Grant Agreement N° 215483

Title: *Final version of the methodology for describing pilot cases.*

Authors: *POLIMI, FBK, Tilburg*

Editor: *Pierluigi Plebani (POLIMI)*

Reviewers: *Patricia Lago (VUA)*

George Ploumakis (UoC)

Identifier: *CD-IA-2.2.5*

Type: *Deliverable*

Version: *1.0*

Date: *30 November 2010*

Status: *Final*

Class: *External*

Management Summary

This deliverable introduces the final version of the methodology for describing pilot cases starting from the first version described in the Deliverable CD-IA-2.2.2. This new version has strict relationships with other S-Cube Integrated Activities: IA-1.1 for the Knowledge Model and IA-3.1 for the Integrated Research Framework. The methodology is also available as a Wiki available at http://scube-casestudies.ws.dei.polimi.it that forms integral and substantial part of this document.

The research leading to these results has received funding from the European Community's Seventh Framework Programme FP7/2007-2013 under grant agreement n° 215483 (S-Cube).

Members of the S-Cube consortium:

University of Duisburg-Essen (Coordinator)	Germany
Tilburg University	Netherlands
City University London	U.K.
Consiglio Nazionale delle Ricerche	Italy
Center for Scientific and Technological Research	Italy
The French National Institute for Research in Computer Science and Control	France
Lero - The Irish Software Engineering Research Centre	Ireland
Politecnico di Milano	Italy
MTA SZTAKI – Computer and Automation Research Institute	Hungary
Vienna University of Technology	Austria
Université Claude Bernard Lyon	France
University of Crete	Greece
Universidad Politécnica de Madrid	Spain
University of Stuttgart	Germany
University of Hamburg	Germany
VU University Amsterdam	Netherlands

Published S-Cube documents

All public S-Cube deliverables are available from the S-Cube Web Portal at the following URL:

http://www.s-cube-network.eu/results/deliverables/

The S-Cube Deliverable Series

Vision and Objectives of S-Cube

The Software Services and Systems Network (S-Cube) will establish a unified, multidisciplinary, vibrant research community which will enable Europe to lead the software-services revolution, helping shape the software-service based Internet which is the backbone of our future interactive society.

By integrating diverse research communities, S-Cube intends to achieve world-wide scientific excellence in a field that is critical for European competitiveness. S-Cube will accomplish its aims by meeting the following objectives:

- Re-aligning, re-shaping and integrating research agendas of key European players from diverse research areas and by synthesizing and integrating diversified knowledge, thereby establishing a long-lasting foundation for steering research and for achieving innovation at the highest level.
- Inaugurating a Europe-wide common program of education and training for researchers and industry thereby creating a common culture that will have a profound impact on the future of the field.
- Establishing a pro-active mobility plan to enable cross-fertilisation and thereby fostering the integration of research communities and the establishment of a common software services research culture.
- Establishing trust relationships with industry via European Technology Platforms (specifically NESSI) to achieve a catalytic effect in shaping European research, strengthening industrial competitiveness and addressing main societal challenges.
- Defining a broader research vision and perspective that will shape the software-service based Internet of the future and will accelerate economic growth and improve the living conditions of European citizens.

S-Cube will produce an integrated research community of international reputation and acclaim that will help define the future shape of the field of software services which is of critical for European competitiveness. S-Cube will provide service engineering methodologies which facilitate the development, deployment and adjustment of sophisticated hybrid service-based systems that cannot be addressed with today's limited software engineering approaches. S-Cube will further introduce an advanced training program for researchers and practitioners. Finally, S-Cube intends to bring strategic added value to European industry by using industry best-practice models and by implementing research results into pilot business cases and prototype systems.

S-Cube materials are available from URL: http://www.s-cube-network.eu/

Contents

1 Introduction

The goal of Work package IA-2.2 is to create links with industry, to present and discuss results from the research to relevant industrial players, and to benefit from industrial experience. More specifically, in this work package, one of the tasks is about the description of pilot case studies that come from industries. As a first result of this task, in the CD-IA-2.2.2 industrial case studies have been collected and analyzed and a first version of a methodology for describing the case studies has been proposed.

Goal of this deliverable is to present a refined version of this methodology that also includes results from other Integrated Activities in S-Cube. As shown in Figure 1, in this new version the description of the case studies are linked to both the Knowledge Model (KM)[1] defined in the IA-1.1 and to the Integrated Research Framework (IRF)[2] developed in the IA-3.1. The links to the KM and the IRF offer two ways to classify the case studies. On the one hand, a case study can be linked to specific research challenges or questions as defined in the IRF; on the other hand, some of the terms included in the KM can be used to describe a case study. As a result, the case studies are directly associated with terms and research challenges that are relevant for the Service Engineering community.

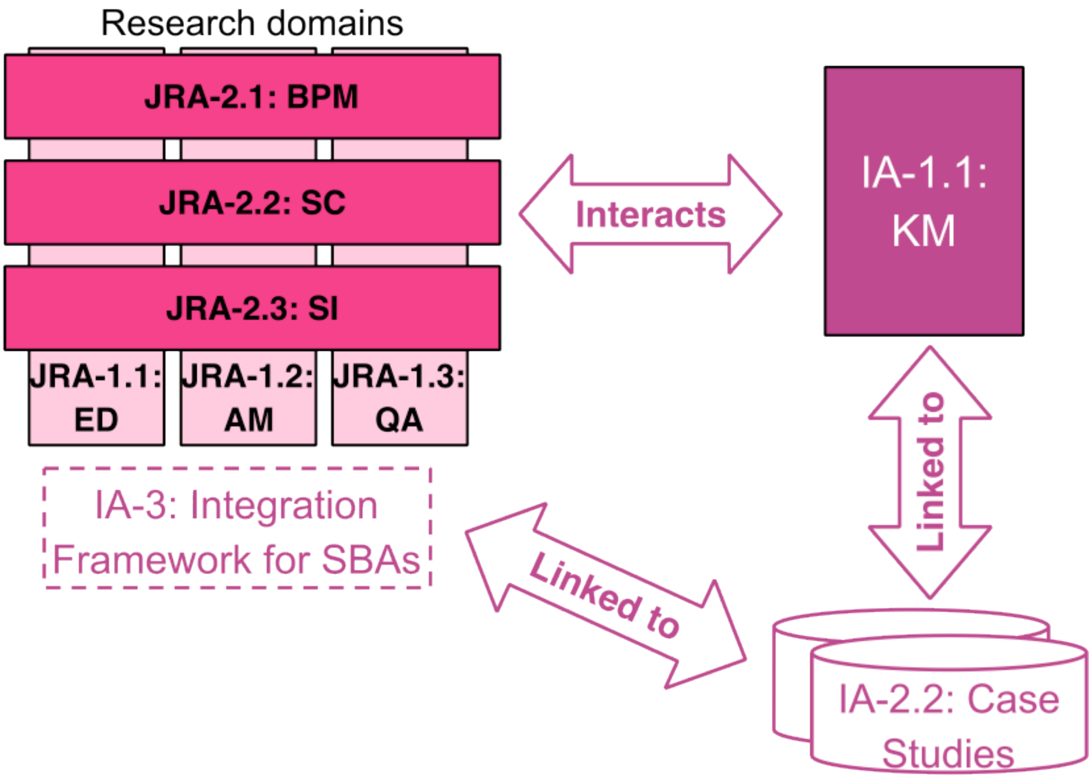

Figure 1 - Interaction among S-Cube Integrated Activities

This new version of the methodology, discussed in Section 2, has also been used and promoted in the EC event on Internet of Service 2010 [3]. The Collaboration Work Group "Collecting Use Cases" will be responsible for driving other EU Projects to describe their case studies, where in case of Integrated Projects (IP), the case studies are originated by industrial partners. For this reason, our methodology has been designed to cover a broad spectrum of researches, with the idea of specifically facilitating the description of industrial cases, so that they could be reused across areas and in an inter-disciplinary.

1 http://www.s-cube-network.eu/km

2 http://grecale.isti.cnr.it/IRF/Login.php

3 http://ec.europa.eu/information_society/events/ssai/ios/index_en.htm

For the refinement of the methodology and in alignment with the mission of S-Cube, to establish a research community on Software Services and Systems, a Wiki [4] was created that includes all case studies introduced in CD-IA-2.2.2 . This Wiki is publicly available to external actors for publishing their own case studies. In this way, the benefit for S-Cube is twofold. On the one hand, as the number of the published case studies increases, more feedback, about the methodology, become available. On the other hand, that is a mean for creating links between industry and research community. Section 3 introduces the Wiki in detail.

Finally, as a result of this effort, the methodology for describing the case studies had been published in a book that collects a set of research results on Service Engineering. In particular, the methodology is used in the first chapter of the book to describe a new case study on a telecommunication company [1].This case study has been referenced as a common case study in the rest of the book. A copy of this book chapter is attached to this deliverable and the description of this telecommunication case study is included in the Wiki.

[4] http://scube-casestudies.ws.dei.polimi.it

2 Methodology

Case study descriptions can include different material, depending on their purposes. For instance, they can include a specific software solution or proof of concept, or they can simply describe an application case without offering a specific implementation solution. Of course, while in the first case the case study description contains also design, implementation, and even deployment and operation details, in the second case it should be implementation- and technology agnostic.
In this section we refer to case study descriptions of the second kind as they can be reusable assets that could be exploited as reference cases in the context of various projects. As such, the description should be focusing on eliciting those goals and assumptions that the software should address [5]. Details about how to describe a case study are given in Section 2.1.
In addition, a classification is introduced to allow us to cluster case studies according to their application domain and the research topics that are relevant for them. For this reason, we introduce two levels of classification (see Section 2.2):

- Domain-oriented classification: it defines a high level classification that makes possible to identify in which domain a case study applies. This classification proposes a list of macro-areas inspired by the EC Workprogramme. Each case study is associated to one of these areas.
- Research-oriented classification: it defines a classification of research challenges that are interesting in a given research area. Using this classification, a case study can be associated to research challenges that emerge from the case study or that needs to be addressed in order to develop a solution for the case study. The IRF developed in the IA-3.1 can be used for this purpose [6] if the case study can be adopted in Service Engineering.

Finally a list of rules to be followed when describing a use case is provided in Section 2.3.

2.1 Case study description

In order to make case studies comparable and easy to understand, S-Cube has defined a case study description approach based on the Requirements Engineering literature [3] and the results achieved by NEXOF-RA [4]. S-Cube has also experimented the approach by revisiting a number of cases described in the NEXOF-RA deliverables. This has allowed us to highlight inconsistencies and incompleteness in the previous descriptions. According to the methodology initially introduced in the Deliverable CD-IA-2.2.2 and now refined, a case study is described by:

- A list of *Business Goals* and *Domain Assumptions*.
- A Domain description.
- A list of Scenario descriptions.

2.1.1 Business Goals and Domain Assumptions

Business Goals define objectives to be pursued and functionalities to be offered. Domain Assumptions describe properties that are assumed to hold for a certain domain. Table 1 defines a template for describing Domain Assumptions and Business Goals. The description includes the involved stakeholders, the rationale, the priority and the material supporting the description, if any.

Table 1 – Business Goal and Domain Assumption description template

Field	**Description**
Unique ID	Give a unique ID for this goal/assumption
Short name	Give a short name for this goal/assumption

Type	One of the following: ▪ Business goal ▪ Domain Assumption
Description	Specify the intention of the goal/assumption
Rationale	Give a justification of the goal/assumption
Involved Stakeholder	Stakeholders involved in the business goal/assumption
Supporting materials	Give a pointer to documents that illustrate and explain this goal/assumption (in particular those of domain analysis)
Priority of accomplishment	One of the following: ▪ Must have: The system must implement this goal/assumption to be accepted. ▪ Should have: The system should implement this goal/assumption: some deviation from the goal/assumption as stated may be acceptable. ▪ Could have: The system should implement this goal/assumption, but may be accepted without it.
Tentative scheduling	Tentative scheduling of accomplishment. To be used only if the case study has to be implemented.

2.1.1 Case study Domain description

The application domain is defined by the set of phenomena occurring in the world together with the laws that regulate such world (e.g., physical laws, social rules, conventions that need to be respected). In the case a software system (a machine) is needed in order to fulfill certain goals, such machine needs to have an impact on the world. Thus, the two corresponding domains – world and machine - have to partially overlap. The phenomena that are at the intersection between the world and the machine are called shared phenomena. These can be either controlled by the world and observed by the machine, or, conversely, controlled by the machine and observed by the world. The study of such phenomena is particularly important since they define the interface between the machine and the world. Of course, shared phenomena (and therefore scenarios) can be understood in the context of the world in which the machine will work and of the laws governing the world. Also, the boundaries between the world and the machine have to be clearly identified. In order to address these aspects we suggest to include in the case study domain description the following items:

- A glossary that defines the main terms of the world. For this aspect, the Knowledge Model defined in the IA-1.1 can be useful as a collection of the relevant terms for the Service and Systems Engineering.

- A description of the relationships between the terms of the glossary. The glossary alone does not highlight the relationships between the various terms nor their relative importance. Thus we need to build a model that highlights these aspects. Class diagrams are usually a good tool for this purpose since they allow the engineer to identify main entities as classes and to express several kinds of relationships between them. Entity-relationship diagrams as well as semantic networks for our purposes have an expressive power that is similar to class diagrams and therefore can be used as well.
- A description of any law that is relevant in the world. Such laws can be expressed in any form that is typical of the application domain that we are considering: mathematics, logics, natural language, and so on.
- Strategic Dependency Diagrams (SDDs) [2]. These are used to model the dependencies between the different actors in the organisational context. They especially help to model user (roles) together with their relations. Dependency edges in the diagram link the actors with needs (dependers) to actors with the capability of meeting those needs (dependees). The needs are expressed in terms of goals (positioned on the edges).
- Context Diagrams (CDs) (Jackson, 1995). These identify the agents that operate in a certain context as well as the machine that needs to be developed. Moreover, they highlight the phenomena shared between agents and machine. Figure 2 shows the notation of context diagrams. In a context diagram, any active entity on the case study to be modeled is represented as a box, while a directed arrow describes phenomena between agents. Both the SDDs and the CDs represent the agents/actors involved in the domain, but while the SDDs show the dependencies among them, CDs highlight also other kinds of relationships among them. Moreover, they clearly identify the boundaries between the machine and the world.

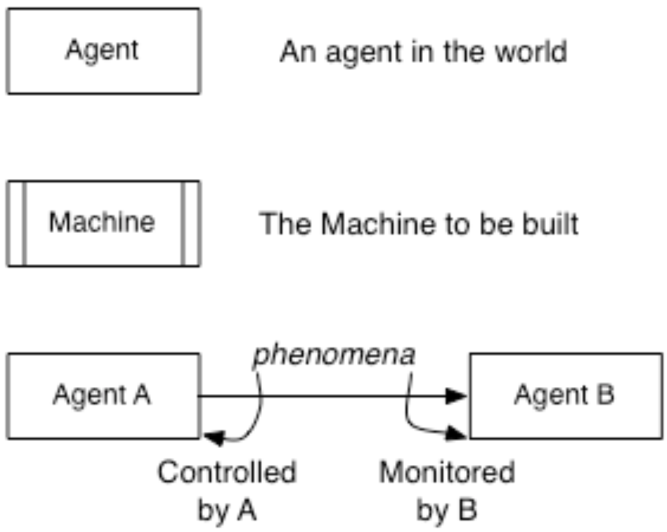

Figure 2 - Context Diagram Notation

2.1.2 Scenario description

The phenomena shared between the world and the machine can be detailed through scenario descriptions. Scenarios have an operational flavour in the sense that they describe the steps that need to be executed by the machine and the world entities in order to accomplish a certain task.

Table 2 describes how scenarios should be detailed and described, and it should be used as a template for any single scenario description. Here, a scenario is described using information about the business goals or the domain assumptions they refer to, the operational description of the scenario, the possible problems involved and the supporting material.

Table 2 - Scenario description template

Field	**Description**
Unique ID	Give a unique ID for this scenario
Short name	Give a short name for this scenario
Related to	Specify the goal/assumption IDs to which the scenario is related
Involved actors	Specify the actors involved in the current scenario
Detailed operational description	Give a textual description of the scenario
Problems and challenges	Describe the specific problems that each scenario addresses or that consumers and providers face
Additional material	e.g. UML diagrams supporting the understanding of the scenario

2.2 Classification

2.2.1 Domain-oriented classification

As a first level of classification, each case study needs to be associated to an application domain. A list of possible application domain includes, and it is not limited to, the following topics:

- Business Application
  - Small Medium Enterprise
  - Large Enterprise
- E-Government
  - Government to Government
  - Government to Citizen
- Utilities
  - Transportation
  - Power supply

2.2.2 Research-oriented classification

An orthogonal classification considers the research topics that could have potential impact on a case study. In more detail, we adopt the S-Cube IRF as a way for organizing the research challenges and results about Service-based Applications (SBAs).

The IRF for SBAs of S-Cube, aims to integrate, align, and coordinate the joint research activities undertaken within the project [6]. To achieve this, IRF provides a coherent and holistic view on the principles and mechanisms for SBAs. In particular, the IRF should encompass those aspects of the research that are cross-cutting and defines proper interfaces among the research components of the overall conceptual network architecture. To continuously coordinate and align the research roadmap of the project as a whole, the framework is continuously validated with the help of the industrial case studies.

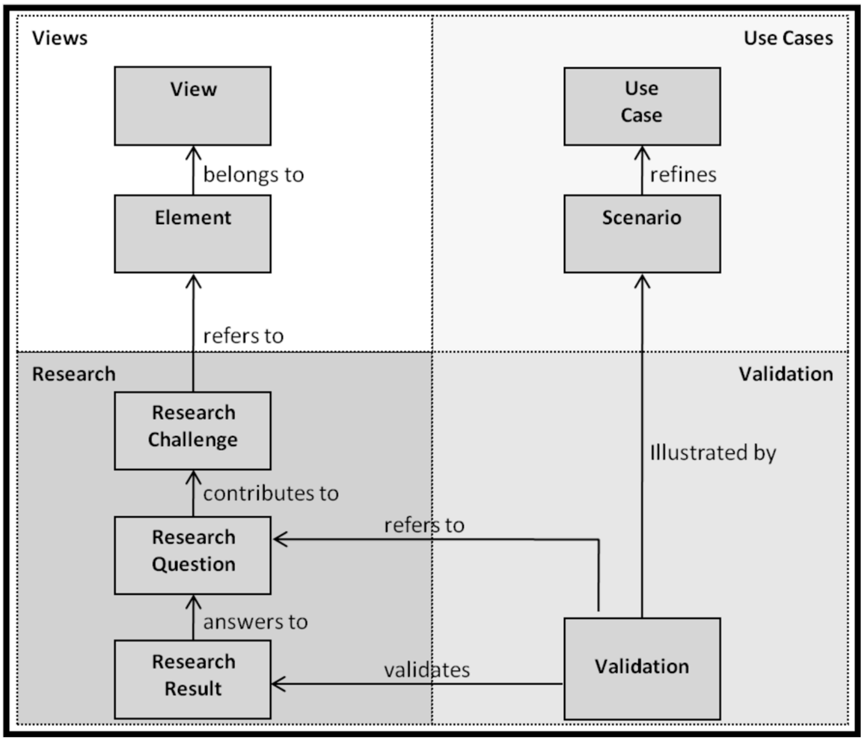

Figure 3 - Structure of the IRF

More specifically, the IRF may be represented with the eight macro elements that are clustered into four conceptual blocks as represented in Figure 3. Focusing on the "Research" block, it defines the objectives and results of the S-Cube research activities layered in:

- Research challenges. They define long-term research goals w.r.t. to the goals of research project in which the case study had been defined. For instance, in S-Cube, several research challenges around the life cycle of Service Based Applications are defined. As an example we have:
  - QoS Aware Adaptation of Service Compositions
  - Proactive Adaptation and Predictive Monitoring
- Research questions. Associated to a research challenge, a set of research questions may exist identifying specific short-terms research objectives. For instance, referring to the "QoS Aware adaptation research" challenge listed before, the following research questions can be identified:
  - Adaptation of QoS-aware Service Compositions based on Influential Factor Analysis and Prediction.
  - How can cost-based derivation of data-aware QoS for a service composition be used to drive adaptation?
  - Linkage between Business Transactions and Service Compositions.
- Research Results. The outcomes and achievements of the research efforts that aim to answer those questions:
  - End-to-End Quality of Service Model

Relationships between a case study and one or more research questions should be defined and the rationale behind each of these associations needs to be properly described. In particular, a relationship exists if the case study will be used to give a valid answer to a research question, or a research question is highlighted by a typical situation as described in the case study. In particular the link between the results and the case study is mapped by the validation that captures the specific goal of the validation activity, the scenario used in that activity, the set up, the validated result, and the outcome of the validation process.

2.3 Applying the methodology

The description of Business Goals, Domain Assumptions, Domains, and Scenarios are not necessarily obtained through a sequential process that starts from the identification of the goals then moves to the analysis of domain, and, finally, to the description of scenarios. Instead, as in many other highly intellectual processes, it is more likely to proceed iteratively, starting from any of the three points and compiling them more or less in parallel. What we can do is to provide a list of simple rules that allow us to understand when we can decide that our case study description has reached a reasonably good form:

- A glossary of relevant terms for the application domain has to be defined.
- The terms used in the scenarios and in the identification of the business goals and of the assumptions are properly described in the glossary and they are related to the other terms in the domain model.
- The entities identified in the domain model are used in some scenario or in some business goals and domain assumptions description.
- All actors that have been identified in the scenarios appear also in the context diagram (and/or in the Strategic Dependency diagram) and vice versa.
- From each scenario there exist at least one related business goal and vice versa.
- Scenarios are not overlapping. Relationships are possible but they should be explicitly identified.
- Goals are not overlapping. Relationships are possible but they should be explicitly identified.
- Research challenges and questions needs to be defined to drive the selection of the case study for validating the research results.

3 Case studies Wiki

The methodology introduced in the previous section and the description of the case studies defined in S-Cube, had been made available on a Wiki. In this way, S-Cube along with the Internet of Services-Collaboration Work Group "Collecting Use Cases" want to promote the approach and will host the description of case studies that other projects and industries provide. We decided to rely on a wiki for this purpose due to the nature of the case studies descriptions: semi-structured documents hardly fit in a relational database. In addition, a wiki make possible an easy way for external contribution to add new case studies. The drawback is the browsing functionalities of the case studies. At this stage, the wiki allows only keyword based search. Next releases will include facilities to navigate the case studies also exploiting the domain and research-oriented classification.

Figure 4, Figure 5, and Figure 6 give an overview of the most relevant pages in the wiki. In particular, Figure 4 shows the main page where the objective of the wiki and the list of already published case studies are provided. Figure 5 shows how a new case study can be defined by using the wiki editor. Finally, Figure 6 shows part of the template made available to drive the contributors during the description of the case study.

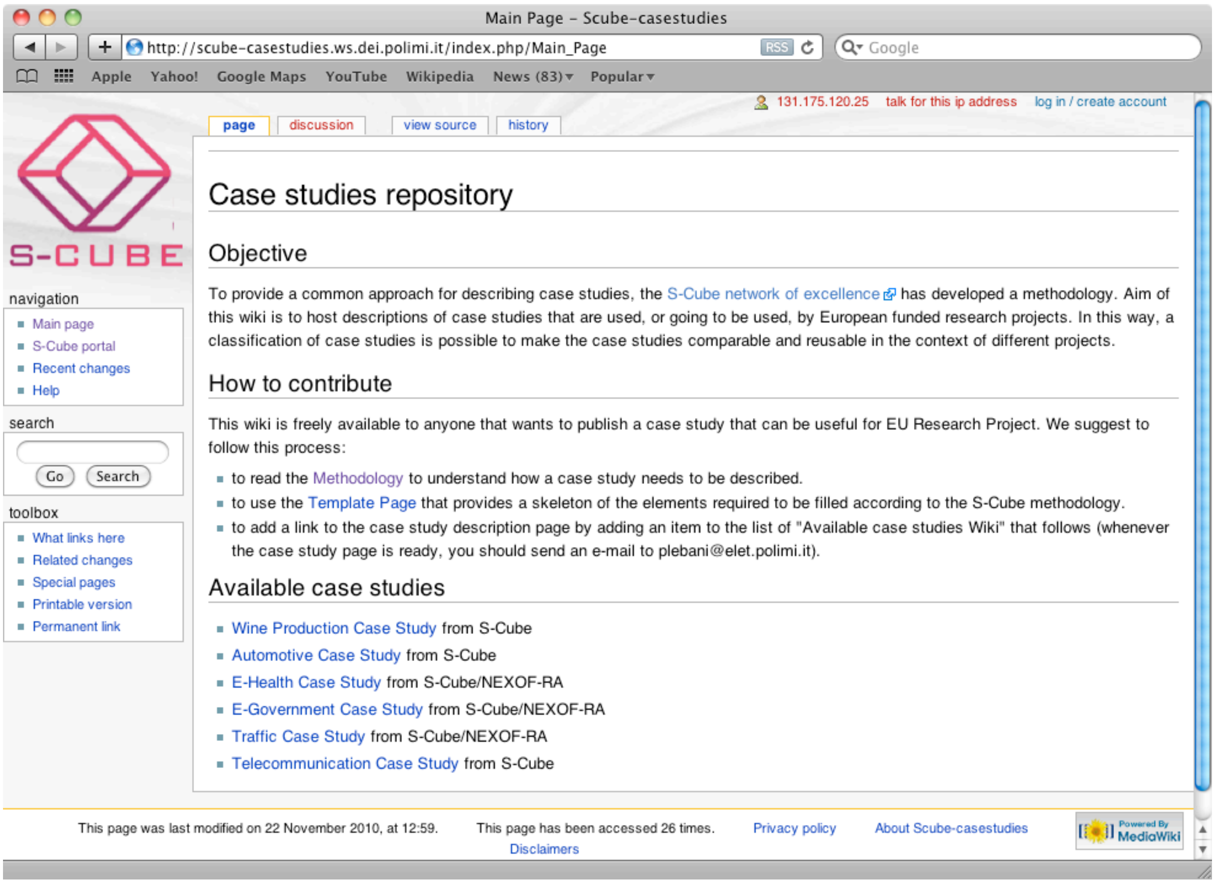

Figure 4 - The main page with the list of the case studies

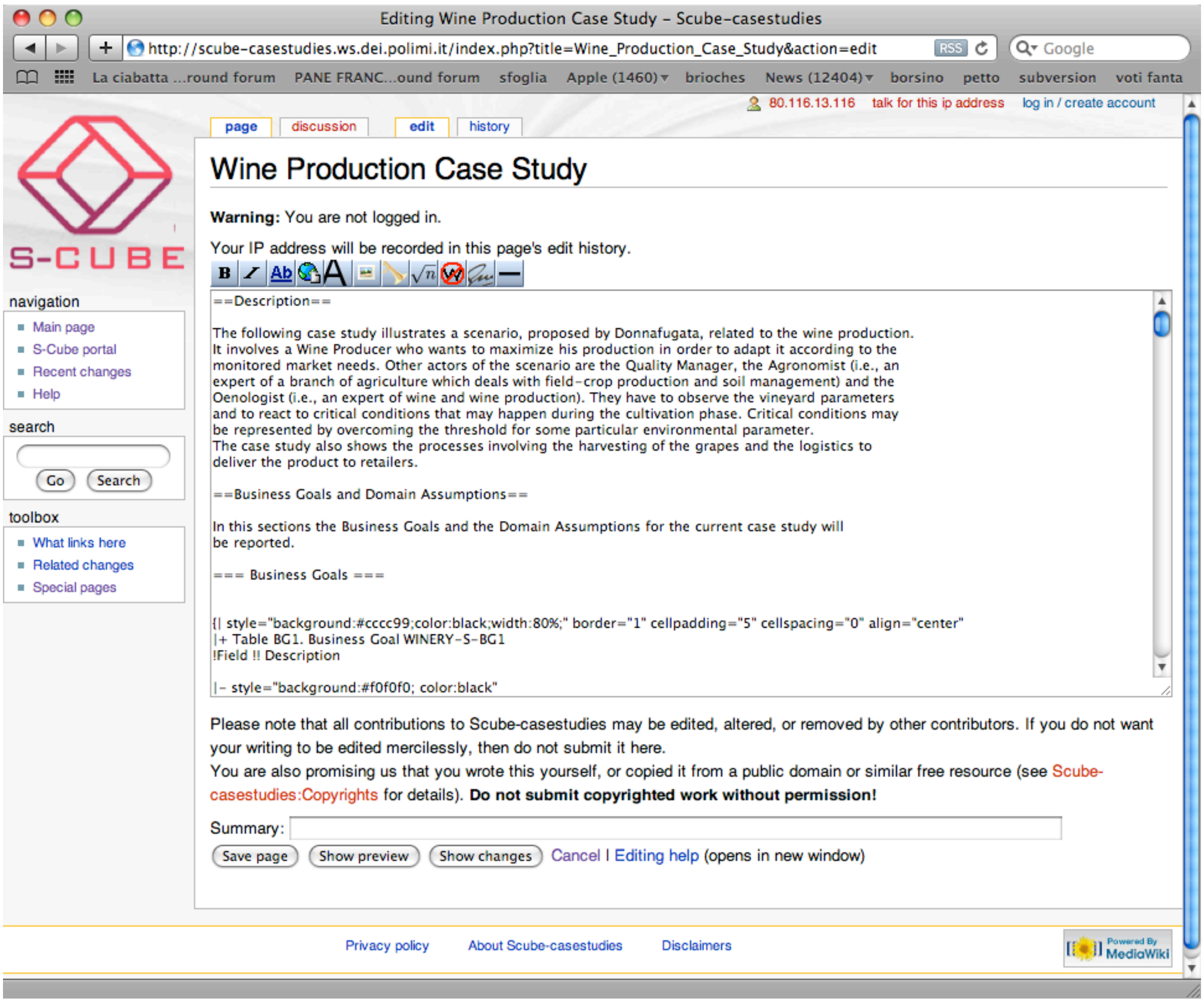

Figure 5 - Definition of a case study

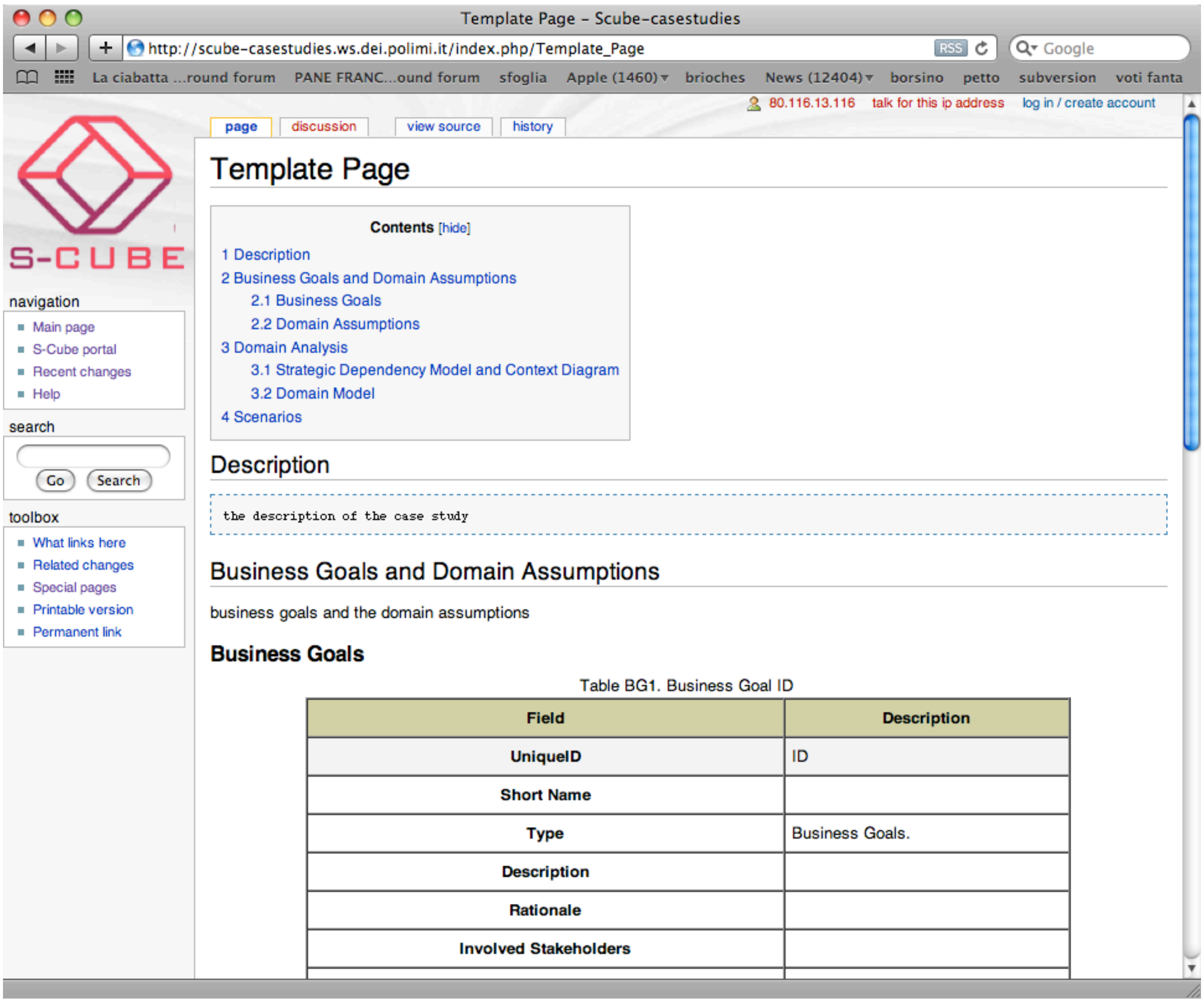

Figure 6 - Template made available to help the case study definition

4 References

[1] A. Bucchiarone, R. Kazhamiakin, V. Mazza, P. Plebani, Describing Case Studies and Classifying Research Approaches, in S. Dustdar, F. Li, “Service Engineering: European Research Results”, October, 2010, Springer WienNewYork.

[2] i-star software formalism, http://www.cs.toronto.edu/km/istar/.

[3] M.Jackson, Software requirements & specifications: a lexicon of practice, principles and prejudices. ACM Press/Addison-Wesley Publishing Co., New York, NY, USA, 1995.

[4] NESSI Open Framework - Reference Architecture (NEXOF-RA): Scenarios and Requirements for Open Framework Construction.

[5] S-Cube Deliverable CD-IA-2.2.2 - Collection of Industrial Best Practices, Scenario and Business Cases.

[6] S-Cube Deliverable CD-IA-3.1.3- First Version of Integration Framework.

Appendix A List of attachments

A. Bucchiarone, R. Kazhamiakin, V. Mazza, P. Plebani, Describing Case Studies and Classifying Research Approaches, in S. Dustdar, F. Li, "Service Engineering: European Research Results", October, 2010, Springer WienNewYork.

Abstract. This initial chapter aims at providing a useful introduction and reference point to the research described in the following chapters. First of all, to provide a homogenous description of the existing approaches on service engineering, we introduce a common case study referring to the telecommunication domain. This case study will be used in the other chapters of this book to motivate and describe the presented approaches. The way in which the case study is described follows the approach developed in the S-Cube Network of Excellence. In addition, we also provide a classification of the research results proposed in the rest of the book, by relying on the S-Cube Integrated Research Framework. Such a classification allows the reader to have an idea about how the contributions deal with research in the service engineering field.