel();

            // loop when starting, to wait for images
            while (starting) {
                try {
                    Thread.sleep(50);
                } catch (InterruptedException e) {
                    throw new RuntimeException(e);
                }
            }

            // schedule update when webcam is open, otherwise schedule
            // second scheduler execution

            try {

                if (webcam.isOpen()) {
                    // FPS limit means that panel rendering frequency is
                    // limited to the specific value and panel will not be
                    // rendered more often then specific value
                    if (isFPSLimited()) {
                        executor.scheduleAtFixedRate(updater, 0, (long) (1000 / frequency),
TimeUnit.MILLISECONDS);
                    } else {
                        executor.scheduleWithFixedDelay(updater, 100, 1,
TimeUnit.MILLISECONDS);
                    }
                } else {
                    executor.schedule(this, 500, TimeUnit.MILLISECONDS);
                }
            } catch (RejectedExecutionException e) {

                // executor has been shut down, which means that someone
                // stopped panel / webcam device before it was actually
                // completely started (it was in "starting" timeframe)

                LOG.warn("Executor rejected paint update");
                LOG.trace("Executor rejected paint update because of", e);

                return;
            }
        }
    }

    /**
```

```
     * Update scheduler thread.
     */
    private Thread scheduler = null;

    /**
     * Is repainter running?
     */
    private AtomicBoolean running = new AtomicBoolean(false);

    /**
     * Start repainter. Can be invoked many times, but only first call will take
effect.
     */
    public void start() {
       if (running.compareAndSet(false, true)) {
          executor = Executors.newScheduledThreadPool(1, THREAD_FACTORY);
          scheduler = new RepaintScheduler();
          scheduler.start();
       }
    }

    /**
     * Stop repainter. Can be invoked many times, but only first call will take effect.
     *
     * @throws InterruptedException
     */
    public void stop() throws InterruptedException {
       if (running.compareAndSet(true, false)) {
          executor.shutdown();
          executor.awaitTermination(5000, TimeUnit.MILLISECONDS);
          scheduler.join();
       }
    }

    @Override
    public void run() {
       try {
          update();
       } catch (Throwable t) {
          errored = true;
          WebcamExceptionHandler.handle(t);
       }
    }

    /**
     * Perform single panel area update (repaint newly obtained image).
     */
    private void update() {

       // do nothing when updater not running, when webcam is closed, or
       // panel repainting is paused

       if (!running.get() || !webcam.isOpen() || paused) {
          return;
       }

       // get new image from webcam

       BufferedImage tmp = webcam.getImage();
```

```
            boolean repaint = true;

            if (tmp != null) {

               // ignore repaint if image is the same as before
               if (image == tmp) {
                  repaint = false;
               }

               errored = false;
               image = tmp;
            }

            if (repaint) {
               repaintPanel();
            }
         }
      }

      /**
       * Creates an overlay webcam panel and automatically starts webcam.
       *
       * @param webcam the webcam to be used to fetch images
       */
      public WebcamViewPanel(Webcam webcam) {
         this(webcam, true);
      }

      /**
       * Creates a new overlay webcam panel which displays video from a webcam with a text
overlay layer.
       *
       * @param webcam the webcam to be used to fetch images
       * @param start true if webcam shall be automatically started
       */
      public WebcamViewPanel(Webcam webcam, boolean start) {
         this(webcam, null, start);
      }

      /**
       * Creates a new overlay webcam panel which display video from a webcam with a text
overlay layer. If
       * panel size argument is null, then image size will be used. If you would like to
fill panel
       * area with image even if its size is different, then you can use
       * {@link OverlayWebcamPanel#setFillArea(boolean)} method to configure this.
       *
       * @param webcam the webcam to be used to fetch images
       * @param size the size of panel
       * @param start true if webcam shall be automatically started
       * @see OverlayWebcamPanel#setFillArea(boolean)
       */
      public WebcamViewPanel(Webcam webcam, Dimension size, boolean start) {

         if (webcam == null) {
            throw new IllegalArgumentException(String.format("Webcam argument in %s
constructor cannot be null!", getClass().getSimpleName()));
         }
```

```
        this.defaultSize = size;
        this.webcam = webcam;
        this.updater = new ImageUpdater();

        setDoubleBuffered(true);

        addPropertyChangeListener("locale", this);

        if (size == null) {
            Dimension r = webcam.getViewSize();
            if (r == null) {
                r = webcam.getViewSizes()[0];
            }
            setPreferredSize(r);
        } else {
            setPreferredSize(size);
        }

        if (start) {
            start();
        }
    }

    /**
     * Set new painter. Painter is a class which pains image visible when
     *
     * @param painter the painter object to be set
     */
    public void setPainter(Painter painter) {
        this.painter = painter;
    }

    /**
     * Get painter used to draw image in webcam panel.
     *
     * @return Painter object
     */
    public Painter getPainter() {
        return painter;
    }

    @Override
    protected void paintComponent(Graphics g) {

        super.paintComponent(g);

        if (image == null) {
            painter.paintPanel(this, (Graphics2D) g);
        } else {
            painter.paintImage(this, image, (Graphics2D) g);
        }
    }

    /** Open webcam and start rendering. */
    public void start() {

        if (!started.compareAndSet(false, true)) {
            return;
        }
```

```
        webcam.addWebcamListener(this);

        LOG.debug("Starting panel rendering and trying to open attached webcam");

        updater.start();

        starting = true;

        try {
            if (!webcam.isOpen()) {
                errored = !webcam.open();
            }
        } catch (WebcamException e) {
            errored = true;
            throw e;
        } finally {
            starting = false;
            repaintPanel();
        }
    }

    /**
     * Stop rendering and close webcam.
     */
    public void stop() {

        if (!started.compareAndSet(true, false)) {
            return;
        }

        webcam.removeWebcamListener(this);

        LOG.debug("Stopping panel rendering and closing attached webcam");

        try {
            updater.stop();
        } catch (InterruptedException e) {
            throw new RuntimeException(e);
        }

        image = null;

        try {
            if (webcam.isOpen()) {
                errored = !webcam.close();
            }
        } catch (WebcamException e) {
            errored = true;
            throw e;
        } finally {
            repaintPanel();
        }
    }

    /**
     * Repaint panel in Swing asynchronous manner.
     */
    private void repaintPanel() {
```

```
      SwingUtilities.invokeLater(repaint);
    }

    /**
     * Pause rendering.
     */
    public void pause() {
      if (paused) {
        return;
      }

      LOG.debug("Pausing panel rendering");

      paused = true;
    }

    /**
     * Resume rendering.
     */
    public void resume() {

      if (!paused) {
        return;
      }

      LOG.debug("Resuming panel rendering");

      paused = false;
    }

    /**
     * Is frequency limit enabled?
     *
     * @return True or false
     */
    public boolean isFPSLimited() {
      return frequencyLimit;
    }

    /**
     * Enable or disable frequency limit. Frequency limit should be used for <b>all IP
cameras
     * working in pull mode</b> (to save number of HTTP requests). If true, images will be
fetched
     * in configured time intervals. If false, images will be fetched as fast as camera
can serve
     * them.
     *
     * @param frequencyLimit
     */
    public void setFPSLimited(boolean frequencyLimit) {
      this.frequencyLimit = frequencyLimit;
    }

    /**
     * Get rendering frequency in FPS (equivalent to Hz).
     *
     * @return Rendering frequency
     */
```

```
    public double getFPSLimit() {
        return frequency;
    }

    /**
     * Set rendering frequency (in Hz or FPS). Minimum frequency is 0.016 (1 frame per
minute) and
     * maximum is 25 (25 frames per second).
     *
     * @param fps the frequency
     */
    public void setFPSLimit(double fps) {
        if (fps > MAX_FREQUENCY) {
            fps = MAX_FREQUENCY;
        }
        if (fps < MIN_FREQUENCY) {
            fps = MIN_FREQUENCY;
        }
        this.frequency = fps;
    }

    /**
     * Is displaying of some debug information enabled.
     *
     * @return True if debug information are enabled, false otherwise
     */
    public boolean isDisplayDebugInfo() {
        return displayDebugInfo;
    }

    /**
     * Display some debug information on image surface.
     *
     * @param displayDebugInfo the value to control debug information
     */
    public void setDisplayDebugInfo(boolean displayDebugInfo) {
        this.displayDebugInfo = displayDebugInfo;
    }

    /**
     * This method return true in case if camera FPS is set to be displayed on panel
surface.
     * Default value returned is false.
     *
     * @return True if camera FPS is set to be displayed on panel surface
     * @see #setFPSDisplayed(boolean)
     */
    public boolean isFPSDisplayed() {
        return frequencyDisplayed;
    }

    /**
     * This method is to control if camera FPS should be displayed on the webcam panel
surface.
     *
     * @param displayed the value to control if camera FPS should be displayed
     */
    public void setFPSDisplayed(boolean displayed) {
        this.frequencyDisplayed = displayed;
```

```
    }

    /**
     * This method will return true in case when panel is configured to display image
size. The
     * string will be printed in the right bottom corner of the panel surface.
     *
     * @return True in case if panel is configured to display image size
     */
    public boolean isImageSizeDisplayed() {
        return imageSizeDisplayed;
    }

    /**
     * Configure panel to display camera image size to be displayed.
     *
     * @param imageSizeDisplayed
     */
    public void setImageSizeDisplayed(boolean imageSizeDisplayed) {
        this.imageSizeDisplayed = imageSizeDisplayed;
    }

    /**
     * Turn on/off antialiasing.
     *
     * @param antialiasing the true to enable, false to disable antialiasing
     */
    public void setAntialiasingEnabled(boolean antialiasing) {
        this.antialiasingEnabled = antialiasing;
    }

    /**
     * @return True is antialiasing is enabled, false otherwise
     */
    public boolean isAntialiasingEnabled() {
        return antialiasingEnabled;
    }

    /**
     * Is webcam panel repainting starting.
     *
     * @return True if panel is starting
     */
    public boolean isStarting() {
        return starting;
    }

    /**
     * Is webcam panel repainting started.
     *
     * @return True if panel repainting has been started
     */
    public boolean isStarted() {
        return started.get();
    }

    public boolean isFitArea() {
        return drawMode == DrawMode.FIT;
    }
```

```
   /**
    * This method will change the mode of panel area painting so the image will be
resized and will
    * keep scale factor to fit into drawable panel bounds. When set to false, the mode
will be
    * reset to {@link DrawMode#NONE} so image will be drawn as it is.
    *
    * @param fitArea the fit area mode enabled or disabled
    */
   public void setFitArea(boolean fitArea) {
      this.drawMode = fitArea ? DrawMode.FIT : DrawMode.NONE;
   }

   /**
    * Image will be resized to fill panel area if true. If false then image will be
rendered as it
    * was obtained from webcam instance.
    *
    * @param fillArea shall image be resided to fill panel area
    */
   public void setFillArea(boolean fillArea) {
      this.drawMode = fillArea ? DrawMode.FILL : DrawMode.NONE;
   }

   /**
    * Get value of fill area setting. Image will be resized to fill panel area if true.
If false
    * then image will be rendered as it was obtained from webcam instance.
    *
    * @return True if image is being resized, false otherwise
    */
   public boolean isFillArea() {
      return drawMode == DrawMode.FILL;
   }

   /**
    * Get default painter used to draw panel.
    *
    * @return Default painter
    */
   public Painter getDefaultPainter() {
      return defaultPainter;
   }

   @Override
   public void webcamOpen(WebcamEvent we) {

      // if default size has not been provided, then use the one from webcam
      // device (this will be current webcam resolution)

      if (defaultSize == null) {
         setPreferredSize(webcam.getViewSize());
      }
   }

   @Override
   public void webcamClosed(WebcamEvent we) {
      stop();
```

```
    }

    @Override
    public void webcamDisposed(WebcamEvent we) {
        stop();
    }

    @Override
    public void webcamImageObtained(WebcamEvent we) {
        // do nothing
    }

    /**
     * This method returns true if image mirroring is enabled. The default value is false.
     *
     * @return True if image is mirrored, false otherwise
     */
    public boolean isMirrored() {
        return mirrored;
    }

    /**
     * Decide whether or not the image from webcam painted on panel surface will be
mirrored. The
     * image from camera itself is not modified.
     *
     * @param mirrored the parameter to control if image should be mirrored
     */
    public void setMirrored(boolean mirrored) {
        this.mirrored = mirrored;
    }

    @Override
    public void propertyChange(PropertyChangeEvent evt) {

    }

/*****************************************************************************/
/**                                                                         */
/** @author Raquel Quesada                                                  */
/**                                                                         */
/*****************************************************************************/


    public void replaceMessages(Collection<String> newMessages)
    {
        this.messages.clear();
        this.messages.addAll( newMessages );
    }

    public void replaceImage(BufferedImage image)
    {
        this.pngimage = image;
    }

/*****************************************************************************/

}
```

Appendix I: AR-TC WebcamMessagePanel.java

```
/*******************************************************************************/
/**                                                                           */
/** Copyright © 2016 Marcus Barstorp and Raquel Quesada                       */
/**                                                                           */
/** THE SOFTWARE IS PROVIDED “AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR  */
/** IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS  */
/** FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR    */
/** COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETER IN  */
/** AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION   */
/** WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.           */
/**                                                                           */
/*******************************************************************************/

package se.his.teamcenter.ar.main;

import java.awt.Color;
import java.awt.Font;
import java.awt.FontMetrics;
import java.awt.Graphics;
import java.awt.image.BufferedImage;
import java.util.Collection;
import java.util.List;
import java.util.concurrent.CopyOnWriteArrayList;

import javax.swing.JPanel;

public class WebcamMessagePanel extends JPanel
{
   private static final long serialVersionUID = 1L;

   public WebcamMessagePanel()
   {
      super();
   }

   private List<String> messages = new CopyOnWriteArrayList<>();
   private BufferedImage image = null;

   public void replaceMessages(Collection<String> newMessages)
   {
      this.messages.clear();
      this.messages.addAll(newMessages);
      this.repaint();
   }

   public void replaceImage(BufferedImage image)
   {
      this.image = image;
   }

   @Override
   protected void paintComponent(Graphics g) {
      super.paintComponent(g);
```

```
      g.setColor(Color.BLACK);
      g.fillRect(0, 0, this.getWidth(), this.getHeight());

/*********************************************************************************/
/**                                                                             */
/** @modified from original by Raquel Quesada                                   */
/**                                                                             */
/*********************************************************************************/

      if( image != null)
      {
         g.drawImage(image, 50, 50, null );
      }

      Font newFont = new Font(Font.MONOSPACED, Font.PLAIN, 50);
      FontMetrics newFontMetrics = g.getFontMetrics(newFont);

      g.setFont(newFont);

      final int lineHeight = newFontMetrics.getHeight();
      int offset = newFontMetrics.getMaxAscent();

      for( String message : messages )
      {
         g.setColor(Color.BLACK);
         g.drawString(message, 0, offset);
         g.setColor(Color.GREEN);
         g.drawString(message, 0, offset+1);

         offset += lineHeight;
      }

/*********************************************************************************/

   }
}
```

Appendix J: AR-TC log4j.properties.xml

```
/*****************************************************************/
/**                                                             */
/** Copyright 2012 Siemens Product Lifecycle Management Software Inc. */
/** All Rights Reserved.                                        */
/**                                                             */
/*****************************************************************/
```

Due to Siemens copyright restrictions this code cannot be publish in a public report but can be obtained from Siemens PLM Software.
For further information:
https://www.plm.automation.siemens.com/en_us/products/teamcenter/

Description:

- Sets the Logging for the entire Web Application.
- The level can be DEBUG, INFO, WARN, ERROR, FATAL, or OFF.
- There are two predefined Apprenders, TheConsole and AppLogFile.

Appendix K: AR-TC oppolicy.xml

```
/*********************************************************************************/
/**                                                                             */
/** Copyright © 2016 Raquel Quesada and Contributors                            */
/**                                                                             */
/** THE SOFTWARE IS PROVIDED “AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR   */
/** IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS */
/** FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR */
/** COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETER IN */
/** AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION */
/** WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.             */
/**                                                                             */
/*********************************************************************************/

<?xml version="1.0" encoding="UTF-8"?>
<ObjectPropertyPolicy xmlns="http://teamcenter.com/Schemas/Soa/ObjectPropertyPolicy">
   <ObjectType name="WorkspaceObject">
      <Property name="object_string" ></Property>
      <Property name="last_mod_date" />
      <Property name="creation_date" />
      <Property name="object_name" />
      <Property name="object_type" />
      <Property name="owning_user" />
      <Property name="ref_list"  withProperties="true" /><Property
        name="item_id">
</Property>
      <Property name="item_revision_id"></Property>
      <Property name="ref_names"></Property>
   </ObjectType>
   <ObjectType name="ImanFile">
      <Property name="file_name" />
      <Property name="file_ext" />
      <Property name="file_size" />
   </ObjectType>
   <ObjectType name="ItemRevision">
      <Property name="creation_date" />
   </ObjectType>

</ObjectPropertyPolicy>
```