Deliverable D100.3

EPES Concept

EPES Project

Eco-Process Engineering System For Composition of Services to Optimize Product Life-cycle

FoF-ICT-2011.7.3-285093

Public Project Report

Project Facts:	**Duration:**	**36 Months (September 2011 – August 2014)**
	Programme:	**FP7 – ICT**
	Website:	**http://www.epes-project.eu**

The EPES Project is co-funded by the European Commission
under the FoF-ICT theme of the 7th Framework Programme (2007-2013)

This publication was completed with the support of the European Commission under the 7th Framework Programme. The contents of this publication do not necessarily reflect the Commission's own position.

Authors and EPES Project Partners

	Fundacion Tecnalia Research & Innovation http://www.tecnalia.com	**Spain**
	ATB **Institut für angewandte Systemtechnik Bremen GmbH** http://www.atb-bremen.de	**Germany**
	Sisteplant SL http://www.sisteplant.com	**Spain**
	VTT Technical Research Centre of Finland http://www.vtt.fi	**Finland**
	Esteco SpA http://www.esteco.com	**Italy**
	Grupe Tamoin SA http://www.tamoin.com	**Spain**
	nkt cables GmbH http://www.nktcables.com	**Germany**
	EADS Innovation Works UK http://www.eads.com	**UK**

Summary

This document represents the deliverable D100.3 *EPES Concept* and presents in detail the concept developed for the envisaged EPES solution, comprising the individual modules of the EPES solution, the EPES reference architecture and the implementation plan.

The EPES project's objective is to support the Product Service System (PSS) by developing a novel eco process engineering system which will constitute a comprehensive platform, enabling dynamic composition of services adaptable to the different products and operating conditions. Thus, the EPES solution will support continuous improvement of products in operation along the life cycle, applying the best up –to-date technologies for end of life disposal of the products and for improving future product designs.

To achieve these goals, the EPES project intends to develop:

- A set of ICT tools allowing
  - easy configuration/adaptation of services
  - storing and re-using knowledge in order to improve existing services and develop new ones
- A methodology and working handbook

The set of ICT tools, together with the methodology and working handbook, will enable the manufacturing companies to enter a continuous process of upgrading their products, along with their life cycle, within the frame of the virtual factory and PSS concept through a configurable and adaptable set of services.

The key components of the EPES solution include:

- **Virtual Collaborative Network** (VCN): to allow the tracking of business optimization opportunities through a networked infrastructure. It also provides collaborative web content and document content management capabilities.
- **Service Generator** (SG): to allow configuring services, deploying them and to provide a cockpit/portal to access the EPES solution.
- **Decision Making Module** (DMM): to allow decision-makers to optimize and to analyze business process through dedicated tools.
- **Simulation Module** (SM): execution of external simulations and provision of parameters for calculation of key performance indicators (KPI).

The EPES methodology will provide a comprehensive approach on how to use the EPES solution. In order to enable applying the solution in a new situation, the methodology will include such aspects as business process model (mapping it and making it explicit), collection of sustainability intelligence (SI) sources, classification and structuring of the SI information, etc. The overall way of working and expected functionalities of the SW modules are also to be included in the methodology.

In order to ensure the industrial relevance of the EPES methodology, three application scenarios at the three industrial users of the consortium were studied and analysed for deriving a list of requirements. Furthermore, the RTD consortium partners have analysed state-of-the-art technologies, developments and available solutions based on their expertise (the results of this process have been presented in deliverable D100.1 *State-of-the-Art Analysis*). Starting with the BC specific requirements and taking into account the technological state-of-the-art, a generic set of re-

quirements has been extracted (presented in deliverable D100.2 *Requirements Analysis*). These are the requirements that the EPES solution must fulfil, and it is with these requirements in mind that the EPES concept has been devised.

This report includes a detailed description of the concept for all key components of the EPES solution, architecture and methodology. The implementation plan, defining the scope of the Early and Full Prototypes, is also defined.

Table of Contents

Abbreviations

API	Application Programming Interface
BC	Business Case
BI	Business Intelligence
BPEL	Business Process Execution Language
BPM	Business Process Management
BPMN	Business Process Model and Notation
BPMS	Business Process Management System
CMMS	Computerized Maintenance Management Systems
CMSD	Core Manufacturing Simulation Data (Standard)
CRUD	Create-Read-Update-Delete
DES	Discrete Event Simulation
DM	Decision Making
DMM	Decision Making Module
DoE	Design of Experiments
DTS	Distributed Temperature Sensing
e.g.	exempli gratia = for example
EPES	Eco-Process Engineering System
ERP	Enterprise Resource Planning
etc.	et cetera
ETL	Extract, Transform and Load
EU	European Union
GIS	Geographic Information System
GUI	Graphical User Interface
i.e.	id est = that is to say
ICT	Information and Communication Technology
IPR	Intellectual Property Rights
IT	Information Technology
KPI	Key Performance Indicator
LC	Life Cycle
LCI	Life Cycle Inventory
LCM	Life Cycle Management
OLAP	Online Analytical Processing

OSOA	Open SOA
PAM	Password Authentication Module
PSS	Product Service System
R&D	Research and Development
REST	REpresentational State Transfer
RESTful	Conforming to REST constraints
RPC	Remote Procedure Calls
RTD	Research and Technological Development
RTE	Round-trip Engineering
S&T	Scientific and Technological
SCA	Service Component Architecture
SCADA	Supervisory Control and Data Acquisition
SDO	Service Data Objects
SG	Service Generator
SGM	Service Generator Module
SI	Sustainability Intelligence
SM	Simulation Module
SME	Small and Medium Sized Enterprise
SOA	Service Oriented Architecture
SOAP	Simple Object Access Protocol
SW	Software
TBL	Triple Bottom Line
TOC	Theory Of Constraints
UML	Unified Modelling Language
VBE	Virtual Breeding Environment
VCN	Virtual Collaborative Networks
VFDB	Virtual Factory Database
VFKDB	Virtual Factory Knowledge Database
WP	Workpackage
WS	Web Services
WSDL	Web Services Description Language
XML	eXtensible Markup Language

1 Introduction

1.1 Document Purpose

This document summarizes the conceptual development of the EPES project. It presents the deliverable D100.3 *EPES Concept*, which is the result of Task T130 *Concept Definition* from Work Package WP100, *Requirement Analysis and Concept*.

The EPES concept provides the overall concept for methodology, architecture, software components, service infrastructure, and implementation framework. The concept has been developed based on the results of the tasks T110 (*State-of-the-art Analysis*) and T120 (*Requirements Collection and Analysis*).

1.2 Overview

The strategic objective of the EPES project is to develop a novel eco process engineering system, which will constitute a comprehensive platform enabling a dynamic composition of services adaptable to the different products and operating conditions, supporting the Product Service System.

This novel service oriented framework will allow industries to evaluate the performance of engineered products considering their whole lifecycle rather than only early stages such as design and manufacturing (see Figure 1-1 below). The capabilities resulting from the research will enable the capitalisation on trustable global and local sustainability intelligence. Product engineering teams can exploit this intelligence to adapt design, operation and disposal strategies through managed "eco-constraints" relevant to their market contexts.

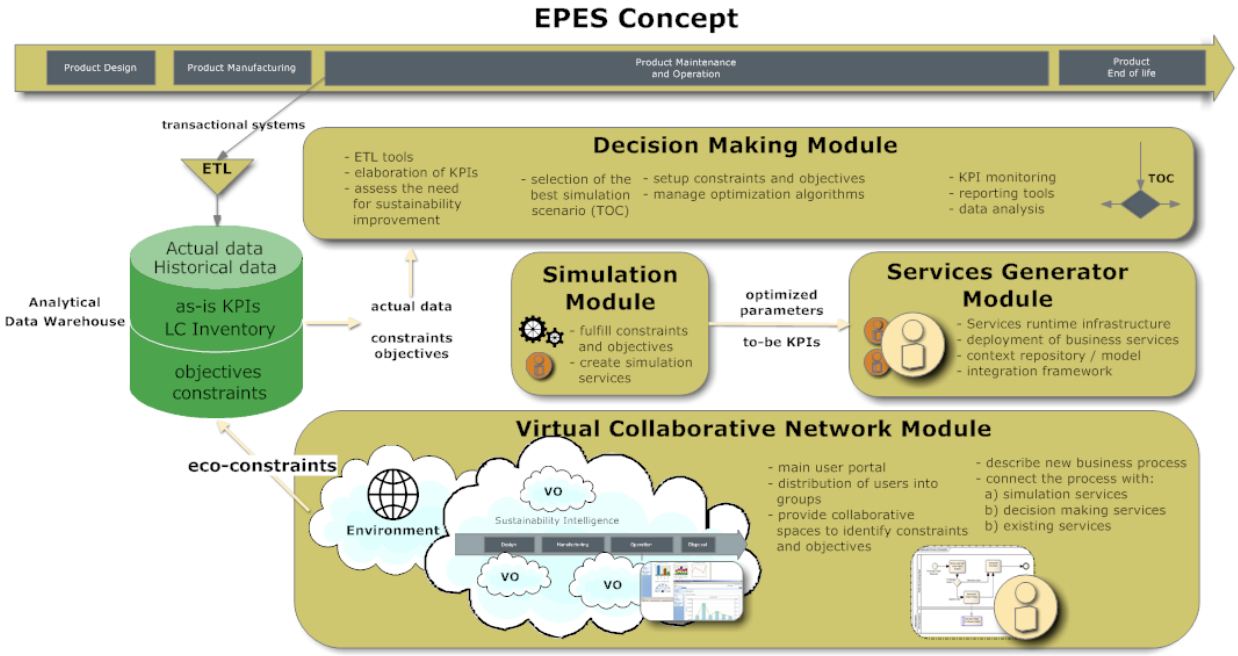

Figure 1-1: Basic EPES Concept

This concept document defines the relation to the state of the art at the beginning where the identified gaps and EPES contributions in relevant R&D areas are mentioned. EPES conceptual development is guided by a generic scenario obtained from analysing detailed requirements of three industrial application scenarios. This ensures industry relevant development of EPES methodology and services/modules as the requirements are directly derived from three different use cases and analysing the current state of the art. Based on these requirements, overall EPES reference architecture, features and functionality required for EPES solutions are derived. In the

final part of this document, required implementation framework, plans for implementation and conclusions are drawn.

1.3 Approach Applied

The EPES concept presented here is the result of a process already partly presented in the previous deliverable D100.2 Requirements Analysis and illustrated in Figure 1-2 below.

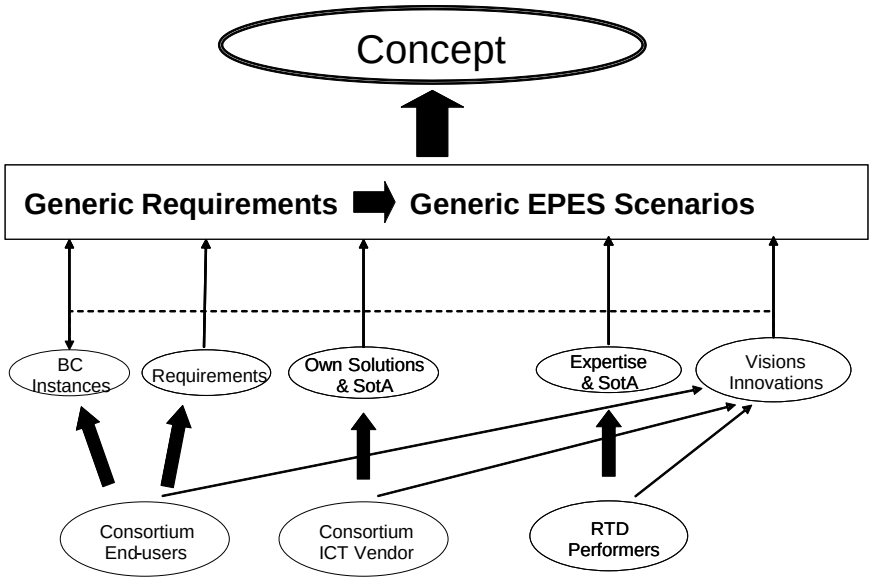

Figure 1-2: Approach applied in the requirements analysis

Shortly, the steps of that process were:

1. Detailed analyses of the application cases by the three industrial partners.
2. Creation of the textual descriptions of the application cases and extraction of needs and requirements.
3. Collection of the information/insight into the market available solutions and into the state-of-the-art of corresponding application.
4. On top of that the RTD performers have created an in-depth analysis of the state-of-the-art R&D activities in the relevant areas, what was used, enriched by the expertise (of RTD performers), for creation of a generic set of requirements and generic application scenarios.
5. All participants in the above described activities (see Figure 1-2) have also provided technical visions and innovation ideas to complete the generic requirements. The attempt was done to introduce the long-term visions for the future improvements of the solutions.
6. The generic scenarios presented in section 2 are to be observed also as a kind of contribution to the generic requirements upon the EPES components functionalities.
7. Based on the defined requirements the key EPES components are specified in detail.

1.4 Document Structure

The document is structured as follows:

Section 1 – presents the purpose of the document, an overview of the project and the report's position in the project, as well as approach applied.

Section 2 – provides the generic scenarios as explained above.

Section 3 – describes the EPES reference architecture.

Section 4 – provides detailed description of the Methodology Concept.

Section 5 – provides detailed description of the Virtual Collaborative Network Module.

Section 6 – provides detailed description of the Service Generator Module.

Section 7 – provides detailed description of the Decision Making Module.

Section 8 – provides detailed description of the Simulation Module.

Section 9 – includes description of the implementation framework and implementation plan.

Section 10 – provides conclusions indicating future work.

2 Generic Scenario

This section presents the process used to identify the generic EPES solution architecture which meets the requirements derived from industrial business cases analysis. In the first phase, the case scenarios are illustrated here from the technical viewpoint for potential EPES solution integration. In the second phase, the potential EPES solution integrations envisaged for each BC are abstracted into a *Generic Scenario* which will serve as the basis for the EPES solution specification.

The EPES project covers three industrial application scenarios (business cases):

- **Business Case 1:** Engineering maintenance services for optimizing maintenance and increasing availability of wind turbines
- **Business Case 2:** Power grid control systems for improved identification of maintenance needs and improved monitoring of grid load and safety limits
- **Business Case 3:** Support for optimized design and manufacturing of aircraft wings

Table 2-1 presents an overview of the envisaged usage of the EPES ICT components within the three EPES Business Cases. The following subsections first describe each of the three business cases[1] and afterwards present the generic scenario derived from these business cases.

Table 2-1: Envisaged usage of the EPES ICT components

EPES ICT Component	BC1	BC2	BC3
Virtual Collaborative Networks / VBE Virtual Factory	++	+	++
Service Generator	+	++	++
Simulation Module	++	++	++
Decision Making Module	++	++	+

2.1 Business Case 1

ENGINEERING MAINTENANCE SERVICES FOR OPTIMIZING MAINTENANCE AND INCREASING AVAILABILITY OF WIND TURBINES

The overall objective of this Business Case is to optimize TAMOIN-TER maintenance procedures carried out at their client's wind farms so as to achieve maximum wind turbines availability at minimum maintenance costs focusing on a holistic sustainable approach, through the KPI monitoring and the adoption of simulation and optimization services provided by EPES system.

The first key objective will focus on a system for capturing environmental eco-constraints derived from local regulations, regional regulations and sector recommendations, aiming at the identification and development of eco indicators involving social and environmental measures. On the other hand, the system will also allow the extraction, transformation and loading of human, material and technical data into a data warehouse in order to support the definition and elaboration of traditional key performance indicators involving profit measures. The monitoring

[1] Detailed overview and objectives of the three business cases is presented in deliverable D100.2

and tracking of these indicators (before and after the implementation of business process optimizations) will support the decision making process to achieve an optimal management of TER maintenance process.

The second key objective is to improve the maintenance scheduling capabilities of TER by the implementation of a scheduling system based on mathematical models aimed to optimize the total energy produced by the wind turbines while minimizing the maintenance costs. These models will be enhanced with service oriented capabilities and offered as services through EPES solution in order to provide collaborative, traceable and trustable decision support to TER. This scheduling facility will be accessed through a collaborative workflow system, which will allow modelling and executing TER maintenance process, the connection of this process to existing IT legacy systems, the assignment of the different process activities to different users and the monitoring of the whole process through the KPIs elaborated as result of the first key objective.

The main components of the BC1 scenario are:

- PRISMA Computerized Maintenance Management System, which acts as an ERP and as a maintenance planning and tracking tool
- Scheduling models (based on linear programming or genetic algorithms), which will provide TER with optimization tools to improve the maintenance process
- Unstructured technical data coming from the wind turbines SCADA systems

Figure 2-1 below illustrates how the EPES solution can be integrated with the BC1.

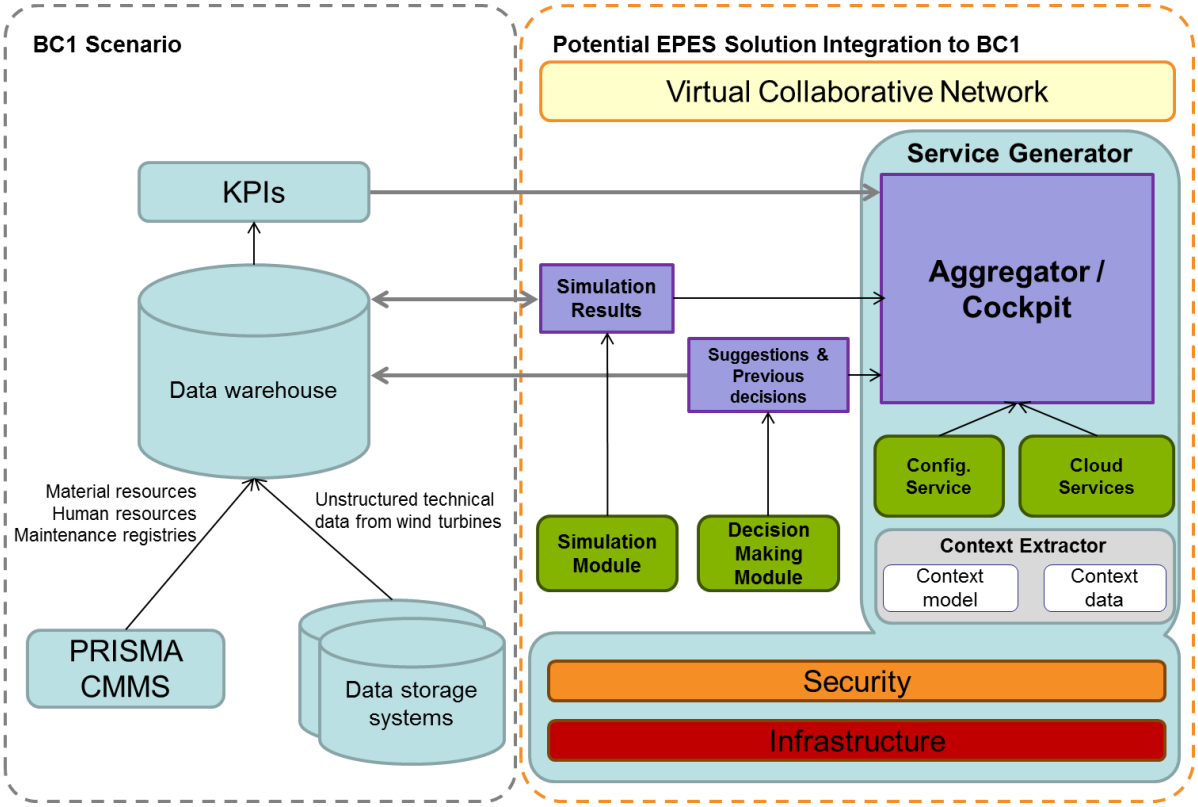

Figure 2-1: An architectural viewpoint of the EPES solution integration with BC1

2.2 Business Case 2

POWER GRID CONTROL SYSTEMS FOR IMPROVED IDENTIFICATION OF MAINTENANCE NEEDS AND IMPROVED MONITORING OF GRID LOAD AND SAFETY LIMITS

The main objective of this business case is to provide new business models to offer their customers maintenance services (e.g. in respect to detailed analysis detection of faults, etc.), as well as configurable cable monitoring services, easily adaptable to customer specific requirements, enabling an optimal usage of available capacity and secure operation of high and extra-high voltage cables.

Identified key objectives of the EPES solution's integration for this business case[2] comprise:

- Improved identification of maintenance needs
- Improved monitoring of the grid load
- Provide adjusted parameter sets to human operator for final decision

Figure 2-2 below presents the architectural view of integrating the EPES solution with the BC2:

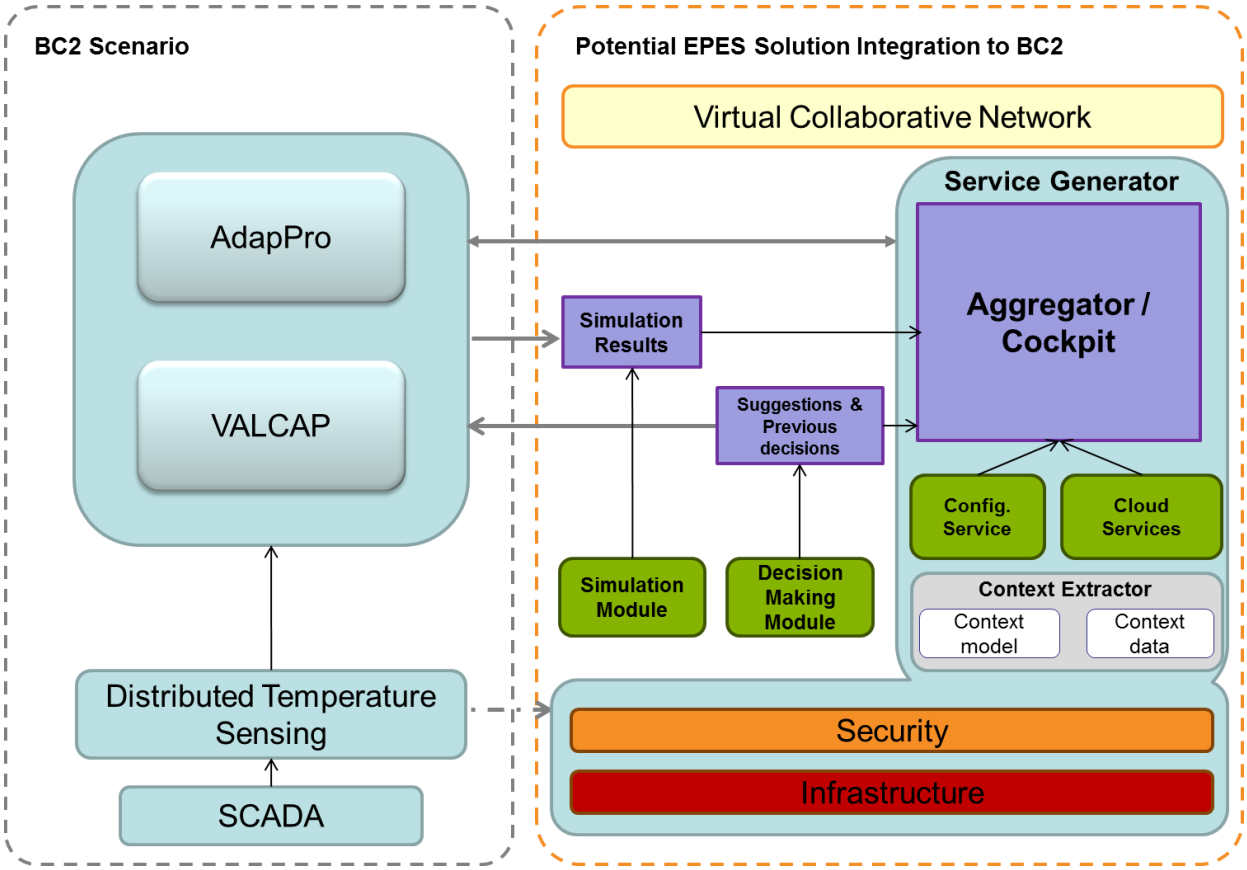

Figure 2-2: An architectural viewpoint of the EPES solution integration with BC2

The main components of the BC2 scenario are:

- AdapPro: allows computing of load predictions for a chosen date and time in the future
- VALCAP: allows monitoring the power grid's parameters
- Distributed Temperature Sensing (DTS): allows sensing the temperature of power cables

[2] Detailed overview and objectives of business case 2 is listed in section 2.2, page 20 of deliverable D100.2

- SCADA: control and data acquisition system used to interface with the power cables

The data from VALCAP and DTS is used in conjunction with the SM and DMM to compute the parameters needed for customizing new services.

2.3 Business Case 3

SUPPORT FOR OPTIMIZED DESIGN AND MANUFACTURING OF AIRCRAFT WINGS

Main objectives of the EPES system in this BC are to bring new supporting services for assessment of productivity and sustainability KPIs on wing design (defined at the conceptual stage). By using such services the EPES system enables to validate production scenarios on early stages considering dynamic conditions. Furthermore, improved decision making for an optimum manufacturing facility will be supported.

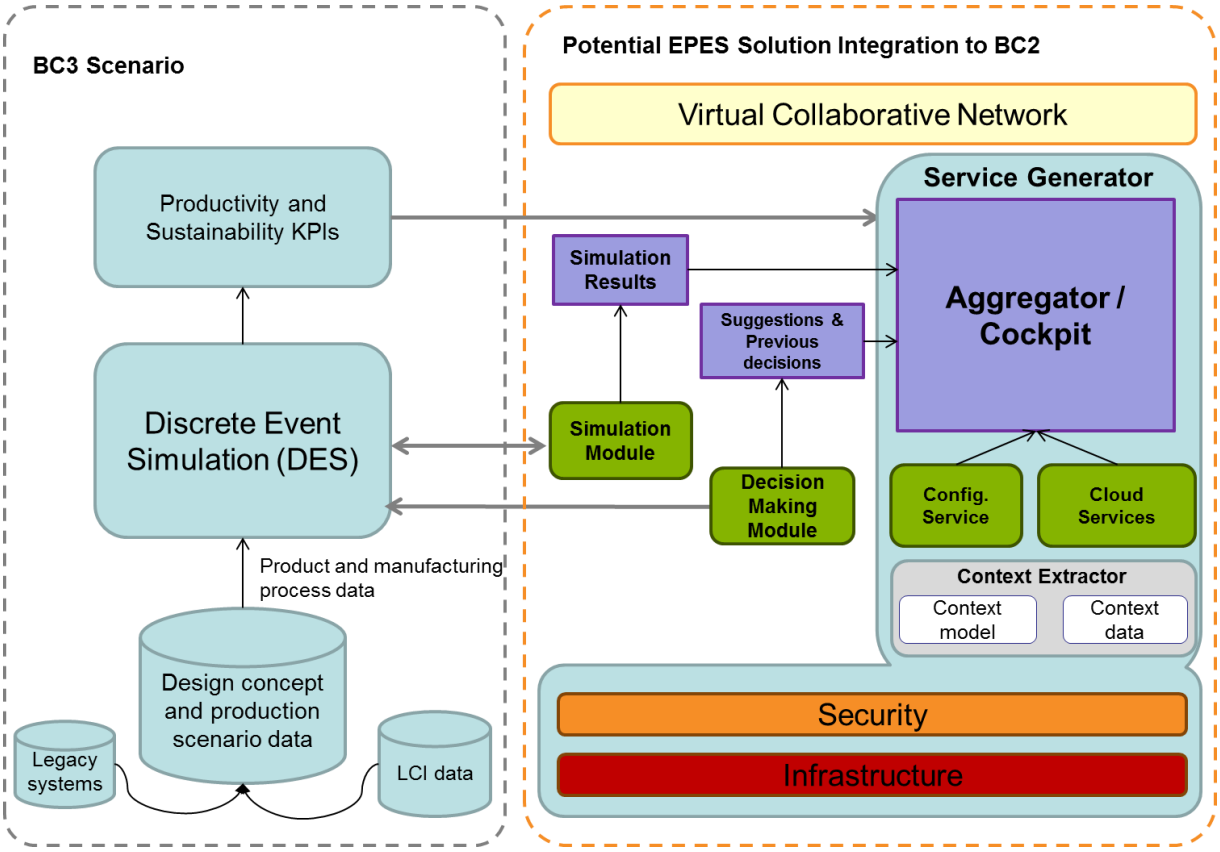

Figure 2-3: An architectural viewpoint of the EPES solution integration with BC3

The end users of the EPES system will be "design for manufacturing" engineers, within the enterprise, contributing in the assessment of design concepts. The EPES system will support them to make informed decisions on the performance of design concepts from the manufacturing perspective. EPES systems brings an opportunity to integrate the assessment of traditional manufacturing Key Performance Indicators (KPIs) such as time and production rate with those related to the sustainability of the production processes. The essential questions answered through this assessment are:

- Productivity KPIs: What production rate can be achieved for a design using a given set of processes and resources?
- Sustainability KPIs: What are the energy consumption, the emissions and the hazardous material waste resulting from the manufacturing for a design using a given set of processes and resources?

The main components of the BC3 scenario are:

- Enterprise existing legacy systems
- Discrete event simulation (DES) software and Excel interface to run the simulation models
- Life Cycle Inventory data

2.4 Common Scenario

The potential integration views of the three BC scenarios with the EPES solution have been analysed in order to extract a generic scenario. Figure 2-4 below presents a generic view of the envisaged EPES solution, highlighting on the one hand the generic EPES solution integration, and on the other hand a generic view of BC-specific infrastructure.

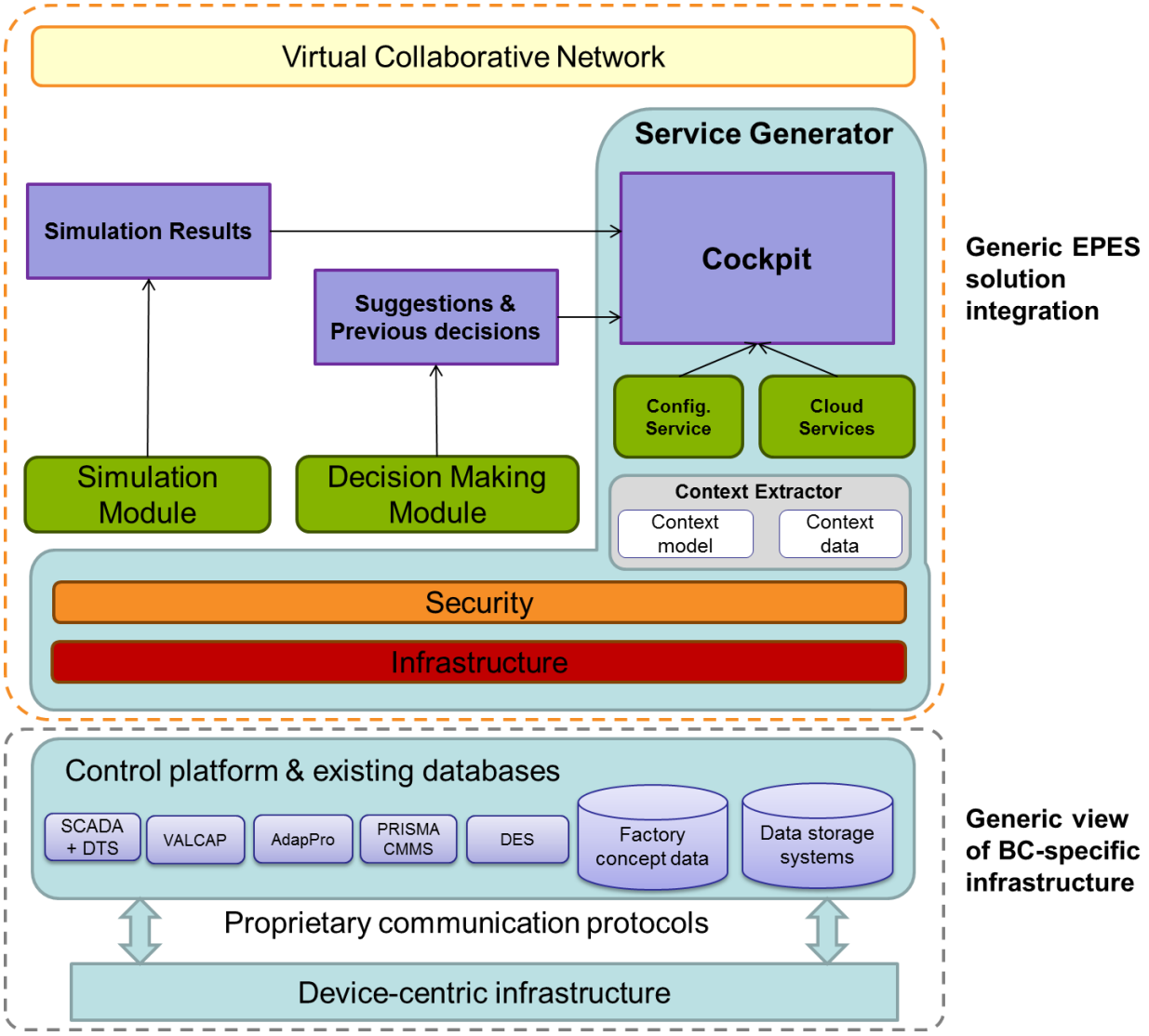

Figure 2-4: EPES Architecture: generic view of business cases

Table 2-2 lists the identified main specific technical aspects of the three cases which will be addressed by the generic EPES solution.

Table 2-2: BC-specific technical aspects addressed by the EPES solution

Business Case	Main Area of Interest	Technical issues to be addressed with EPES
BC1	Engineering maintenance ser-	Monitor parameters of wind turbines to detect

	vices for optimizing maintenance and increasing availability of wind turbines	(preventive) maintenance needs
BC2	Power grid control systems for improved identification of maintenance needs and improved monitoring of grid load and safety limits	Monitoring of cable temperatures Monitoring of grid load
BC3	Support for optimized design and manufacturing of aircraft wings	Use simulations to estimate production rate, energy consumption, emissions, hazardous material waste

Figure 2-5 below presents a UML use case diagram of the main features of the EPES modules. There are three kinds of users of the system. The developer user is responsible for the integration of external simulation tools by using the interactive features of the Simulation module. The expert user is responsible for configuring simulation services in the Service Generator module, by wrapping the services provided by the Simulation module. The expert user uses the interactive features of the Decision Making module to run simulations and process the results.

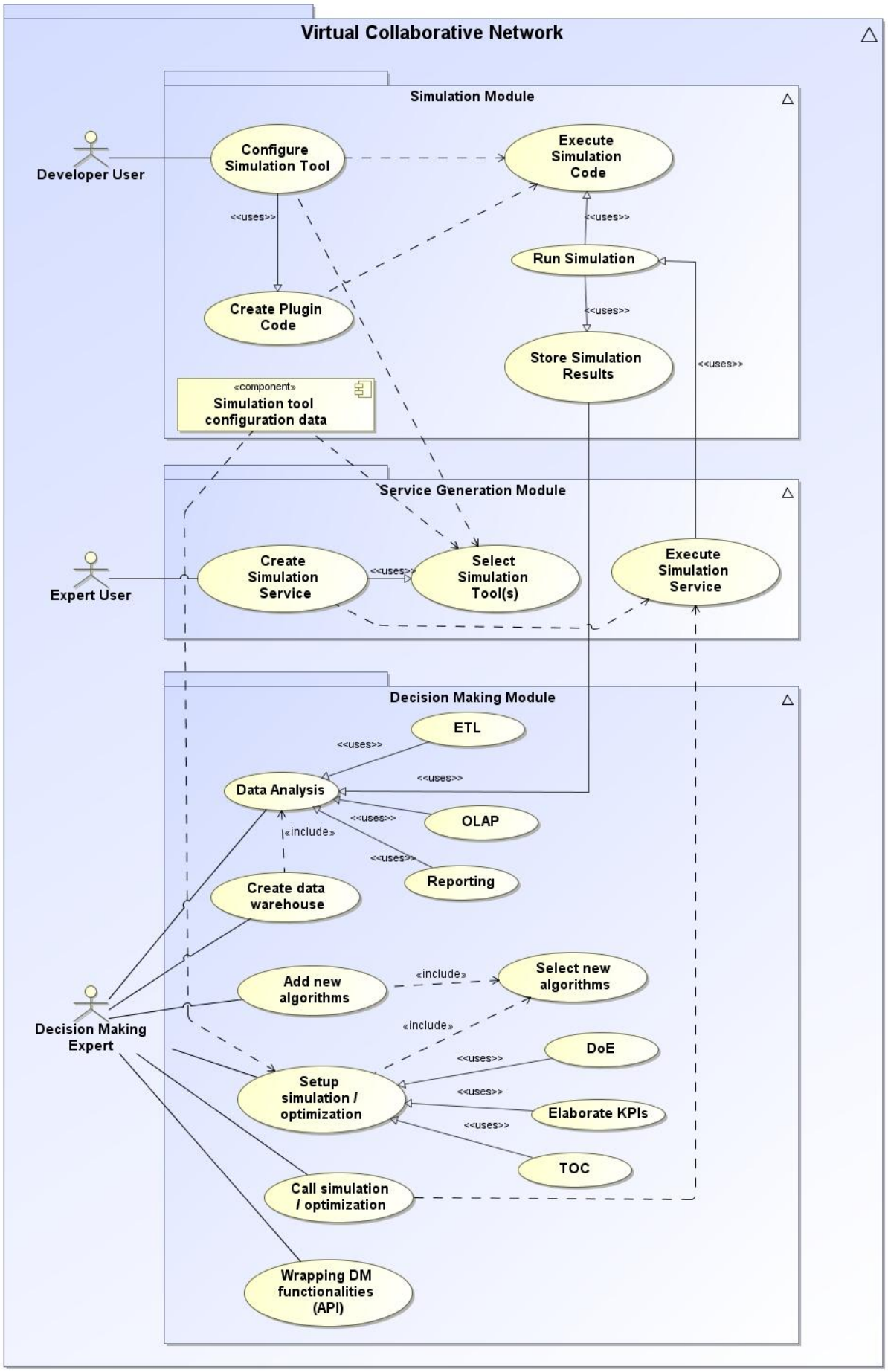

Figure 2-5: A UML use case diagram of the main EPES module features

There are several dependencies between the use-cases of the modules. The dashed arrows in the figure represent flows of persistent data that must be available through the common Content Repository. The “uses” arrows represent inclusion of use-cases and “extends” represent extension. A “uses” dependency between the modules will be implemented as a call to a service interface, while the dashed arrows represent access to common persistent data in the content repository.

3 EPES Reference Architecture

3.1 Rationale

A Reference Architecture captures the essence of the architecture of a collection of systems. The purpose of a Reference Architecture is to provide guidance for the development of architectures for new versions of the system or extended systems and product families[3].

Reference architectures[4] have emerged as a special type of architectures that provides major guidelines for the specification of concrete architectures of one class of systems. Depending on the context in which they are defined, two types of reference architectures can be discerned: practice-driven and research-driven reference architectures. *Practice-driven* reference architectures are defined when sufficient knowledge has been accumulated in a domain to propose the "best of best practices" architecture. They are designed to provide a standardized view on a class of systems. *Research-driven* reference architectures provide a "futuristic" view on a class of systems that are expected to become important in the future, but by the time of the architecture definition are seen as hard to build (e.g., due to functional complexity). These architectures aim at facilitating the design of the first systems from a class of systems.

In developing the EPES concept, it has become apparent that a research-driven reference architecture is needed for synchronizing the implementation in the three BC and in future applications of the EPES solution. This reference architecture is described in the following section, also providing the common vocabulary used to refer to the solution's various components and aspects.

3.2 Architecture Description

The EPES Project addresses a generic solution for supporting the Product Service System (PSS) by enabling dynamic composition of services adaptable to the different products and operating conditions. In order to support this, the EPES solution will enable continuous improvement of products in operation along the life cycle, applying the best up-to-date technologies for end of life disposal of the products and for improving future product designs.

The overall EPES reference architecture is illustrated in Figure 3-1 below.

[3] Muller, G., "A Reference Architecture Primer", Eindhoven University of Technology, Eindhoven, 2011.

[4] Angelov, S., Trienekens, J. J., Grefen, P., "Towards a Method for the Evaluation of Reference Architectures: Experiences from a Case", in *Proceedings of the 2nd European conference on Software Architecture* (ECSA '08), Ron Morrison, Dharini Balasubramaniam, and Katrina Falkner (Eds.), Springer-Verlag, Berlin, Heidelberg, pp. 225-240.

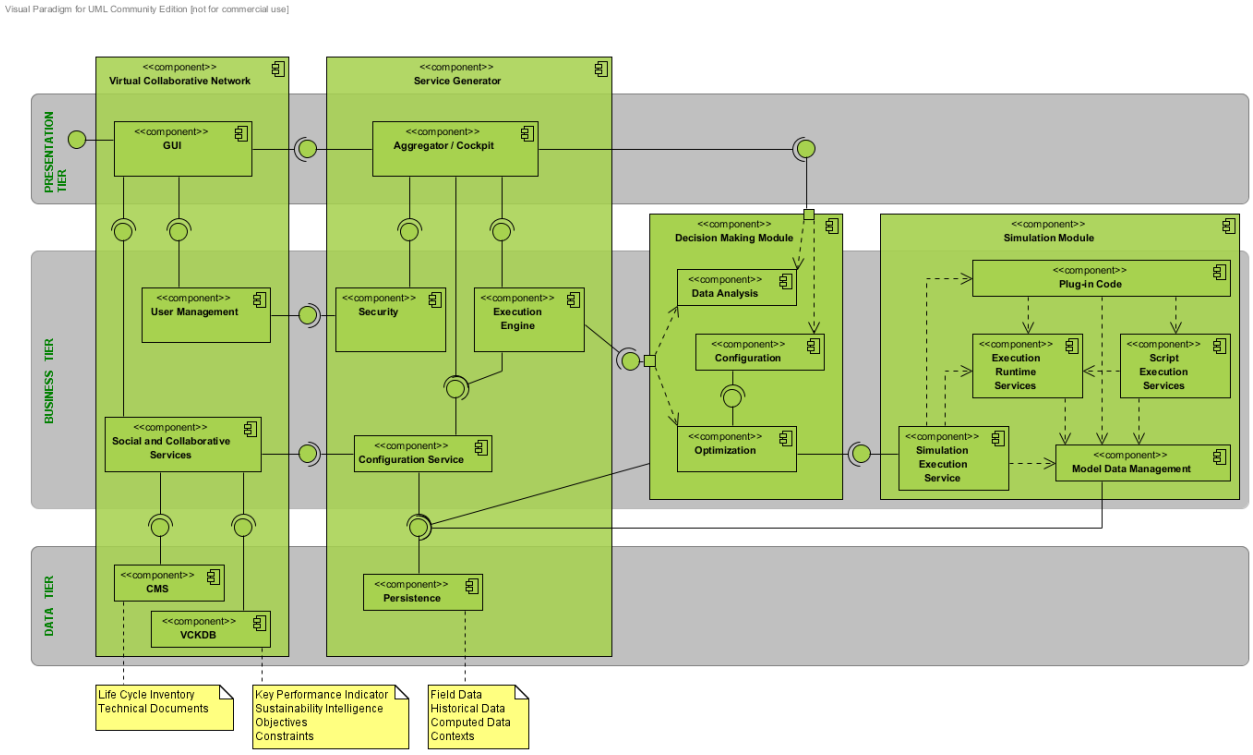

Figure 3-1: EPES Reference Architecture

Virtual Collaborative Network

The VCN main functionality is to provide a main point of access for the EPES non-expert end-users through user interfaces, including capabilities for supporting the aggregated GUIs provided by the Service Generation Module. On the other hand it provides the technical infrastructure for the distribution of users into groups; content management and sharing (documents and web content) mechanisms; workflow engine for contents production, consumption and transformation. The use of workflow capabilities will let end-users connect the business processes to high level optimization and simulation business services composed by the Service Generator Module or other EPES Modules. In addition, the VCN will provide a VFKDB, which will contain actual data, historical data, identified constraints and objectives from the collaborative networks, KPIs and Life-Cycle Inventory data. The database will be remotely accessible by all EPES modules for supporting CRUD operations.

Service Generator

The Service Generator's main functionality is creating, updating, and deploying configurations for application-specific services. It interacts with the Simulation Module and the Decision Making Module to compute parameters for updated configurations. The SG can also store / retrieve configurations in/from the Virtual Collaborative Network.

Interaction between the SG and the other modules is done programmatically (a candidate method considered during the concept development phase is to invoke methods over web service end-points); however, the Aggregator/Cockpit component of the SG is a GUI wrapper that enables users at the VCN level to have a visual overview of the SM and DMM, by aggregating their GUIs.

Simulation Module

The simulation module of the EPES platform provides a capability for running numerical analyses related to the life-cycle assessment process. It provides simulation services through an ab-

stract service interface that allows the higher level components of the EPES platform to use simulation as an interchangeable service, according to the principles of the service-oriented de-sign paradigm.

Through the service interfaces of the SM, the other modules are able to configure and execute simulation models without any dependencies on the nature or the data representation of the simulation models. It allows the other modules to present configuration options and numerical parameters of simulation modules without any prior knowledge of the internal structure of the model.

Decision Making Module

DMM is an interactive computer-based system intended to help decision-makers to use data and models to identify and solve problems and to make decisions. The Simulation Module will be used in order to explore as much as possible the space of all possible solutions, possibly generating a large amount of simulation data. DMM interacts with VCN module to obtain and save the domain knowledge needed to the multi objective analysis, the optimization procedure, the screening and decision phase.

The DMM will make use of TOC methodology in order to allow the EPES system to focus on the constraints of the described process. On the one hand, there will be some Traditional indicators for measuring the overall factory performance, but on the other hand, there will be new KPIs established and defined in the DMM. These KPIs will be traced to the measured constraints, and the DMM will show, in each Business Case, where the constraints are, and how to measure the performance level of the organization, in order to take the best decision in a range of values proposed by the SM and taking into account the localized constraint.

The DMM and the SM will be directly related in a bidirectional way, to ensure feedback analysis before the final decision is taken.

4 Methodology Concept

4.1 Introduction

Globalization is demanding more requirements, to both private companies and public organizations in their ability to react to the changes demanded by the market. These include changes in the type of demand or changes in regulations. The ability of organizations to adapt their offers of goods and services is a fundamental part of the new concept of customer value. The products themselves are not sufficiently attractive because there is generally an oversupply and what makes the difference is to provide services around these products. These challenges include compliance with internal regulations, focusing on external and international quality control (traceability), fraud prevention and care of the environment. Therefore, the EPES methodology will help to introduce processes in organizations that allow them to enter in a continuous improvement cycle to comply with these requirements over time.

In this section the basic concept of EPES methodology is defined and the structure of it is proposed.

4.2 EPES Methodology Objectives, Structure and Challenges

EPES methodology goes beyond organization efficiency; it will help organizations to understand what value is delivered by an organization and how it is achieved. This means analysing, documenting and improving the way that people and systems work together. The methodology responds to the question of how an organization can achieve a greater agility, efficiency and effectiveness. The answer is greater control and efficiency in the ability to change the business processes. It proposes to integrate the various disciplines of corporate management directly to the operation of processes, i.e. proposes a systemic approach to BPM. To integrate the different management disciplines, it will provide models of innovative SME-SME and SME RTD networking applicable a specific form of collaboration patterns, methods and services for set-up and management of such networks, as well as methods for collection, structuring and reusing of specific knowledge necessary for operation of these networks. EPES methodology for setting up collaborative environments will elaborate a strategy for collaboration defining roles and workflow through business process management (BPM).

BPM is a holistic management approach focused on aligning all aspects of an organization with the requirements and needs of clients. It promotes business effectiveness and efficiency while striving for innovation, flexibility, and integration with technology. BPM attempts to improve processes continuously. It can therefore be described as a "process optimization process." It is argued that BPM enables organizations to be more efficient, more effective and more capable of change than a functionally focused, traditional hierarchical management approach[5]. BPM encompasses the increasing support of IT to improve, innovate and manage the process from start to finish, which determine the business results, create customer value and enable the achievement of business goals with greater agility[6]. Goals like promoting efficiency and effectiveness are typical targets that the management of an organization tries to achieve. BPM can be regarded as a management discipline and therefore it is obvious that these kinds of goals are part of the

[5] vom Brocke, J.HKVJH & Rosemann, M. (2010), Handbook on Business Process Management: Strategic Alignment, Governance, People and Culture (International Handbooks on Information Systems) (Vol. 1). Berlin: Springer.

[6] Smith and Fingar, BPM Third Wave, 2002

targets. EPES methodology based on Business Process Management discipline (BPM) [7] will help to create a fully functional business process which can be grouped together in 5 categories, which are often referred to as the BPM life cycle (see Figure 4-1)

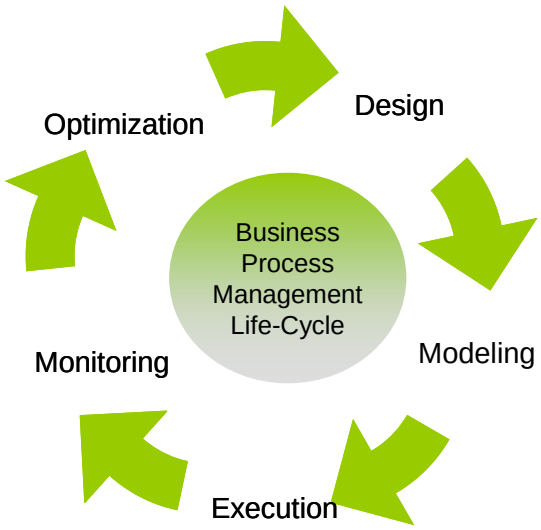

Figure 4-1: BPM life cycle

- Design: The first step contains the activities to define business process. This means that the high level activities are identified, possible organizational changes are discussed, service level agreements are defined and process details such as actors, notifications and escalations are specified.
- Modelling: In this step, the business process is fully specified and validated. The process flow is formalized by using for example BPMN, process variables are defined and candidate services are identified that can be used to execute an activity.
- Execution: The modelled business process is implemented in a business process application often a business process management system (BPMS). Technical details still need to be added to the business process to be able to execute it.
- Monitoring: The processes are monitored for business goals that are defined by key performance indicators (KPIs).
- Optimization: Based on new insights, changing business requirements and monitoring results, the implemented business processes will need to be optimized. When the optimization phase is done, the business process goes into the design phase again and completes the circle.

EPES methodology will be structured according to the target audience:

- EPES RTD Methodology, addressed to the RTD community and it is aimed at the area of eco-optimization (i.e. energy consumption) along production / processes life cycle and will include EPES solutions for challenging S&T problems
- EPES Industrial Methodology aims to provide an approach for industrial practitioners on how to combine advanced technologies to enhance eco-constraints use considerations in manufacturing industry for product/process life cycle management (LCM). It will address both equipment vendors (engineering companies) and users. The methodology has to:

[7] ABPMP, Guide to Business Process Management, Common Body of Knowledge, BPM CBOK, 2da. Edición, 2009.

- Address both technological and organisational aspects related to collaborative work and sharing of knowledge on eco-constraints, PSS and new Services for LCM between agents in the value chain of the Virtual Factory i.e. designers, equipment manufacturers, users and RTD experts.
- Serve as guidelines for companies which intend to apply the new approach to improve their product/process LCM with eco-constraints aspects using EPES components.

The methodology needs to provide theoretical background and guidelines on the following issues:

- How to address Eco-Innovation in product/process
- Organisational / Human issues in a Virtual Factory
- How to set up the EPES Virtual Factory repository
- How to customise the EPES components to specify company's need
- How to integrate the EPES with the other legacy systems
- How to design new services solutions within the LC and new extended monitoring and decision support services in these processes
- User manuals for the EPES components

4.2.1 Structure of the methodology

Based on user's requirements and analysis of state of the art, the following initial structure for EPES methodology is proposed. Some of these methods will be assigned to RTD or Industrial audience in Deliverable 200.1 *Methodology Specification,* which will be the base for the methodology development in T310.

- Methods to set up and maintain collaborative working environment (Business Community) where general information about sustainability, directives, local regulations, or sectorial recommendations can be shared among the different organization members. This topic of the methodology will achieve or improve business agility in an organization because they could charge their process adapting them to regulations, rules, eco-constraints, etc. The pressure is therefore mounting for businesses to align operational processes with the three objectives of sustainable development.
- Methods to model a business process and implement a process execution model following the BPMN 2.0 standard, which will be used for executing workflows in a collaborative fashion and for supporting people to people communication. These methods will comprise an opportunity to bring business and IT close together and will provide guidelines to support the Round-trip engineering (RTE) concept. RTE is a functionality of software development tools that synchronizes two or more related software artifacts, such as, source code, models, configuration files, and other documents. Thus, the application of RTE in EPES platform will avoid misunderstandings between IT developers and business analysts.
- Methods to monitor and optimise a business process in the Business Community developing economic, environmental and social indicators. It will define and give the guidelines to monitor socio-environmental indicators to be integrated into the organisational strategy for assess the degree of sustainability of a business operation. It wants to achieve the concept of Triple Bottom Line. The TBL is an accounting framework that incorporates three dimensions of performance: social, environmental and financial. TBL accounting attempts to describe the social and environmental impact of an organisation's

activities, in a measurable way, to its economic performance in order to show improvement or to make evaluation more in depth[8].

- Methods to set up the EPES Business Community repository allowing the storage of different data. It will explain how to use Business Intelligence tools and methods to generate reports showing analytics on that data that will help end-users for supporting decision making in order to manage a sustainable business lifecycle. It will explore how Business Intelligence models can support the management of sustainability business practices in contemporary firms. Furthermore, it will improve the definition, gathering, analysis and dissemination of socio-eco-financial information among employees, clients, suppliers, partners and community.
- Methods to assess environmental impacts and resource requirements throughout the most important phases of life cycle of products and processes. It will be an integrated concept for managing the life cycle of goods and services toward more sustainable production and consumption.
- Methods to simulate scenarios, where the constrained resource and other resources are selected providing a multi-parameter input in order to achieve a set of solutions to be capitalized with a decision making process that would focus on an optimization of KPI.
- Methods to assess the decision making process of different participants, RTD-s or industrial audience and other interlocutors with a particular Business Process Management to be analysed. After SM runs, it will provide the manner to relate business traditional indicators, with other KPIs related to the organizations environment, social, and economic performance. On the other hand, focusing on the performance of the constraint of this particular business process will be the key to choose the best option among various simulated scenarios, in order to increase the performance of the overall company. Units of constraints consumed in a period of time is the base for positioning the performance of teams in a factory, the performance of a machine or the performance of a mix of interrelated installations.
- Methods to create a service generator field where simulation output, decision making output, and actual and historical information, will be the base in order to generate adequate service to each kind of organization. The KPIs indicators, when implemented, will show the values of an improved organization performance when all steps of EPES system are carried out systematically and services are tested and appears to respond to the organization needs.

4.2.2 Organisational / Human issues in a Business Community

The part of the methodology dedicated to organisational / human aspects will include specifically guidelines on:

- How to organise collaboration among actors in Value Chain (equipment/installation manufacturers/engineering companies and users) of Business Community in order to assure optimal product/process LCM regarding eco constraints and optimal environmental impact, addressing also new business models (new services) allowed by such collaborative design approach.
- How to organise work within Business Community in general
- How to address Intellectual Property Rights (IPR) issues

Special attention will be paid to IPR issues at Business Community (Collaborative Space) in an Organisation keeping in mind that companies working in a network environment have to be

[8] Elkington, J., 1998. Cannibals with Forks: The Triple Bottom Line of 21st Century Business. New Society Publishers, Gabriola Island.

aware of the need to share data, documents, information and knowledge that could be protected under IPR. The methodology will provide IPR Policy Guidelines for Business Community collaboration consisting of several paragraphs, describing in detail the most important issues related to intellectual property, such as:

- Basic definitions and their description used in IPR Policy
- Rules of participation in a Business Community (Collaborative Space), addressing usage of the pre-existing know-how which partner brings and rules of exclusions from specific pre-existing know-how
- Knowledge ownership and description of rules of knowledge access
- Rules for publication and dissemination of knowledge
- Copyrights, licensing and patenting of knowledge developed within EPES Business Community and rules of utilising trademarks or service marks

If some intellectual property issues will be not covered by the above described Intellectual Property Rights Policy the national or international law will have the force to regulate them.

5 Virtual Collaborative Network

5.1 Overview

The Virtual Collaborative Network (VCN) module provides EPES end-users (non-experts) with a main point of access to the whole EPES ICT system. Simulation experts and developers will require other dedicated access points provided by other EPES modules. One of the main roles of the VCN module is to set-up a networking infrastructure that enables the distribution of the users into groups and the capturing of users' knowledge for later reuse. The main group is the *Business Community*, which involves and contains all the actors within the Virtual Factory domain, such as users, suppliers, clients, governmental agencies and RTD partners. This VCN module will allow, based on the identification of business optimization opportunities or necessities, the creation of *Collaborative Spaces* out of the *Business Community* to carry out the identified opportunity. A *Collaborative Space* represents a specific goal-oriented group aimed at, e.g. the collaborative identification of sustainability constraints and environmental objectives that would shape the organizations' business processes (modus operandi) towards sustainability. To enable this capability, the VCN module will provide the organizations with Enterprise Content Management features, which include business process design, modelling, execution, monitoring and optimization. The business process execution will be supported by workflow engines and the process monitoring and optimization will be provided by the Decision Making and Simulation modules functionalities.

5.2 Structure

Figure 5-1 below presents the VCN module structure. The VFKDB (Virtual Factory Knowledge Database) contains a common public data section and several private sections. The *Business Community* generates and consumes data that belongs to the public domain, while the *Collaborative Spaces* deal with private data and can also generate and consume public domain data. For instance, within EPES project consortium, we can consider the different project partners as belonging to the *EPES Business Community*, which provides a cooperation space where general information about sustainability directives, local regulations or sectorial recommendations can be shared among the different organization members. The aim of the *Business Community* is the identification of optimization opportunities based on the network actor's knowledge. Once an opportunity is identified, some of the *Business Community* organizations can rearrange themselves in a *Collaborative Space*, where the optimization of a particular business process is performed. Within EPES project domain, we can consider each of the Business Cases as a *Collaborative Space*, where the organizations that belong to that space can share private data for the development of the optimization opportunity.

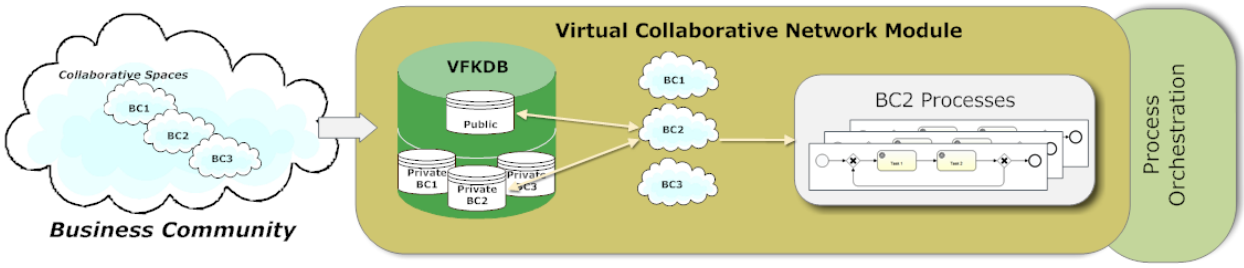

Figure 5-1: VCN Module structure

Figure 5-2 exemplifies the creation of the *BC2 Collaborative Space* out of the *EPES Business Community* in order to carry out a business optimization opportunity, that is to say, the implementation of power grid control systems for improved identification of maintenance needs and improved monitoring of grid load and safety limits. The outcome of this *Collaborative Space* is

the identification of constraints and objectives, which will shape the organizations business process towards sustainability with the contribution of EPES modules.

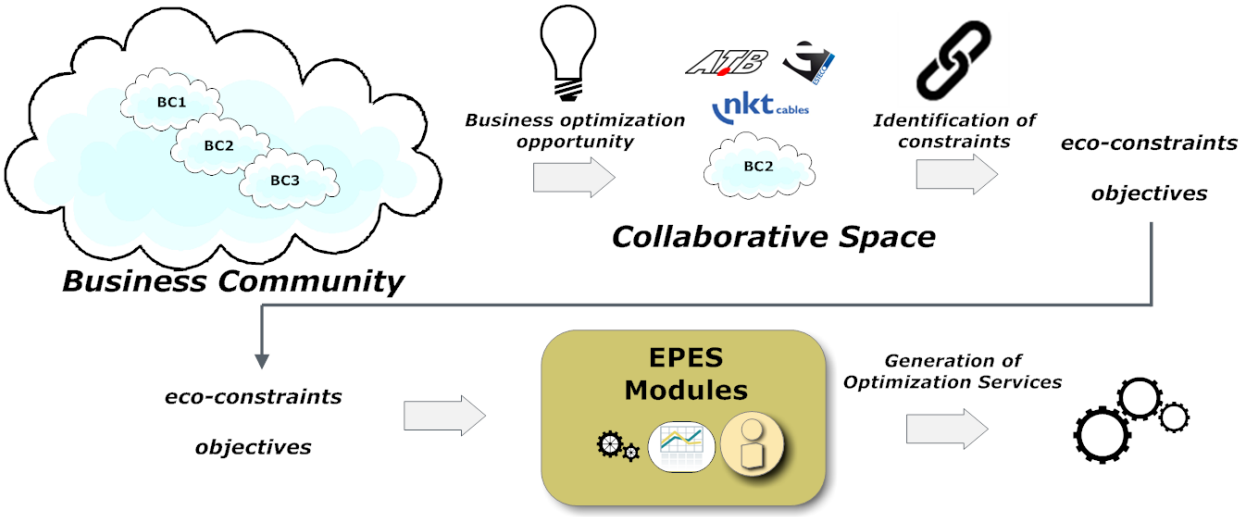

Figure 5-2: BC2 Collaborative Space creation

5.3 Interfaces

The VCN module will interact with the rest of the EPES modules and with the end-users via interfaces. The users will access the functionalities of the VCN through Graphical User Interfaces (GUI), while the communication of the VCN with the rest of the modules will be done through APIs. The following interfaces are envisaged:

- User interfaces (GUIs) to access the VCN functionalities. The VCN will have one or more use interfaces to access the collaboration functionalities, the content management functionalities and the workflow engine capabilities. On the other hand, the VCN module will render the Simulation Module and Decision Making Module dedicated GUIs, making use of the Services Generator Module GUI aggregation capabilities, as shown in Figure 5-3
- Interfaces to access and consume the high level Business Services composed by the EPES Modules, e.g. using BPEL through BPMN
- Interfaces that will let the EPES modules remote access to the VFDB data model, e.g. based on SOAP or RESTful web services

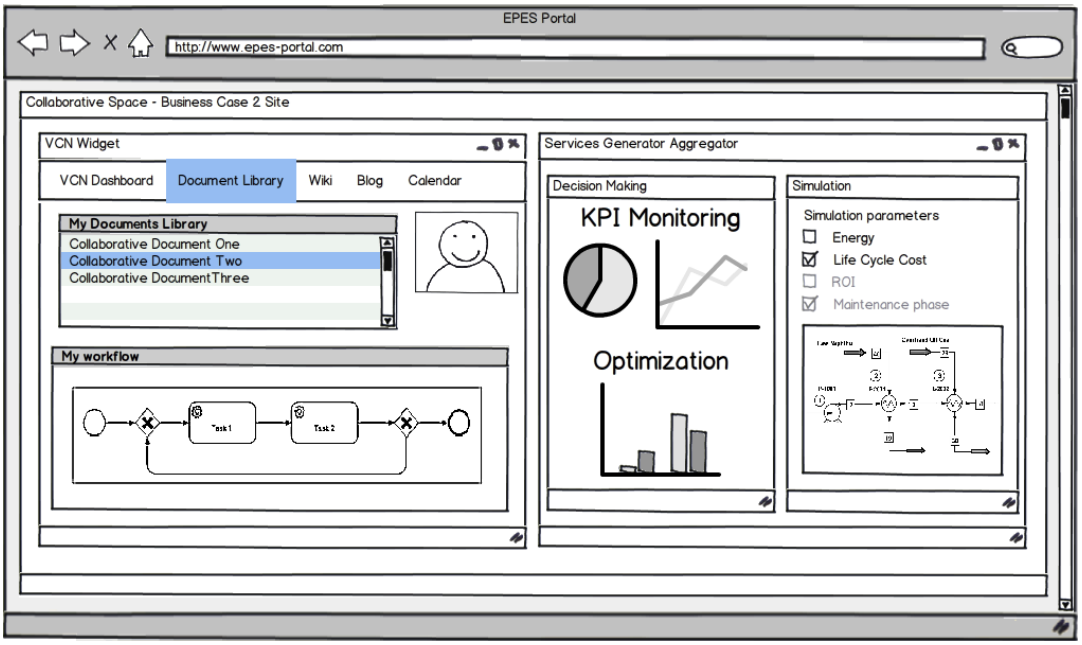

Figure 5-3: VCN Module user interface

5.3.1 Inputs/Outputs

In order to identify the generic inputs/outputs of the VCN, the BC-specific inputs/outputs have been analysed first, in section 5.3.1.1. Then, in section 5.3.1.2, the generic inputs/outputs have been abstracted.

5.3.1.1 Inputs/Outputs of the Virtual Collaborative Network in the three Business Cases

5.3.1.1.1 Inputs/Outputs of the VCN in BC1

The VFKDB will contain the constraints generated by the Collaborative Spaces and data coming from PRISMA CMMS, such as material resources, human resources and maintenance records (maintenance works carried out and short-term planned maintenance tasks). In addition, technical data coming from wind turbines SCADA systems will be aggregated to the VFKDB. This technical data may be used as a support for the early failure detection and prognosis of wind turbines, i.e. the discovery and prioritization of maintenance tasks

Other documents that may include useful parameters, such as resources availability and capacities required for the configuration and execution of the scheduling optimization tasks, will be handled by the VCN Content Management capabilities

Inputs:

- Workflow definitions to design and model TER maintenance business process

Outputs:

- Execution of the optimized maintenance process workflow enabling the use of simulation and optimization techniques to end-users (non-experts)
- Monitoring of the optimized maintenance process to allow its further optimization

5.3.1.1.2 Inputs/Outputs of the VCN in BC2

For the BC2, the interaction with the VCN will have the purpose of storing and retrieving parameter sets for different cable types.

5.3.1.1.3 Inputs/Outputs of the VCN in BC3

For the BC3, the interaction with VCN will have the purpose of storing and retrieving parameter sets for conceptual product manufacturing assessment with simulation. Other documents that may include generic or common useful parameters, such as resources availability (work-shift, factory calendar, etc.), resource energy consumption data, energy carrier data, and other specific data required for the configuration and execution of the simulation tasks, will be handled by the VCN Content Management capabilities.

5.3.1.2 Generic Inputs/Outputs of the Virtual Collaborative Network

Inputs:

- Objectives and eco-constraints will be captured from the environment through the *Business Community* network and *Collaborative Spaces*
- Field data (actual), historical data, KPIs and Life Cycle Inventory data will be captured from existing systems
- Workflow definitions to design, model, execute and monitor the Business Processes

Outputs:

- Objectives and eco-constraints to be used by the Decision making module so as to assess the need for sustainability improvement
- Implementation of the Business Process Management approach, which allows:
    - the continuous assessment of the need for sustainability improvement via "trustable" research networks focused on Sustainability Intelligence
    - the real implementation of business process changes through simulation and optimization techniques
    - the measurement, evaluation and improvement of the adapted business processes

5.4 Functionality

The main functionalities of the VCN module are:

- Provision of a main point of access for the EPES end-users (non-experts) through user interfaces
- Capabilities for supporting the aggregated GUIs provided by the Service Generation Module
- Technical infrastructure for the distribution of users into groups. The Business Community represents the main group that contains the Collaborative Spaces, which are goal oriented sub groups aimed at the optimization of one phase of a particular product life-cycle. The Business Community involves all the stakeholders in a sector and contains public data (members profiles and competencies), while Collaborative Spaces involve only some stakeholders in the value chain who collaborate in a particular optimization problem. The main characteristic of these Collaborative Spaces is that they represent the entry point for the use of the different EPES modules functionalities, since the data involved in the simulation and optimization is normally private
- Collaborative features through web 2.0 mechanisms, such as wikis, blogs, shared document libraries, calendars, message boards and users' activity tracking systems
- Content management and sharing (documents and web content)

- Workflow engine for contents production, consumption and transformation through the design, modelling, execution and measurement of business processes. The use of workflow capabilities will let end-users, on the one hand, capture and reuse knowledge for improving future products design and, on the other hand, connect the business processes to high level business services composed by the Service Generator Module or other EPES Modules
- Public and private VFKDB, which will contain actual data (allowing Business Analysis Monitoring), historical data (enabling Business Intelligence analysis), identified constraints and objectives from the collaborative networks, KPIs and Life-Cycle Inventory data. The database will be remotely accessible by all EPES modules for supporting CRUD operations

6 Service Generator

6.1 Overview

The Service Generator (SG) is the module responsible for adjusting the configuration of deployed services to reflect changing conditions (though the SG's *Configuration Services*) and deploying the changed configurations (through the SG's *Cloud Services*).

The SG's methods are exposed as web services, in order to allow programmatic access to them. However, the user can also interact with the SG through the *Aggregator/Cockpit UI*. This is a dashboard-style user interface which enables interaction with the SM/DMM exposed interfaces and allows analysing contextual information and previously taken decisions in order to update an existing configuration and use it to reconfigure a particular deployed application-specific service.

The Context Extractor is responsible for identifying changes in the context of the environment. The current identified context will be used to extract available context knowledge. The results of the Context Extractor will be used for updating the system behaviour. The purpose of the EPES context model is to define a fundamental data model for context extraction. In EPES the context model will be mainly used to model the context. So the context model will not try to bring full description of context, but to index context to help to identify context. The EU projects K-NET[9] and Self-Learning[10] will be the guideline for context extraction, interpretation and modelling in a flexible and distributed environment.

6.2 Structure

Figure 6-1 below presents the SG in the context of the envisaged EPES solution. The SG is expanded to show a high-level view of its internal structure.

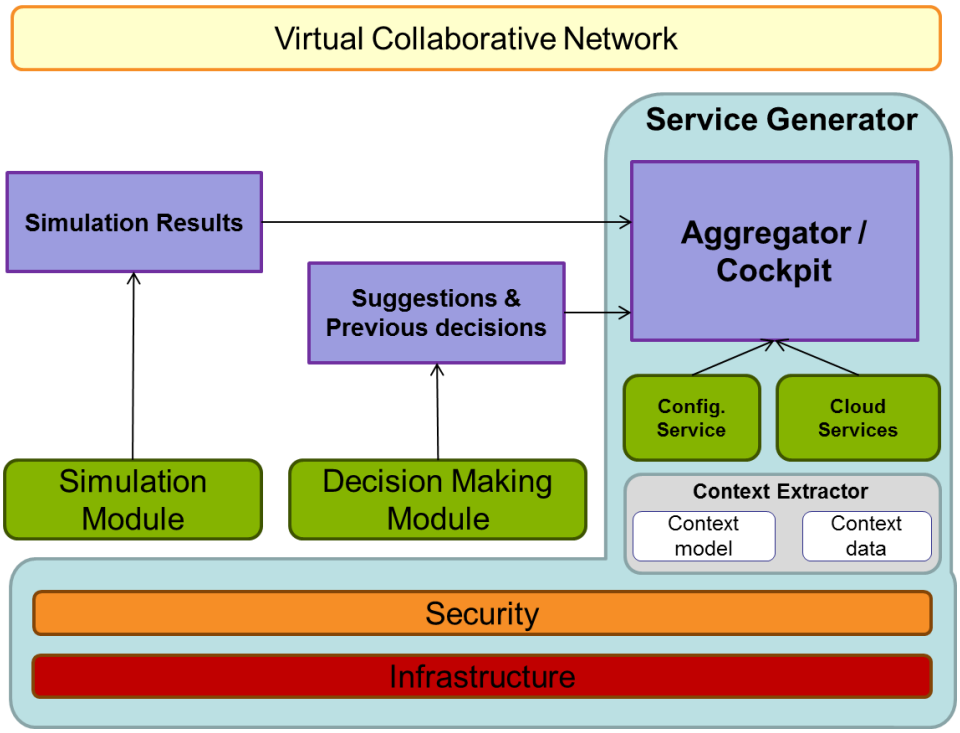

Figure 6-1: Service Generator Module in the EPES context

[9] K-NET project: Service for context sensitive enhancing of knowledge in networked enterprise, *FP7-215584,* http://www.k-net-fp7.eu/

[10] Self-Learning project: Reliable Self-Learning Production Systems Based on Context Aware Services, FP7-NMP-2008-228857, http://www.selflearning.eu/

The SG's main components are the following:

- **Configuration services:** allow the user to modify configurations for existing services.
- **Cloud services:** allow the user to deploy the modified service configurations.
- **Aggregator/Cockpit:** the GUI of the SG; offers a graphical overview of the SG and integrates the GUIs of the SM/DMM.
- **Context Extractor:** extracts the user's context in order to refine the SG's outputs. The context is extracted on the basis of the context model and the gathered context data:
    - **Context model:** used to model the context of the user's interaction with the EPES solution.
    - **Context data:** data gathered in order to detect a user's context.

6.3 Interfaces

The SG module interacts with the following modules of the EPES solution (also see Figure 6-2 below for a more detailed illustration of the various interaction paths):

- Decision Making Module
- Simulation Module
- Virtual Collaborative Network

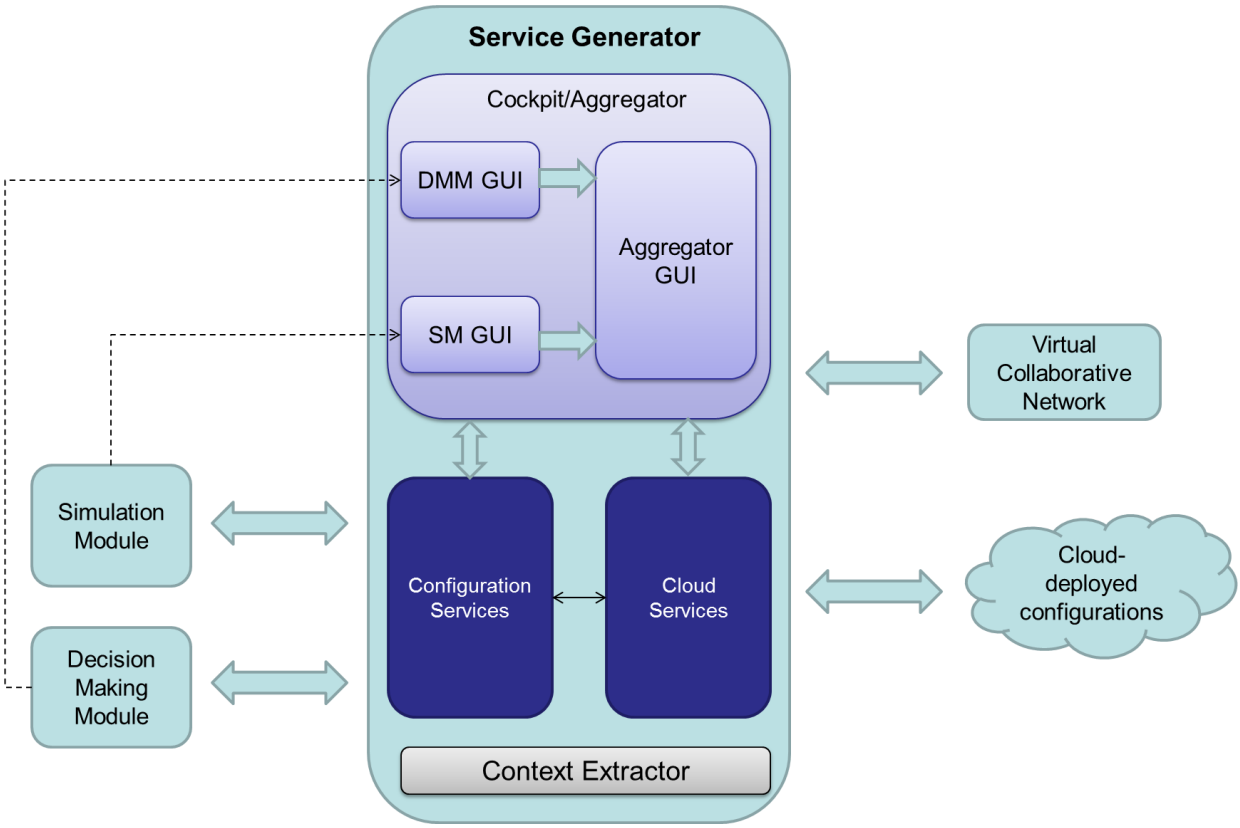

Figure 6-2: Interfaces between SG and other EPES modules

The data passed over the interfaces between the SG and these modules are presented in the next section.

6.3.1 Inputs/Outputs

In order to identify the generic inputs/outputs of the SG, the BC-specific inputs/outputs have been analysed first, in section 6.3.1.1. Then, in section6.3.1.2, the generic inputs/outputs have been abstracted.

6.3.1.1 Inputs/Outputs of the Simulation Module in the three Business Cases

6.3.1.1.1 Inputs/Outputs of the Service Generator in BC1

Input from VCN:

- Read parameters for running the optimization

Output to DMM:

- Configuration of the optimization algorithm

6.3.1.1.2 Inputs/Outputs of the Service Generator in BC2

Input from Simulation:

- Predicted cable temperature
- Predicted cable load
- Cable Hotspot detection
- Grid load (several cable sections combined)

Input from DMM:

- Previous cable load (balancing) decisions
- Maintenance prediction in single cable and in grid
- Provision of “best” suitable algorithm

Input from VCN:

- Parameter Sets for different cable types

Output to Simulation

- Configuration

Output to DMM

- Configuration

Output to VCN

- Parameter Sets for different cable types

6.3.1.1.3 Inputs/Outputs of the Service Generator in BC3

Input from Simulation Module configuration and discovery services to the SG configuration services:

- Simulation tool properties (metadata)
- Simulation model properties (metadata)

Output to Simulation Module from the SG:

- Selected simulation tools and specific configuration parameters

- Manufacturing simulation data and models to be simulated
- Run parameters

Output to VCN

- Service configurations and related simulation parameters

Input from VCN

- Manufacturing simulation model repository (equipment, tools, human resources, processes, factory calendar, bill of materials, etc.)
- Previously saved service configurations and related simulation parameters

Output to DMM

- Decision-making process models
- Output constraints, optimisation variables
- Acceptable values, target values

Input from DMM

- Decision-making task/service descriptions (metadata)
- Optimization tool property descriptions (metadata)
- Visualization and reporting configuration property descriptions (metadata)
- Optimization results, line balance, number of resources

6.3.1.2 Generic Inputs/Outputs of the Service Generator

By abstracting the BC-specific inputs/outputs presented in sections 6.3.1.1.1, 6.3.1.1.2, and 6.3.1.1.3, the following generic inputs/outputs have been established for the SG:

Inputs:

- Predictions
- Simulation metadata
- Decision making metadata
- Existing service configurations
- Previous decision history
- Simulation results

Outputs:

- Simulation Module Configurations
- Decision Making Module Configurations
- New / Updated services configurations

6.4 Functionality

The SG's functionalities are split in several separate modules. The functionalities of these modules are presented in the following sections.

6.4.1 SG Configuration Services

The SG Configuration Services allow the user to reconfigure existing services in order to maintain their relevance and react timely to changes.

These configuration services represent the data access layer for the SG's interactions with the SM and the DMM. They have interfaces for communicating and passing data between the SG Cockpit/Aggregator and the SM/DMM; this data is used to update existing configurations which are deployed and used by BC-specific services.

6.4.2 SG Cloud Services

The SG Cloud Services allow gathering data and retrieval and deployment of configurations for BC-specific services. The user can update these configurations through the GUI of the Cockpit/Aggregator, by using data retrieved from the cloud (by the SG Cloud Services) and from the SM/DMM (by the SG Configuration Services).

6.4.3 Cockpit/Aggregator

The Cockpit/Aggregator gives the user an overview of the SG's interaction with the SM and DMM (through the SG Configuration Services), as well as with the resources deployed in the cloud (through the SG Cloud Services). Using the Cockpit/aggregator GUI, the user can compose new configurations for running services, or edit existing ones.

This component of the SG aggregates, on one hand, the specific GUIs of the SM and the DMM. This aggregation is supported by the SG Configuration Services, which act as data access layer between the SG and the SM/DMM. On the other hand, the Cockpit/Aggregator aggregates the GUI needed to interact with configurations deployed in the cloud. At the data level, the SG Cloud Services take care of the connections needed for this.

6.4.4 Security

The security module is a Java based authentication and authorization framework. From the authorization point of view, the security module provides functionality to manage users, groups and entities and define a fine-grained permission hierarchy. From the authentication point of view, the security framework provides the functionality to authenticate users with existing credential repositories and to use the provided authentication mechanisms (NT, Unix via PAM, DBMS).

6.4.5 Infrastructure

The Service Infrastructure is a key enabler of a Service-Oriented Architecture, a framework capable of managing all the operational aspects related to Web-Services. It can be seen as an execution environment which enables discovery, selection, mediation, and invocation of Web-Services. To build the services and operate the infrastructure, the following (existing) software, such as JBoss, Tomcat, Axis and jUDDI, is used.

6.4.6 Context Model

The Context Model contains ontology based generic models which are most appropriate for all three business cases. This generic repository can be extended by specific concepts, necessary for each business case. The models are shared by different software components at run time. The Context Repository allows update and storage of extracted/processed contextual information for later retrieval. Information flow among the modules is event driven in some cases and time based in other cases.

6.4.7 Context Extractor

Context Extraction is based on set of services responsible for identifying changes in the context of the environment. The current identified context is used to extract available context knowledge. The results of the Context Extraction are used for updating the system behaviour.

7 Decision Making Module

7.1 Overview

The decision making module (DMM) supports and guides the user in the decision making process. It helps to analyse and process information gathered from the life cycle of the product and compare it with the output of the simulation system which provides the "should be" performance metrics.

Figure 7-1 describes the scenario in which the DMM operates: it receives business process definitions, structured data, raw data sets, KPI, and possible other inputs and with the interaction of a human expert it produces transformed data, graphic analysis, updated business process, and the best solution from among a set of reasonable alternatives.

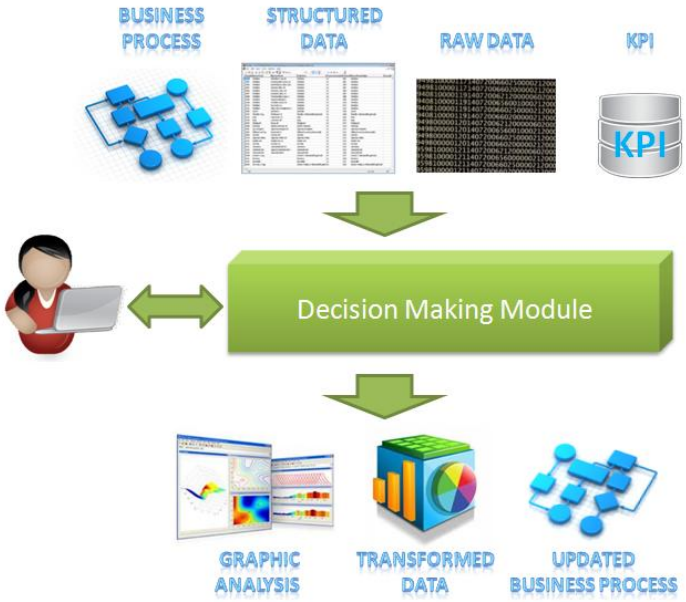

Figure 7-1: The DMM scenario

The best solution is directly related to the constraint or limitation of the selected business process. The scheduling of teams, the production mix or the configuration of a system, in order to maximize the organizations performance, is very important to achieve a good decision making module. This factor is necessary but not sufficient, because the production mix and the configuration of installations or teams must be strongly related to the constraints of the system itself.

When a process is selected, the constraints must be located, and in order to locate them, the process must be wide enough to reach good conclusions and not sub optimize the entire process within EPES system platform.

7.2 Structure

DMM is an interactive computer-based system intended to help decision-makers to use data and models to identify and solve problems and to make decisions. The tasks, which can be very complex, are based on repeated interactions of a human expert with a set of "decision making tools". The most relevant methodologies, algorithms and tools are described in section 2.8 of deliverable D100.1. DMM supports a continuous upgrading process following the concepts of TOC.

The tasks of supporting the selection and screening for critical paths and bottlenecks in eco product/process development have a high level of complexity, which coupled with the large

amount of data, require leverage all the multi objective optimization and multi criteria decision methodologies proposed in the D100.1 deliverable.

This aspect becomes even more critical for the automated product/process engineering approach proposed in the project, where large amount of simulation data will be produced in order to explore as much as possible the space of all possible solutions. The multi objective analysis and optimization procedure is deemed to improve the eco characteristics (e.g. reduction of consumptions, wastage and energy usage) of all the potential product/process solutions under exploration, while the decision support techniques should drive the screening phase on the set of the available designs in order to extract the final candidates (e.g. the most reliable or suitable for specific markets).

Figure 7-2 describes the components of the DMM. The DMM tasks concerning a business process usually start from an input formal model (e.g. a workflow). By using the GUI, the DM expert can perform one or more of the following tasks:

1. Enrich the model with one or more KPI variables, to be used as:
    - Output for monitoring purposes.
    - Constraints for simulation purposes.
    - Targets for optimization purposes.
2. Define one or more DoE, call the simulations, and perform the statistical analysis.
3. Select the optimization algorithm and call the optimization. The optimization algorithm is related to the constraints within the process selected for a particular typology of organization.
4. Analyze the data through:
    - Data visualization.
    - Post processing (charts, cluster).
    - Geographic information system (GIS) visualization.
    - Data mining.
    - Dashboards.
    - Extract, transform and load (ETL).
    - Online Analytical Processing (OLAP).
5. Populate the data warehouse.

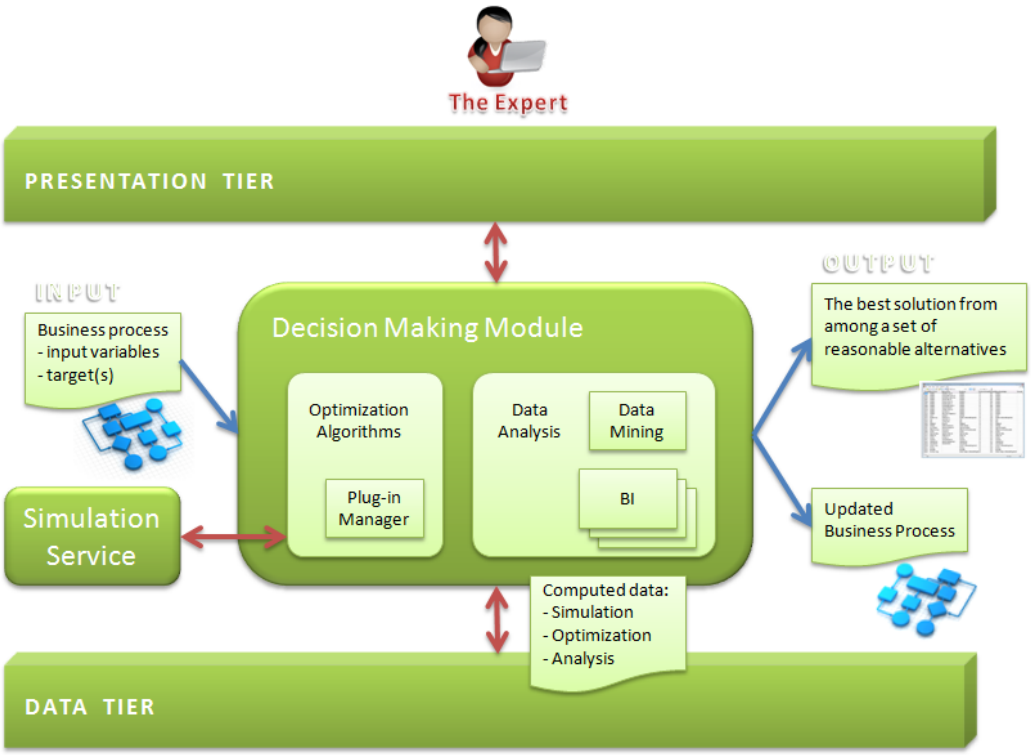

Figure 7-2: The DMM architecture

7.3 Interfaces

The DMM interacts with human experts and software components. The interfaces are detailed below:

Human expert(s): the interaction is through a GUI defined in the presentation tier, which must allow interaction with a heterogeneous set of tools.

End user(s): the interaction is through a GUI defined in the presentation tier, which must provide a set of well-structured and easy to use, well-known graphic tools.

Simulation Module: the DMM interacts with the SM for simulations and optimizations tasks. The calls can be synchronous or asynchronous and the data needed can be passed directly between the components or through the data tier.

Data tier: it must support read and write data operations from a heterogeneous set of databases.

The Plug-in manager in the Algorithms component offers a common interface for connection to external systems with specific interfaces implemented case by case in order to support addition of new algorithms.

7.3.1 Inputs/Outputs

In order to identify the generic inputs/outputs of the DMM, the BC-specific inputs/outputs have been analysed first, in section 7.3.1.1. Then, in section 7.3.1.2, the generic inputs/outputs have been abstracted.

7.3.1.1 Inputs/Outputs of the Decision Making Module in the three Business Cases

7.3.1.1.1 Inputs/Outputs of the DMM in BC1

In BC1, the DMM will be used to calculate to-be KPIs aimed to fulfil a constraint (e.g. increase wind turbines availability by a given percentage) and to obtain optimization results, such as:

- Optimal scheduling of maintenance tasks taking into account external restrictions, such as forecasted weather conditions or hired equipment availability
- Assignment of the scheduled maintenance tasks to the available maintenance teams according to their capabilities and availability
- Dispatching of the maintenance teams to the different farms based on efficiency criteria

The DMM (or the SGM) will expose the optimization algorithm as a Business Service (e.g. as a web service) to the VCN workflow engine.

7.3.1.1.2 Inputs/Outputs of the DMM in BC2

In BC2, the DMM will be used for supporting the following decisions:

- Previous cable load (balancing) decisions
- Maintenance prediction in single cable and in grid
- Provision of "best" suitable algorithm, taking into account the maximization of "unit of service" related to actual installations

In order to compute these results, the DMM will receive configurations of BC2-specific services as inputs.

7.3.1.1.3 Inputs/Outputs of the DMM in BC3

In BC3, the DMM will be used for comparison of product and related manufacturing scenarios and to supporting the following decisions:

- Selection of best product design concepts from the manufacturing performance and sustainability perspective based on calculated KPIs,
- Selection of optimum manufacturing facility based on assessment of productivity and sustainability KPIs
- Selection of the best production mix related to KPIs and other traditional indicators

In order to compute these results, the DMM will receive configurations of BC3-specific services as inputs:

- A BC3-specific decision-making process model (workflow) and configuration
- A selected manufacturing simulation model and related input data
- Optimisation parameters and variables, constraints
- Acceptable and target values

Inputs from the simulation module runtime services:

- Simulation results
- Values of KPIs: Production efficiency, equipment utilisation, throughput, etc.
- Environmental KPIs: Sustainability, energy, CO2 emissions and hazardous waste

7.3.1.2 Generic Inputs/Outputs of the Decision Making Module

The inputs of DMM are:

- One or more business process definitions specifying input variables, target(s), etc. E.g.: a workflow. When the restriction of the system is found, the wider the scope of the process

is, the easier will be to find the overall organization performance constraint that limits the throughput of the company.

- The DM expert directives to be used to judge the results, choose between alternatives, and/or decide to refine the analysis. The quality of the decision is closely related to both: the quality of the results obtained from simulation/optimization, and the quality of the skills of the expert(s) involved.
- All the useful data from data tier (field data, historic data, simulations and optimizations data, previous analysis data, KPI, SI, etc.).

The outputs are both the best solution from among a set of reasonable alternatives, and an enriched business process. All the simulation/optimization results and the analysis results can be made persistent through the data tier.

7.4 Functionality

The main features of the DMM involve:

- Set up the simulation and optimization:
  - Define the design of experiments (DoE).
  - Define constraints, targets (also bound to KPIs).
  - Choose the optimization algorithm.
  - Take in account the concepts of theory of constraints (TOC): locate constrained resource in the overall process.
  - Make sure that the correct extent (scope) of the process is selected to identify the constraints.
- Call simulation and optimization through Simulation Module.
- Data analysis and reporting through Business Intelligence (BI) analytics:
  - Extract, transform and load (ETL).
  - Online Analytical Processing (OLAP).
- Data analysis through graphical tools:
  - Data visualization.
  - Post processing (charts, cluster).
  - Geographic information system (GIS).
- Support addition of new algorithms:
  - Local through scripting language
  - Remote (call to external tool)
- Aggregate information in the right format for building the data warehouse.
- Wrapping specific DM functionalities into (low level) services, make available its API to the Services Generator module.

Furthermore, the below features are considered relevant:

- It is required the interaction with the DM expert through a dedicated GUI.
- It is required the interaction with the data tier:
  - Field data.
  - Historic data.
  - Computed data (simulations, optimizations, analysis).
  - Key Performance Indicator (KPI).
  - Sustainability Intelligence (SI).

- Objectives / Constraints.
- May be useful to exploit the collective intelligence:
  - Use of social tools to interact with other experts.
  - Interact with external DB or other source of structured knowledge.

8 Simulation Module

8.1 Overview

Simulations and other numerical analyses are an essential part of the Eco-Process Engineering System. The simulation module of the EPES platform provides a capability for running numerical analyses related to the life-cycle assessment process. It provides simulation services through an abstract service interface that allows the higher level components of the EPES platform to use simulation as an interchangeable service, according to the principles of the service-oriented design paradigm.

Through the service interfaces of the SM, the other modules are able to configure and execute simulation models without any dependencies on the nature or the data representation of the simulation models. It allows the other modules to present configuration options and numerical parameters of simulation modules without any prior knowledge of the internal structure of the model.

The Simulation Module (SM) achieves this abstraction of simulation models by providing the tools for a developer to encapsulate the functionalities of third-party simulation tools into pluggable software components that can be manipulated through unified interface interfaces, and presenting the simulation tool features to the higher-level modules through an abstract service interface.

Figure 8-1 presents the functional dependencies between the modules of the EPES architecture. A user interface in the Presentation Tier is presented to the developers for configuring and scripting the external simulation tools. The simulation models and simulation results are stored in a database in the VFKDB or legacy systems in the Data Tier. Metadata that presents the configuration options and numeric parameters of simulation models are communicated to the Service Generator. Parameter values and configuration options for executing simulations are received from the Decision Making Module.

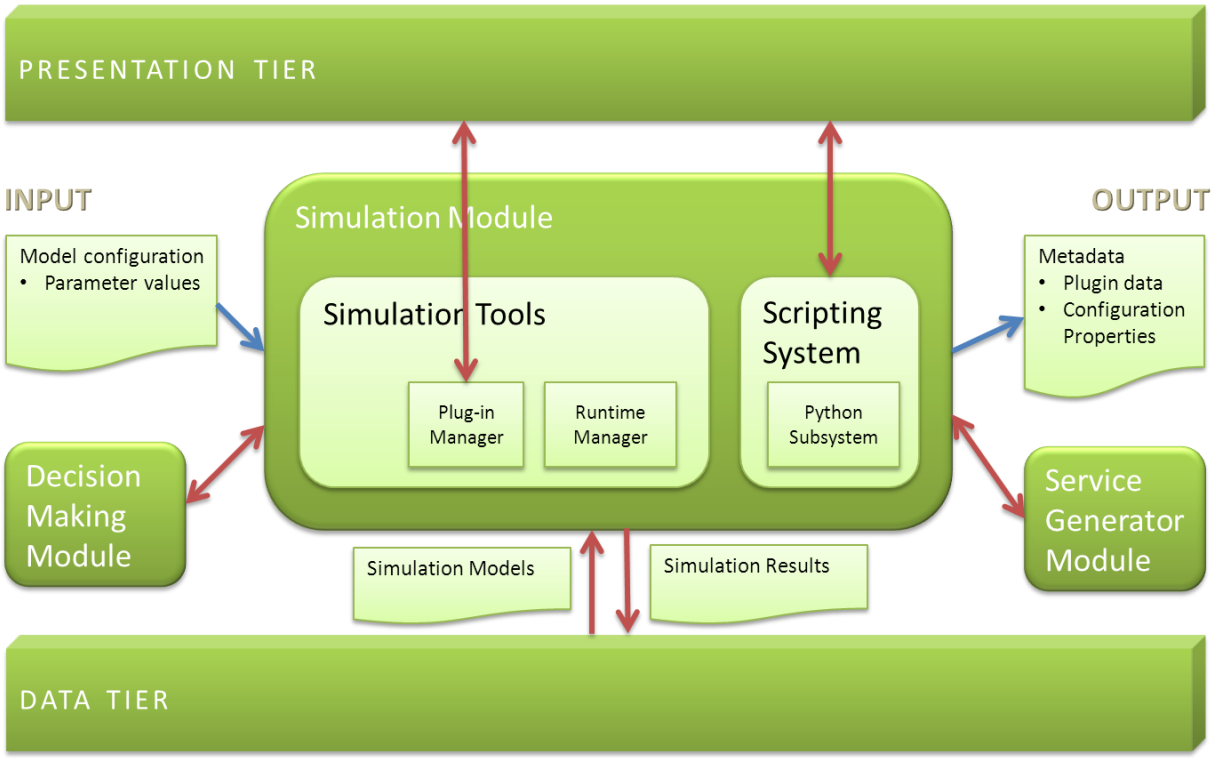

Figure 8-1: The Simulation module architectural concept

8.2 Functionality

The Simulation module provides functionality for running individual simulations using a simulation model and a model configuration that are provided by other modules in the system. The simulations (or other numerical analyses) can be performed either directly by a pluggable software component (plug-in), or by executing a third-party software code. The third-party software code must be able to run without user intervention. No user interface for the simulation tools are provided to the end user.

All analysis processes are performed by pluggable software components. The Simulation module provides services for the plug-ins to facilitate easy customization and configuration of the plug-ins. It must also provide all the information that is needed by the plug-ins for the execution of the integrated tools.

The simulation module includes a scripting functionality that allows the developer to provide a part of the functionality in a plugin as an external script that allows a model execution to be more easily customized. Python is a widely used and supported tool for processing such user-generated code and is easy to integrate into many programming language environments.

The services provided by the simulation module only handle a single simulation process at a time. When a complex analysis, that requires the use of several simulation tools, is needed, the "choreography" of the individual simulation tasks is performed by the caller of the simulation module.

Optimization tasks are different from ordinary simulations tasks in that they involve several runs of a single simulation code with different input values. The simulation module does not directly support any optimization methods. These, and other statistical analyses, are performed by the decision making module, by repeatedly calling the simulation module's services with varying parameter values.

8.3 Structure

Figure 8-2 presents a suggested outline of the internal submodules of the Simulation module. The implementation is divided into five modules.

The simulation execution service provides the interface to the other major modules of the EPES system. It receives requests for simulator execution from the Decision-Making module.

Each plug-in is implemented as a separate module. It provides an interface for the simulation execution service that allows it to perform pre-processing, execution and post-processing tasks. Each plug-in also provides interfaces for the management of simulation model properties.

The execution runtime service modules provides interfaces that allow the other modules to set up the execution environment for the external code, including the creation of file system structures that are needed by the external code.

The script execution service implements a scripting system that allows the plug-in code to refer to a user-supplied script to perform some of its functionality.

The model data management module provides access to all the simulation model data that is needed by all the other modules.

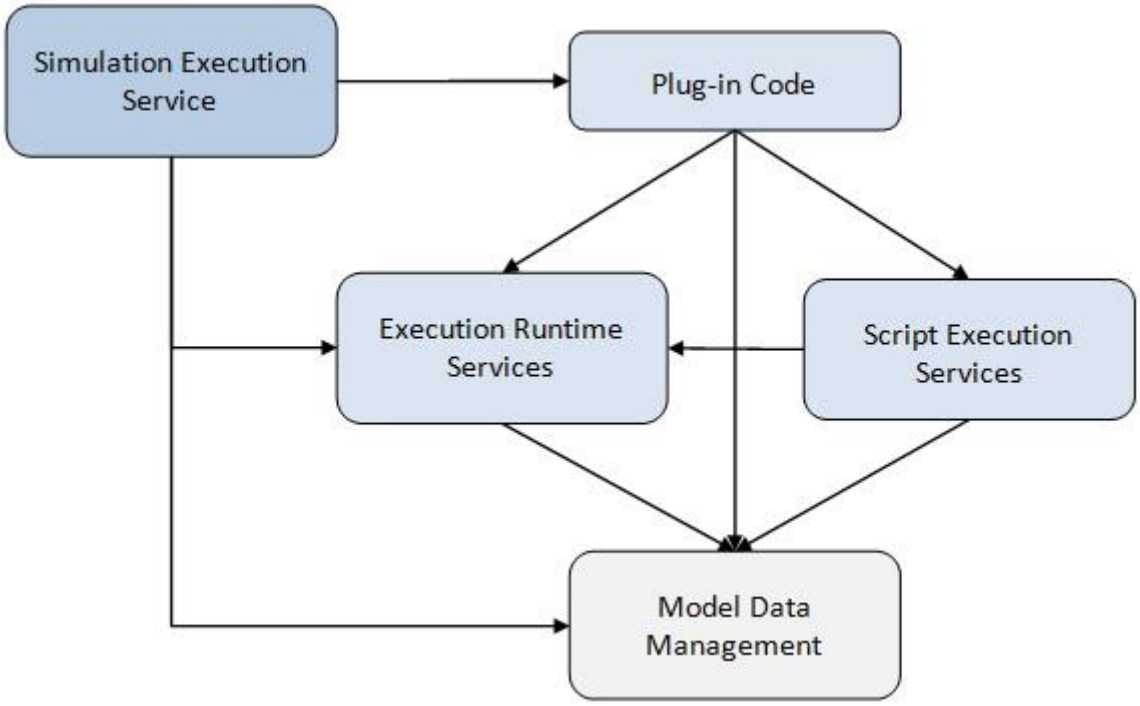

Figure 8-2: Functional dependencies between the internal submodules of the Simulation module.

It is possible that a common submodule for handling the external codes involved in both simulation and decision making processes is developed in order to minimize unnecessary repetition of development resources.

8.3.1 Use of existing platforms

Many of the intended features of the EPES modules are already supported by commercially supported tools. Simulation integration features are provided by for example:

- SOMO and modeFRONTIER by Esteco
- ModelCenter by Phoenix Integration Inc.

SOMO is still at a beta-testing stage, and is not commercially available. It provides functionalities for simulation integration in a web-based environment. The possibilities of making use of its existing functionality in the implementation of the early and full prototypes of the EPES platform must be considered.

modeFRONTIER is a multi-disciplinary and multi-objective optimization tool that also provides features for simulation tool integration. It has a graphical scripting system for creating simulation workflows and provides a plug-in architecture for connecting third-party simulator codes into the workflows.

ModelCenter is a software package from Phoenix Integration Inc. for simulation integration. It provides a plug-in architecture for simulator integration, graphical simulation process modelling tools and support for tool scripting with various scripting languages.

The possibilities of making use of any existing software tools in the development of the EPES platform must be researched to avoid unnecessary duplication of work. It must be determined, if the simulation module can be implemented on top of ModelCenter or modeFRONTIER. Both of them provide a plugin architecture or external code execution.

The features of these commercial tools also cover at least some of the functionalities of the Decision Making and Service Generation modules. One possible approach is to implement the whole

EPES platform on top of an existing solution, in which case the project work could be more focused on integrating the tools that are required for the specified business cases, including the product life cycle analysis tools and data sources.

8.4 Interfaces

The interactions between the Service Generation module and the Simulation Module (also see Figure 8-1) include communication of data regarding the available external simulation tools and the simulation models that are created by the user of the EPES solution. The nature of this interaction is mostly asynchronous and indirect, consisting of messages about updates to the contents of the common data model, which should probably be facilitated through a database.

The interactions between the Decision Making Module and the Simulation Module include data regarding the selected simulations tools and models and the input and output parameter data that is passed between the two modules. The Simulation module is called by the Decision Making Module in a synchronous nature that resembles a procedure call or a request-and-response exchange.

The Decision Making Module is responsible for the selection of input data for a number of simulation runs, according to the needs of the optimization or analysis tasks that is to be performed.

A simulation model consists of data that describes the system to be simulated, and a computer code that is able to process the model in order to produce the required output values. The computer code part of the simulation models are provided through the plug-in architecture. The data part of the model is provided in either a neutral or code-specific format.

The simulation module should be as independent as possible of the data sources from which the input parameters are acquired from. The simulation module needs to have access to data that specifies the model to be simulated and the initial values for all of the properties of each entity in the model.

An interface shall be defined to abstract all data access between data sources and the simulation module. One option is that the Decision Making Module takes responsibility of providing a fully populated model with all initial values in a neutral data format, like the CMSD XML file. Another option is that the service generator provides a model that only includes references to data sources, from which the simulation module acquires the initialization data at the start of each run. The plugins for each simulation code reads the initialization data in a neutral format and produces input files that are suitable for the tool.

CMSD [11]is a standard model definition for production simulation models. It defines a data model both as a UML specification and an XML schema for exporting the model into an ordinary file. CMSD[12] can be used as a neutral storage for production simulation models that are independent of the simulation tools. It has been shown that discrete event simulation models for various tools can be created from a suitable CMSD model[13].

[11] SISO. 2010. “Standard for: Core Manufacturing Simulation Data – UML Model.” Core Manufacturing Simulation Data Product Development Group, Simulation Interoperability Standards Organization. Accessed May 14, 2012. http://www.sisostds.org/DigitalLibrary.aspx?Command=Core_Download&EntryId=31457

[12] Soeren Bergmann, Soeren Stelzer and Steffen Strassburger (2011). Initialization of Simulation Models Using CMSD. Proceedings of the 2011 Winter Simulation Conference. S. Jain, R.R. Creasey, J. Himmelspach, K.P. White, and M. Fu, eds.Available at http://www.informs-sim.org/wsc11papers/200.pdf

[13] Jonathan Fournier (2011). Model Building With Core Manufacturing Simulation Data Translators. Proceedings of the 2011 Winter Simulation Conference. S. Jain, R.R. Creasey, J. Himmelspach, K.P. White, and M. Fu, eds.. Available at http://www.informs-sim.org/wsc11papers/199.pdf

8.4.1 Inputs/Outputs

In order to identify the generic inputs/outputs of the SM, the BC-specific inputs/outputs have been analysed first, in section 8.4.1.1. Then, in section 8.4.1.2, the generic inputs/outputs have been abstracted.

8.4.1.1 Inputs/Outputs of the Simulation Module in the three Business Cases

8.4.1.1.1 Inputs/Outputs of the SM in BC1

Since BC1 will use optimization algorithms, the interaction with the SM will be limited by the future needs for using simulation services. During the specification phase, the inputs and outputs of the BC1 with the SM will be revisited.

8.4.1.1.2 Inputs/Outputs of the SM in BC2

In BC2, the SM will be used for computing KPI values and simulation results, such as:

- Predicted cable temperature
- Predicted cable load
- Cable Hotspot detection
- Grid load (several cable sections combined)

In order to compute these results, the SM will receive configurations of BC2-specific services as inputs.

8.4.1.1.3 Inputs/Outputs of the SM in BC3

In BC3, the SM will be used for computing KPI values and simulation results such as:

- Productivity KPIs: What production rate can be achieved for a design using a given set of processes and resources?
- Sustainability KPIs: What are the energy consumption, the emissions and the hazardous material waste resulting from the manufacturing for a design using a given set of processes and resources?

In order to compute these results, the SM will receive configuration of simulation model and related parameters such as

- List of processes,
- Number and type of resources (equipment, tooling, etc,) and related cycle times,
- Equipment/resource energy consumption data
- Factory calendar and work shifts
- Components, materials needed for manufacturing, hazardous material data
- Life Cycle Inventory (LCI) data, e.g. energy carriers and related CO2 emissions
- Production orders

8.4.1.2 Generic Inputs/Outputs of the Simulation Module

8.4.1.2.1 Simulation service interface

The Simulation module provides a service interface that can be used for executing individual simulations. One design option is to provide a separate service address for each different simulator that the module is capable of running.

Inputs:

- An identifier for the simulator that is to be executed.
- A reference to the model that is to be simulated. The model itself is situated in the knowledge repository.
- Values for any open parameters in the model.
- Possibly other simulator-specific settings.
- A reference to a place where the simulation results are to be stored in the knowledge repository.

Outputs:

- Status information and/or possible error messages for the simulation run.
- Simulation results

8.4.1.2.2 Simulator service discovery interface

The Simulation module provides a service interface that allows the other modules to discover the simulator codes that are available in the Simulation module.

Inputs:

- Possible search terms or other conditions

Outputs:

- List of matching simulator plugins (in XML)

8.4.1.2.3 Simulator metadata query interface

The Simulation module provides a service interface that allows the other modules to query for detailed metadata for a plugin.

Inputs:

- Identifier for a simulator plugin

Outputs:

- Metadata for the simulator plugin (in XML), including
    - Name of the simulator tool
    - A description of the types of models the simulator is capable of processing
    - A description of possible simulator specific options
    - A description of the format of the simulator output

9 Implementation framework and plan

9.1 Software Engineering Approach

The EPES Architecture is based on the service-oriented development approach that represent several different processes as services that are fully interoperable and allow further re-use for specific process reoccurrences. This infrastructural foundation offers overall conceptualization, modelling and software architecture paradigms to integrate and export business services over system boundaries. The granularity is basically incoherently connected in the architecture and formally based on the possible abstract interaction between services offered through a global location interface or by the discoverability of each service (sometimes encapsulated in specification files for e.g. Web-Services).

The implementation concept therefore complies with this basic SOA rule-set. All layers of the EPES concept must be seen as part of an overall service structure so that every component could interact with each other but is additionally focussed around specific transition services. Therefore, EPES separates itself into a three parted structure that complies in every layer to a service oriented approach.

As knowledge backbone operates an abstracted storage service layer that handles every access, in or out, to the database. Engineered on SOA principles it administers the EPES repositories, allows creation and altering based on object-models and offers connections to the data for the embedded EPES services. The service layer can now support the operation through the persistence layer whereas it assembles all other services under a Service Component Architecture (SCA) that delivers the possibility to orchestrate and compose them. By that the internal as well as the external, e.g. for business specific applications, components may utilize all offered services through an overall registry. By that, external applications get the opportunity to attend to the EPES logical structure with different compositions as well as the possibility to make use of any service on its own.

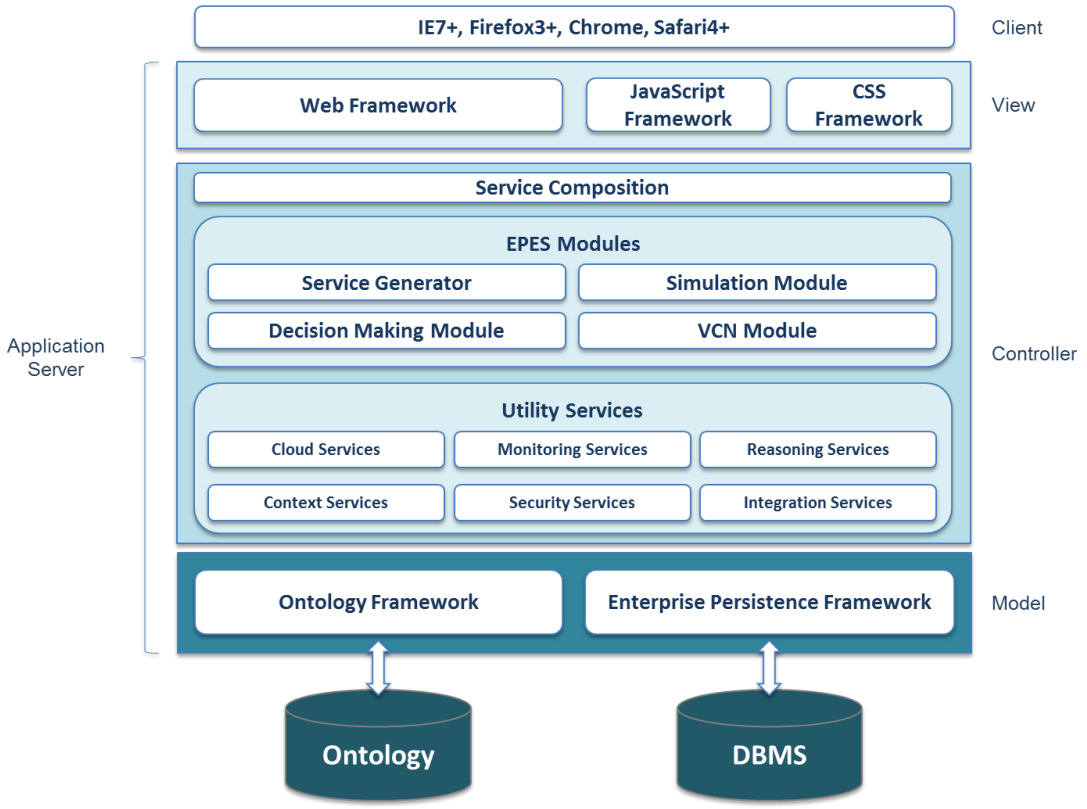

Figure 9-1: EPES Implementation Architecture

9.1.1 Service oriented engineering

In distributed software environment where different functionalities need to be congregated, the SOA approach offers an overall concept to base the engineering on standardized schemata for abstraction and communication between. By that, each service keeps its autonomy but concurrently becomes reusable and discoverable and therefore able to be composed in and by an overall SOA discovery service. This ability offers a EPES wide shared internal access to any registered specific service just as the external applications can dock onto the registry and call for explicitly configured orchestrations or single services. By that, a wide separation of every major aspect for specialized usage is accomplished that gives the possibility to interact between services.

9.1.1.1 Service component architecture

An all-encompassing solution that encapsulate all formulated interests based on the principles of SOA with a further technological approach and therefore better to conclude on internal and external software is the software component architecture.

The SCA delivers an easy way to orchestrate a separated service execution environment to specific needed process flows by recomposing the same services into different procedures that offer different internal or external overall services. These services are composed by the assembly part of the SCA and handle the entry points to the different services and composites, no matter if used inside or mapped to outside applications that need to dock onto the EPES system. The mapping and referencing of all components is contained by the SCA wires. Now implementations or business processes that conform to the components architecture can reuse the abilities provided by accessing the registry.

Devolved to the main EPES solution components, according service parts result in the execution environment that are channelled through the SCA concept. But, independent from the overall structure each component can consist of various services that are then orchestrated in parts or fully through the assembly.

9.1.1.2 Service Composition

The specification of the Open Service Oriented Architecture Group[14] specifies overall procedures how to implement and provide a SCA. The specification by default is not translated into usable components for software engineering but is constructed by various companies like BEA, IBM, Oracle or Apache. With the comprehensive support of many accredited developers throughout the industry the SCA achieved an overall acceptance as more technologically adaptable as the abstract SOA principles by themselves.

An as open-source published example reference implementation is the *Tuscany* framework from Apache[15]. In combination with a modular graphical representation of the composition structure through e.g. the *Eclipse STP/SCA* project, the framework offers two essential characteristics: The overall SCA and in addition the utilization of Service Data Objects (SDO), also pre-set by the OSOA. The framework therefore is no container that confines the services but delivers the structure for interoperating among them without explicit knowledge of interfaces but through discovery and composition of service needs.

So the pivotal point in the service architecture is the SCA implementing framework handling all internal as external requests for operations remitting and recomposing them to the right services that may themselves be conditioned to use additional work steps for the system. By that, the extension and modification of services or the workflow of these is separated into a modular concept

[14] http://www.osoa.org/
[15] http://tuscany.apache.org/

that allows quick and easy changes or additions. Furthermore through the standard compliance in the service architecture as well as the communication through possible SDOs an internal and external variation is made easy based on the specification.

9.1.1.3 Service Registry

To interoperate in a service execution environment, it is important to offer the ability to find other services. This is given by the stated SCA principle. It registers, references and composes services in the EPES system to ease up the modularity and alter/extension processes. The services are preconfigured handled in the architecture to work together in a flow as described throughout the concept. To apply the option of "docking" to external applications onto the EPES architecture, that gives a wider range for business specific application, it is necessary to offer external systems the ability to reach specific services.

The main approach to accomplish that is to keep every service accessible through Web-Services that should be attainable outside of the system (i.e. through the Internet). Each service can therefore define external interfaces based on one of the different Web-Service specifications like *Simple Object Access Protocol* (SOAP[16]), *Representational State Transfer* (REST) or *Remote Procedure Calls* (RPC) and publish them through common protocols (e.g. the *Web Service Description Language,* WSDL).

Through the use of standard components as the Java Enterprise environment and to predicate on the Service Component Architecture, the already available structures, orchestration and entry points allow, through simple expansion, the declaration of Web-Services by the *Java API for XML – Web-Services* (JAX-WS) with according reference frameworks like Apache CXF in correlation with Tuscany.

9.2 Implementation schedule

The preceding EPES concept describes the interaction of different services inside different environments, ranging from cable production up to maintenance of windmills. The overall aim is to allow industries to evaluate the performance of engineered products considering their whole lifecycle rather than only early stages such as design and manufacturing. The parts of simulating, making decisions and generating/configuring services all based on inputs from a VCN feature the main EPES process chain and will be developed by the RTD partners.

Based on the presented conceptual structure and architecture, the implementation will separate into two Milestones, delivering a simplified but methodologies complete early prototype, testable by the industrial partners. Further, the proof-of-concept full prototype will feature expanded application and business specific components, evolved throughout the early stages and tests. The context of the full prototype will extend the previous by tailored key parts of EPES to industrial and overall requirements, which are able to make use of the EPES solution.

The following sections describe and analyse the content of each prototype and the according implementation of the methodology. Whereas the early prototype conforms to the basic structure with simplified services and modules, assembling the foundation for further more specific extensions and evolved by testing services, the full prototype implements the full feature-set of the methodology. To distinguish between both, the implementation of the early prototype fully features the conceptualized methodology in a simplified manner. The full prototype will complete the methodology and expand the feature-set about further services, user compliant structures and additional implementations of services/modules.

[16] Nowadays only recalled as abbreviation SOAP to separate it from the SOA concept.

The prototypes are developed in a strongly end-user driven cycle of design, implementation and evaluation. The analysis and evaluation during the development phases as well as the active assessment inspection ensures the desired feature implementation and minimizes the project risks through permanent testing of all key elements.

In the next step, the RTD partners will specify the implemented content of the early and full prototype in work packages 300 and 400 (also including WP 500), according to the project's description of work. These are integrated into the overall prototype solution as a testable working result.

9.2.1 Early prototype content

The full prototypes in all work packages will be developed to cover all the functionalities that are defined in the Early Prototype Specification. In order to minimise risks, the project will apply early prototype approaches to enable early testing, gaining of experience and feedback from the users and early accumulation of the knowledge needed to build and test the applicability of new solutions in industry.

The early prototype of the components will be integrated and tested in the defined 3 BCs. The results of these tests will provide the first feedback from the ICT vendors and users. This information will be used for the development of the full prototype (FP) of components. The early prototype will also be used to examine organisational approach to collaborative design.

9.2.2 Full prototype content

The full prototypes in all work packages will be developed to cover all the functionalities that are defined in the Full Prototype Specification, taking into account the end users' feedback from the early prototype testing. Experience and feedback derived through user environment testing will be used to measure and compare the EPES solution with the list of expected benefits in order to detect issues to be improved in the full prototype. Therefore, existing services will be enhanced and refined to fully comply with the business cases requirements.

10 Conclusions

The main objective of the Task 130, *Concept Definition*, was to define the EPES concept for the methodology, architecture and software components. The results of this task are presented in detail in this Deliverable D100.3.

By identifying the gap between the existing state-of-the-art (surveyed in Deliverable D100.1 *State-of-the-Art Analysis*) and the requirements identified in Deliverable D100.2 *Requirements Analysis*, a conceptual view of the EPES solution's components has been developed. The main objective of deriving the features and functionalities of the Virtual Collaborative Network, Service Generator, Simulation and Decision Making modules, as well as the required service infrastructure for the EPES solution, has been achieved.

In the following Table **10-1**, the concepts of the main components and expected results are briefly described.

Table 10-1: Overview of the main components and expected results

EPES Solution Component/Result	Description
Virtual Collaborative Network	The Virtual Collaborative Network provides end-users (non-experts) with a main point of access to the whole EPES solution, enabling the distribution of the users into groups and the capturing of their knowledge for later reuse. The main group is the *Business Community*, which involves and contains all the actors within the Virtual Factory domain, such as users, suppliers, clients, governmental agencies and RTD partners. Based on the identification of business optimization opportunities or necessities, more specialized *Collaborative Spaces* can be created. A *Collaborative Space* represents a specific goal-oriented group aimed at, e.g. the collaborative identification of sustainability constraints and environmental objectives that would shape the organizations' business processes towards sustainability.
Service Generator	The Service Generator enables the user to perform the following: • Adjusting the configuration of deployed services to reflect changing conditions • Deploying the updated configurations • Storing/retrieving configurations • Detecting user context in order to refine the parameters of configurations.

EPES Solution Component/Result	Description
Simulation Module	The Simulation Module enables the user to run numerical analyses of production systems, maintenance operations, or other aspects of the life-cycle assessment process. It is responsible for the integration of several simulation tools and data sources into a single service that the other modules can utilize in order to perform simulations or other complex analysis tasks. In order to perform a requested analysis, the SM acquires a description of the model, and a set of parameter values. The model description is provided by the Service Generation module, while the parameter values are provided by the Decision-Making module. The Simulation module combines these into a "black box" model that can be executed as a part of a simulation process that is specified by the user.
Decision Making Module	The Decision Making Module supports and guides the user in the decision making process. It enables the analysis and processing of information gathered from the life cycle of the product, also allowing its comparison with the "should be" performance metrics which are the output of the simulation system. The typical usage is to provide the DMM with business process definitions, structured data, raw data sets, KPI; then, with the support of a human expert, it produces transformed data, graphic analyses, updated business processes, and the "best" solution from among a set of reasonable alternatives.
EPES Methodology	The EPES Methodology will contain guidelines on the usage of: • Methods for setting up and maintaining collaborative working environments (Business Communities) • Methods for modelling business processes and implementing process execution models • Methods for monitoring and optimising business processes in the Business Community • Methods for setting up the EPES Business Community repository, allowing the storage of different data • Methods for assessing environmental impacts and resource requirements throughout the most important phases of life cycle of products and processes

EPES Solution Component/Result	Description
EPES Architecture	Based on the service-oriented development approach that represents several different processes as services that are fully interoperable and allow further re-use for specific process reoccurrences. All layers of the EPES concept are seen as a part of an overall service structure so that every component could interact with each other, but is additionally focused around specific transition services. EPES separates itself into a three parted structure that complies in every layer to a service oriented approach.

Furthermore, the implementation plan, defining the scope of the Early and Full prototype, has been defined. In the next phase of the project, the components, underlying infrastructure and methodology will be specified in detail. This specification will serve as the basis for the Early Prototype first and then for the Full Prototype. The prototypes will be tested within the application scenarios defined for the industrial partners in the consortium in Deliverable D100.2.