FI.ICT-2011-285135 FINSENY

D8.1 v.1.1 Final

FINSENY Experimentation Lab Set-up

Contractual Date of Delivery to the CEC:	*31.03.2012 (Month12)*
Actual Date of Delivery to the CEC:	*v1.0 Final 10.04.2012*
	v1.1 Final 15.10.2012 (Revision according to Review Recommendation)
Author(s):	*ENG*
Participant(s):	*ALUD, ATOS, EDD, FIR, Grenoble INP, Intune, NSNF TI, TID, TSSG, OLF.*
Workpackage:	*WP8 – Domain Specific Enablers and Experimentation*
Estimated person months:	*25*
Security:	**PU** = Public
Nature:	**R** = Report
Version:	1.1 Final
Total number of pages:	71

Abstract: This deliverable describes the set-up of the common framework (in terms of methodology and capabilities) of the FINSENY Experimentation Lab, which will support the activities for identifying, defining, evaluating, testing and validating the feasibility of scenarios and their specific enablers. The document provides guidelines for describing how to apply the methodology and the different combinations and configurations of capabilities of the Experimentation Lab for validating the scenarios and their domain specific enablers. Finally, the document provides a list of intended experiments, adopted experimentation lab capabilities and expected outcomes.

Keyword list: FI-PPP, FINSENY, Experimentation Lab, Common Framework, Evaluation Methodology, Feasibility Analysis, Relevance, Practicability, Domain Specific Enablers, Experimentation, Real-time Simulation, Traceability, CMMI, Best Practices, Software Engineering.

Disclaimer: N/A

Executive Summary

One objective of Work package 8 is the demonstration and experimentation of domain specific enablers. Those enablers are derived by the requirements and it is important to ensure they are suitable solutions for supporting and enacting all the investigated scenarios. In this perspective, it is important to analyse and validate the feasibility of the enablers for implementing the scenarios, and that feasibility has to be carried out on the characteristics of the enablers and based on common acceptance criteria (such as relevance and practicability).

For aiming this objective, Work package 8 is setting up an experimentation lab that provides a common framework (in terms of tools and methodology) for supporting the feasibility analysis during the evaluation and validation phase.

The activities of Task 8.1 focus on identifying and describing the Experimentation Lab in terms of: i) methodology and tools, ii) requirements, iii) capabilities, iv) architecture, v) set-up configurations and vi) guidelines for the evaluation and validation phase. This document exactly reflects those activities in its own structure.

Suggested methodology and tools for evaluating the relevance of enablers is the Traceability, which is a best practice on Software Engineering coming from Capability Maturity Model Integration – CMMI. The intent of this specific practice is to maintain the bidirectional traceability of requirements for each level of product decomposition. When the requirements are managed well, traceability can be established from the source requirement to its lower level requirements and from the lower level requirements back to their source. Such bidirectional traceability helps determine that all source requirements have been completely addressed and that all lower level requirements can be traced to a valid source. Thereby it will be ensured that the domain specific enablers cover the requirements of all investigated scenarios.

Suggested methodology and tools for evaluating the practicability is the Simulation and the Experimentation with real capabilities, which are usual approaches for evaluating behaviour and performance of algorithms and modelled systems. Because the simulator functions in real time, the power system algorithms are calculated quickly enough to continuously produce output conditions that realistically represent conditions in a real network. Real-time simulation has many significant advantages: i) allows fast, reliable, accurate and cost effective analysis, test and validation of power grid and its components; ii) system-failure scenarios and/or physical damage event can be analysed and detected without affecting the operation and security of the real power grid environment.

The methodology allows identifying requirements that have to be satisfied by the Experimentation Lab in order to properly perform the evaluation and validation. Based on those requirements, it is carried out an in an investigation and selection of existing capabilities in current EU research project trials, for supporting the real-time simulation and experimentation. So, the Experimentation Lab is set-up by composing and properly configuring those capabilities. Indeed, due to the various and different scenarios and their specific needs, it is necessary to configure the capabilities of the Experimentation Lab.

The tools and methodology described in this deliverable represent key elements of the FINSENY Experimentation Lab and those will be adopted along the whole lifecycle of the WP8, and in particular: i) Traceability for identifying and selecting suitable enablers, and provide a ranked list of suitable enablers based on relevance criteria (mainly the activity carried out in Task 8.2); ii) Real-time Simulation and Experimentation for evaluating the complexity of the enablers to be implemented in Phase 2, and provide a ranked list of suitable enablers based on practicability criteria (mainly the activity carried out in Task 8.3); iii) Finally, Traceability and Real-time Simulation and Experimentation results from Task8.2 and Task 8.3 will be inputs and supports the selection of broadly accepted and widely applicable enablers that are specific to the energy domain (mainly the activity carried out in Task 8.4).

For the above reasons, this deliverable describes the FINSENY Experimentation Lab which provides a common framework in terms of methodology & tools for evaluating the feasibility for implementing the enablers (based on an industry best practice), provides a template for gathering description of existing capabilities (based on experiences from the contributing partners) and reports the most prominent ones for the FINSENY Use Case Scenarios, and finally demonstrates how to apply the methodology & tools of the FINSENY Experimentation Lab to a set of sample scenarios. The latter aspect allows describing the architecture of the Experimentation Lab and its set-up configurations.

In order to address the recommendations of the review panel ("Recommendations concerning the period under review" in the technical review report referenced **Ares(2012)1023647 - 03/09/2012)**, a new release of this deliverable (v1.1 Final 15.10.2012) has been issued with the following updates and addenda:

- A re-structure and update of paragraph 2 “Introduction”, which provides “[..] *the scope of the Experimentation Lab set up”*;
- The introduction of paragraph 3.1 “Value added to the FINSENY project by the Experimentation Lab” which address the request from the reviewers to point out how “[..] *the Experimentation Lab provides clear value added for the project itself and a possible phase 2”*;
- The introduction of paragraph 6.4 “Intended experiments and expected results” which address the request from the reviewers to clearly “[..] *specify the scope and requirements for the Experimentation Lab based on a clarification of the intended experiments and expected results*”

Authors

Partner	Name	Phone / Fax / e-mail
Alcatel-Lucent Deutschland AG		**Bashir Ahmadi** Phone: +49 30 7002 17482 Fax: +49 30 7002 17481 e-mail: Bashir.Ahmadi@alcatel-lucent.com
Alcatel-Lucent Deutschland AG		**Thomas Loewel** Phone: +49 30 7002 17572 e-mail: Thomas.Loewel@alcatel-lucent.com
ATOS Spain SA		**Martin Nicolai Wagner** Phone: +34 91 214 8227 e-mail: martin.2.wagner@atosresearch.eu
Engineering Ingegneria Informatica spa		**Luigi Briguglio** Phone: +39 06 8307 4074 e-mail: Luigi.Briguglio@eng.it
Engineering Ingegneria Informatica spa		**Debora Galante** e-mail: Debora.Galante@eng.it
Engineering Ingegneria Informatica spa		**Pasquale Andriani** Phone: +39 06 8307 4313 e-mail: Pasquale.Andriani@eng.it
Engineering Ingegneria Informatica spa		**Massimiliano Nigrelli** e-mail: Massimiliano.Nigrelli@eng.it
Waterford Institute of Technology		**Jesse Kielthy** Phone: +353 51 304054 e-mail: jkielthy@tssg.org
Telecom Italia S.p.A.		**Valter Bella** Phone: +39.011.228.5643 e-mail: Valter.bella@telecomitalia.com
Telefonica Investigacion y Desarollo SA		**Javier Lucio** Phone: +34 914832756 e-mail: lucio@tid.es
Synelixis Solutions Ltd		**Theodore Zahariadis** e-mail: zahariad@synelixis.com
Institut Polytechnique de Grenoble		**Didier Boëda** e-mail: Didier.Boeda@g2elab.grenoble-inp.fr
Institut Polytechnique de Grenoble		**Raphael Caire** e-mail: Raphael.Caire@G2Elab.grenoble-inp.fr
Orange Labs		**Gilles Privat** e-mail: gilles.privat@orange.com

Orange Labs	**Mengxuan Zhao**
	e-mail: mengxuan.zhao@orange.com

Table of Contents

1. Glossary and List of Abbreviations

This document uses the list of abbreviations and terms defined in the general FINSENY Glossary and Terms [Gloss]. The terms, which are not included in that glossary, are defined below.

Term	Definition
Enabler	According to [FI1], an enabler is a functional building block. Any implementation of an enabler is made up of a set of components which together supports a concrete set of functions and provides a concrete set of APIs and interoperable interfaces.
Capability	Capabilities in this deliverable are software artifacts and/or hardware components that provide particular services, capabilities and test facilities. For FINSENY, those capabilities heavily rely on experiences and results obtained in various other national and international research projects such as BeAware, BeyWatch, Energy@Home, ESB trials, Smart Wheels, eEnergy. Furthermore smart grid specialised test equipment like the ACS Laboratory Infrastructure are relevant capabilities.
Experimentation Lab	It provides a common framework (in terms of methodology, tools and capabilities) for the evaluation, validation and testing of enablers, which will ensure that enablers emerging from various scenarios can be evaluated through a feasibility analysis (based on common criteria) and also partially tested with respect to their practicability. The Experimentation Lab is composed by capabilities, which can be properly configured in order to satisfy the specific needs of the various scenarios.

2. Introduction

2.1 Objective of FINSENY WP8

One objective of FINSENY Work package 8 is the investigation of domain specific enablers by experimenting, prototyping, reviewing and the demonstration of their feasibility towards large scale experimentation as planned in Phase 2.
Those enablers are derived by the requirements and it is important to ensure they are suitable solutions for supporting and enacting the selected scenarios. In this perspective, it is important to analyse, evaluate and validate the feasibility for implementing/instantiating the enablers, based on common criteria (such as relevance and practicability).

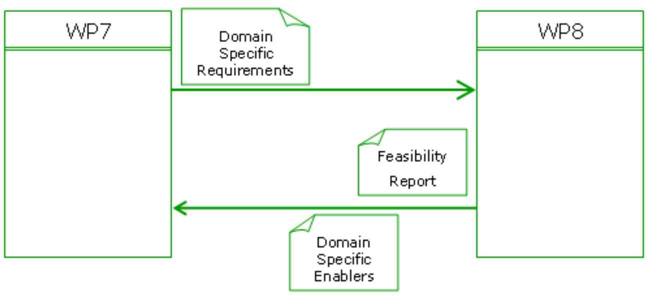


Figure 1: WP7–WP8 interactions in FINSENY

For aiming this objective, the Work package 8, and in particular the task 8.1, is carrying out an investigation of existing capabilities in current EU research project trials and partner labs, and it is setting up an experimentation lab, in order to provide a common framework and support the evaluation and validation phase with tools and methodology.

This document describes the activities of task 8.1 and in particular the set-up of the experimentation lab, its capabilities and configurations and, finally, how to evaluate the feasibility of scenarios and their enablers.

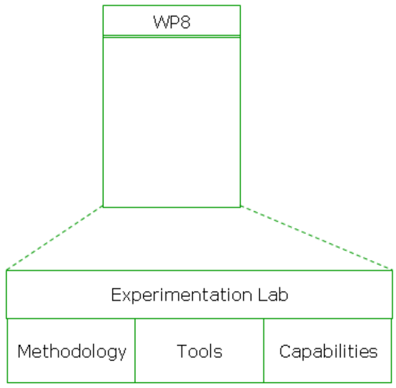


Figure 2: WP8 and the Experimentation Lab

Mainly the experimentation lab provides:

- methodology and procedures for comparing information and reporting the results;
- tools for supporting the methodology and procedures;
- capabilities for supporting partly testing/experimentation.

It is important to highlight that the methodology and tools identified in the common framework will be used along all the lifecycle of the project and in particular for the experimentation phase in Task 8.3 when dealing with the evaluation and validation of domain specific enabler candidates and selecting the candidates that are most relevant to performing implementation trials. It makes sense that evaluation and validation have to be based on standard industry best practice and, eventually, with prototypes evaluation.

2.2 Structure of the deliverable

Focus of this document, which documents the outcomes of the FINSENY Task 8.1, is the detailed description and setup of that experimentation lab, and for this reason, the document is structured in four main sections:

- Chapter 3 “The Rationale behind the FINSENY Experimentation” – which clarifies the idea and the reasons for putting in place the FINSENY Experimentation Lab;
- Chapter 4 presents the “Common Framework” in terms of Methodology and Tools adopted by the FINSENY Experimentation Lab and identifies the requirements for selecting the capabilities (to be included in it);
- Chapter 5 presents the “Selected Existing Capabilities” – based on requirements derived from the methodology, a selection of existing capabilities is considered for composing the Experimentation Lab;
- Chapter 6 presents the architecture components of the Experimentation Lab, its set-up and configurations. Indeed each domain specific scenario, in order to be evaluated, may need its specific configuration of capabilities. Moreover, a list of intended experiments and expected outcomes is provided too.

It is interesting to highlight that chapter 4 provides a simple demonstration of how methodology and tools that are part of the FINSENY Experimentation Lab can be used to select the existing available capabilities to be included in the Lab itself.

2.3 Scope of the Experimentation Lab set up

The above mentioned capabilities represent “(part of) environments / platforms” within the FINSENY Experimentation Lab to be used for testing and trial purposes, and so have to be properly configured for carrying out the evaluation/validation of the different scenarios and their enablers.

Chapter 6 (precisely 6.2 and 6.3) will provide an example of how Lab capabilities can be set up and configured to be used for testing purposes.

3. Rationale behind the FINSENY Experimentation Lab

Identification and definition of domain specific enablers has to be done with respect to the inputs from the WP Consolidation, and specifically the domain specific ICT requirements.
Once the domain specific enablers have been identified and defined, the feasibility analysis has to be carried out through the evaluation, and eventually experimentation, in order to guarantee that those enablers are suitable solutions for enacting the scenarios and satisfying their specific requirements.
Indeed it is a very important pre-requisite for any implementation phase to proceed with the feasibility analysis of the identified enablers and scenarios in order to properly evaluate potential fault and identify corrective actions and plan them, besides providing potentially feedback and details for enhancing the requirements and scenarios.

According to [PMHut], a Feasibility Study Report template is used to provide information about the outcomes and success of a feasibility study. The report should include details on methodology used, the evaluation criteria, options analysed with findings and recommendations resulting from the study.

In this perspective, the experimentation lab has to identify primarily the **methodology**, **guidelines** and **tools** for carrying out the feasibility analysis. Further, the experimentation lab has to identify and provide **acceptance criteria** for the evaluation and validation of the enablers.

Evaluation and validation could require eventually a prototyping phase where some preliminary solutions are sketched and experimented, by allowing to identify potential issues and fault of the depicted scenarios and/or the system to be implemented. For that reason, it makes sense that the experimentation lab supports experimentation through preliminary solutions and it has been considered the option to analyse and select **existing capabilities** from the state of the art in order to match part of required functionalities of the most prominent scenarios in the context of smart energy grid. Indeed, most of the partners involved in FINSENY have already experienced and built interesting solutions in various other national and international research projects such as BeAware, BeyWatch, Energy@Home. As need arises, those capabilities will be available and properly configured for the experimentation phase.
By applying the above outcomes of the experimentation lab, the Work package aims to provide the expected domain specific enablers and the feasibility report and recommendations.
The Figure 3 below summarises the process of the Work package 8, considered as "Specification of Experimentation", its main tasks (the yellow boxes), its outcomes (the document shape) and the interactions with the other FINSENY WPs (i.e. WP7 Consolidation and WP[2..6] Scenarios).

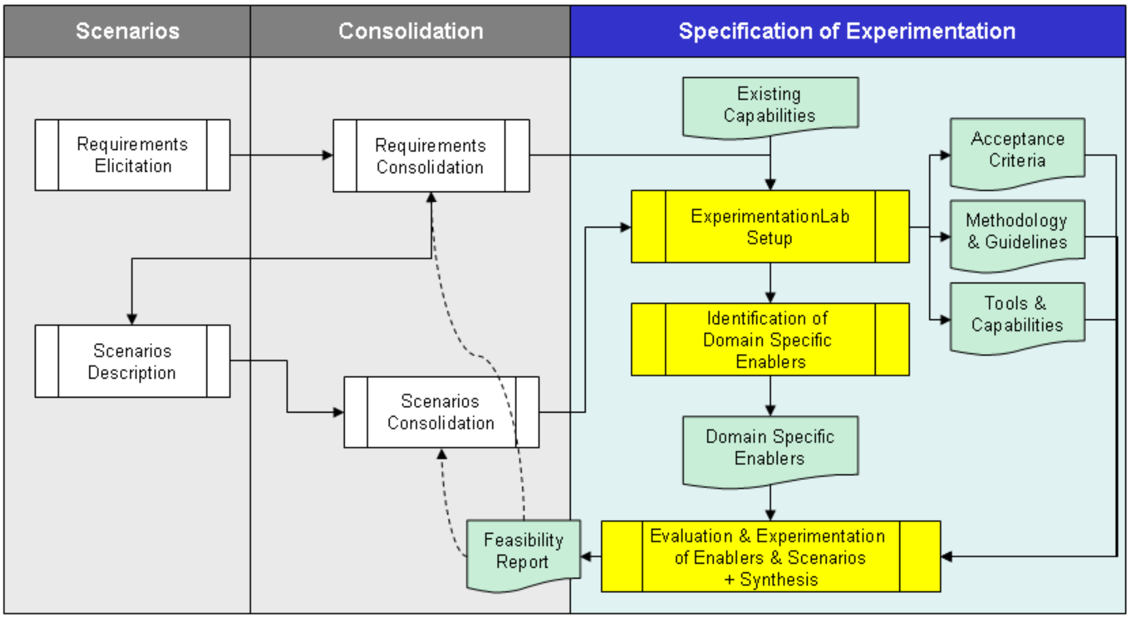


Figure 3: WP8 activities, outcomes and interactions within FINSENY WPs

3.1 Value added to the FINSENY project by the Experimentation Lab

The Experimentation Lab (as a whole) is going to provide the FINSENY project with the following features that substantially represent the real added value to the project itself:

- An environment that perfectly fits testing and experimenting the DSEs that have to be selected during T8.2 and experimented during T8.3; otherwise testing and experimenting the DSEs would be difficult and costly if conducted at third parties laboratory premises. A list of intended experiments and expected outcomes is provided in par. “6.4 Intended experiments and expected results”
- Advantage of sharing FINSENY partners’ knowledge of the technical fields which the DSEs are referring to. This knowledge will be fundamental while carrying forward the experimentation activity and, on the contrary, very difficult and, again, costly if it should be found on the market.
- Availability of “dedicated” experimentation sites and capabilities. Third parties labs would be eventually very strict with slots and times and, so, these constraints would have been reversed on the timetable and schedule of the whole FINSENY project itself thus not permitting any adjustment or variation while in progress. Further details of the experimentation sites and capabilities will be provided in chapter “5 Selected Existing Capabilities”

4. The Common Framework

The feasibility considered in this document is strictly related to analyse the feasibility of the technical implementation of selected scenarios (coming from the Scenarios Work packages) through their relative enablers.

In practice, the feasibility starts from the identification and definition of potential domain specific enablers which are derived by the specific requirements. Then it is necessary to evaluate if those enablers may be objectively considered suitable candidates for supporting the selected scenarios.

In this perspective, the common framework has to allow to **manage the requirements** and the **definition of the potential domain specific enablers**. In particular, it is important that the common framework allows to **trace the enablers and the relative satisfied requirements**.

Further, the common framework, as well as described in above sections, has to provide methodology, tools and capabilities for performing the evaluation. That evaluation is done through a **normalisation of the information** describing the scenarios, their requirements, and the potential domain specific enablers. The normalisation allows to obtain comparable information which may be analysed and evaluated by acceptance criteria: Relevance and the Practicability. Those criteria are mainly functions of key parameters such as: number of supported scenarios, importance of scenarios, availability of supporting functions, evolvability of the components and performances.

For Relevance, this document considers the following definition:

> *Relevance of an enabler (Rel_e) is the degree of capability of an enabler to appropriately support one or more specific (use case) scenarios.*

The definition of Rel_e means that the degree of capability of an enabler has to take into account the availability of proper functions *(f)* to support a number of supported scenarios *(N_s)*, the satisfied requirements (REQ_s) and the importance of scenarios *(W_s)*. So, it is possible to assert that:

$$Rel_e = Rel_e(f,N_s,REQ_s,W_s)$$

In this perspective, the common framework has to provide a **suitable methodology for evaluating the relevance of enablers**, based on the above definition.

For Practicability, this document considers the following definition:

> *Practicability of a specific scenario ($Pract_s$) is the degree of capability of a specific (use case) scenario of being implemented, done, and put in place in a trial.*

The definition of $Pract_s$ means that the degree of capability of a specific scenario of being implemented and enacted has to take into account the availability capabilities implementing proper functions *(f)* to support its specific requirements (REQ_s). The capabilities have to guarantee: i) a degree of evolvability/maintainability (M_c) in order to be adapted/extended/integrated and; ii) that the overall performances (P_c) are satisfied.

So, it is possible to assert that:

$$Pract_s = Pract_s(f,REQ_s,M_c,P_c)$$

In this perspective, the common framework has to provide a **suitable methodology for evaluating the practicability of scenarios**, based on the above definition.

The following sections provide further details on the rationale behind the choices for the selection of methodology (based on best practice), tools and capabilities for supporting the analysis and evaluation of the Relevance and Practicability, based on the above definitions.
In particular, a set of requirements are identified for supporting the methodology and the analysis/evaluation phase and so, the selection of tools and capabilities is driven by those requirements.

For the evaluation of "relevance", as reported in section 4.1, the FINSENY project is considering a methodology (based on the Capability Maturity Model Integration - CMMI[1]) and development tools for analysing and evaluating the requests from the scenarios.

For the evaluation of "practicability", as reported in section 4.2, the FINSENY project is considering the Real-time simulation and the experimentation with real capabilities.

4.1 Methodology and Tools for Relevance

According to the above section, the Relevance of an enabler depends from the functions provided by the enabler for supporting a specific use case scenario and its requirements.
From that definition follows that a suitable methodology for supporting the analysis and evaluation of the Relevance has to be able to **identify the relationships** between the use case scenarios, its requirements, the functions and the enabler. In particular, it is important to trace the relationships among requirements coming from the use case scenarios and the functions implemented by the enablers. Indeed, in the definition of the Relevance of an enabler is based on key parameters such as functions and requirements.

That is a usual task adopted in development process and for that reason a Software Engineering best practice is considered.

The IEEE Standard Glossary of Software Engineering Terminology defines traceability as "the degree to which a relationship can be established between two or more products of the development process, especially products having a predecessor-successor or master-subordinate relationship to one another." [IEEE-610]

Traceability is used to track the relationship between each unique product-level requirement and its source. For example, a product requirement might trace from a business need, a user request, a business rule, an external interface specification, an industry standard or regulation, or to some other source.

Traceability is also used to track the relationship between each unique product-level requirement and the work products to which that requirement is allocated. For example, a single product requirement might trace to one or more architectural elements, detail design elements, objects/classes, code units, tests, user documentation topics, and/or even to people or manual processes that implements that requirement.

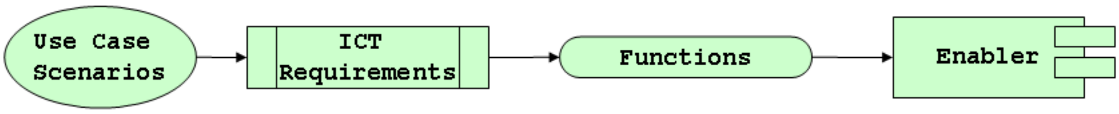


Figure 4: Traceability Chain

Good traceability practices allow for bidirectional traceability, meaning that the traceability chains can be traced in both the forwards and backwards directions.
Forward traceability ensures proper direction of the evolving product (that we are building the right product) and indicates the completeness of the subsequent implementation.
Backwards traceability helps ensure that the evolving product remains on the correct track with regards to the original and/or evolving requirements.

[1] Capacity Maturity Model Integration – CMMI, in software engineering and organizational development, is a process improvement approach that provides organizations with the essential elements for effective process improvement. CMMI is registered in the U.S. Patent and Trademark Office by Carnegie Mellon University.

By tracing each “item” of the traceability chain, during the whole development life-cycle, it is possible to identify the relationships (in terms of UML relationships such as implement, depend, trace, and association) between the use case scenario, the requirements, the functions and their enablers. The relationship allows evaluating the relevance.

Traceability and Impact Analysis

Forward and backward traceability can also be used to assess the impact of requirement changes. If there are changes in the business environment (and usually they come), then if good forward traceability has been maintained, that change can be traced forward to the associated requirements and all of the work products that are impacted by that change.

Impact analysis is a core activity of requirement change management (RCM). The pre-requisite of impact analysis activity is traceability information.
It is possible to minimise the risk of introducing undesirable effects to the system by increasing the understanding of how the proposed change might affect the elements in the system.
Further, it is possible to perform an impact analysis to investigate the potential consequences of a change before that is applied to a system, and you can browse the dependency relationships in a diagram. Visualising dependency relationships between system elements can help the evaluation/validation phase to understand the dependencies that exist between system elements and how the system works.

In this perspective, the traceability information and the impact analysis can support the evaluation/validation phase in identifying and planning suitable alternative solutions caused by the change in requirements and/or any other condition (e.g. availability of implementation of a sub-system) in the development process.

Traceability Matrix and Bidirectional Traceability

Tracking all the requirements outlined in the functional specification document and checking whether all the requirements have been met by the enabler can be a cumbersome and a laborious process. Not surprisingly, many enterprise quality systems, such as CMMi, Six Sigma or ISO 9000, require organizations to have formal traceability procedures.

Traceability Matrix is an industry-accepted format for tracking requirements. It provides a convenient format that helps to visually represent associations between user requirements and the work products developed and implemented.

In the requirements engineering field, traceability is about understanding how high-level requirements - objectives, goals, aims, aspirations, expectations, needs - are transformed into low-level requirements.

A traceability matrix is a document, usually in the form of a table.

Common usage is to take the identifier for each of the items of one document and place them in the left column. The identifiers for the other document are placed across the top row. When an item in the left column is related to an item across the top, a mark is placed in the intersecting cell. The numbers of relationships are added up for each row and each column. This value indicates the mapping of the two items. Zero values indicate that no relationship exists.

Bidirectional traceability is the ability to trace both forward and backward (i.e., from requirements to enabler and from enabler back to requirements).

4.1.1 Tools for Supporting the Evaluation of Relevance

The FINSENY project is adopting UML and Enterprise Architect [EA] for its modelling tasks and managing the whole design phase.

Unified Modelling Language (UML) is a standardised general-purpose modelling language in the field of object-oriented software engineering. The standard is managed, and was created, by the Object Management Group. It was first added to the list of OMG adopted technologies in 1997, and has since become the industry standard for modelling software-intensive systems.

UML includes a set of graphic notation techniques to create visual models of object-oriented software-intensive systems.

Enterprise Architect is a Model Driven UML tool which includes the following features:

- Diagrams for modelling strategic and business level concepts;
- Domain-specific profiles and reusable model patterns;
- Baseline and version management for tracking and integrating changes;
- Role-based security to help the right people contribute in the right way.

With built-in requirements management capabilities, EA supports designers to trace high-level specifications to analysis, design, implementation, test and maintenance models using UML, SysML, BPMN and other open standards.
And using high quality, built-in reporting and documentation, it is possible to deliver a truly shared vision easily and accurately.

EA has powerful tools to help visualize requirements and integrate them into the development environment such as: Visualize Requirements, Model Use Cases, **Trace Requirements through to Implementation, Analyze the Impact of Change and Share a Common Glossary.**

In particular those features of EA support the development and evaluation phase for the full traceability.

Full Traceability support through Enterprise Architect

Traceability is the means of capturing implementation and dependency relationships in the model. For example, a business process will require some system functionality (use cases) to implement the process functions. Enterprise Architect allows to capture this information by using Realization links.
Use Analysis diagrams associated with processes, use cases, classes & etc to capture the realization relationships.
Implementation details are usually placed in Analysis diagrams under the principal model element (in the picture below). Once these relationships are made, they can be queried using the Implementation and Dependency details available from the Project Browser context menu.

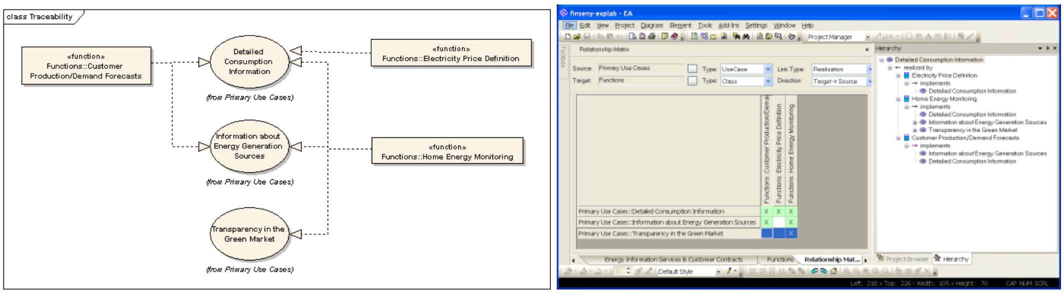


Figure 5: Traceability and Traceability Matrix: Enterprise Architect tool

4.1.2 How to apply the methodology and tools: a practical sample

In order to demonstrate how the methodology and tools of the Experimentation Lab may be applied for evaluating the relevance, the Electronic Market place for Energy use case scenarios are being considered.

By gathering the information from D6.1, where the use case scenarios have been specified, and the D6.2, which identifies preliminary functions, it is possible to define the following traceability matrix. That latter allows to trace the relationships between the 3 Market Areas (i.e. Energy Information Services & Customer Contracts Market, Demand Side Management Market, Local Energy Market), their use case scenarios and the implementing functions. The columns coloured in grey remark functions potentially fitting with the FI-WARE Generic Enablers.

Use Case Scenarios	FUNCTIONS: Electricity Price Definition	Provide Schedules and Balance Reports	Customer Production/Demand Forecast	Contract Management	Final Customer Habits Profiling	Customer Energy Information Service	Monitor Electricity Network	Home Energy Monitoring	Energy usage optimization	Pre-paid Contract Services	Balance Energy	DSM Contract Management	Shape Demand	Market Mechanisms for Solving Grid Issues	Aggregated Production/Demand Forecast	Aggregated Energy Monitoring	Energy Trading
Energy Information Services & Customer Contracts Market																	
Transparency in the Green Market			✓	✓		✓		✓									
Information about Energy Generation Sources	✓					✓		✓	✓								
Detailed Consumption Information	✓					✓		✓	✓								
Colored Ethical Bid				✓	✓	✓											
Energy Contract Brokering		✓		✓													
N_s	2	1	1	3	1	4	0	3	2	0	0	0	0	0	0	0	0
Demand Side Management Market																	
Trading flexible capacity							✓							✓			
Flatten Demand Curve	✓		✓					✓	✓								
N_s	1	0	1	0	0	0	1	1	1	0	0	0	0	1	0	0	0
Local Energy Market																	
Trading for the Good of All										✓				✓	✓		
Supplier Side Local Trading											✓				✓	✓	✓
Incentive based green market				✓	✓	✓									✓	✓	✓
Energy market for neighborhoods		✓															
N_s	0	1	0	1	1	1	0	0	0	1	1	0	0	1	3	2	2

Table 1: Traceability Matrix: eMarket4E Functions supporting Use Case Scenarios

As shown in Table 1, the traceability matrix allows to identify the potential and relevant functions/capabilities which support the implementation of the use case scenarios. In particular, for each function, the matrix allows to see the numbers of supported use case scenarios (N_s), which is one of the

key parameters for estimating the Relevance (Rel_e). The table shows how for different scenarios may be identified key functions.
Enterprise Architect, the modelling tool adopted by FINSENY, allows to support the traceability analysis as shown in the picture below.

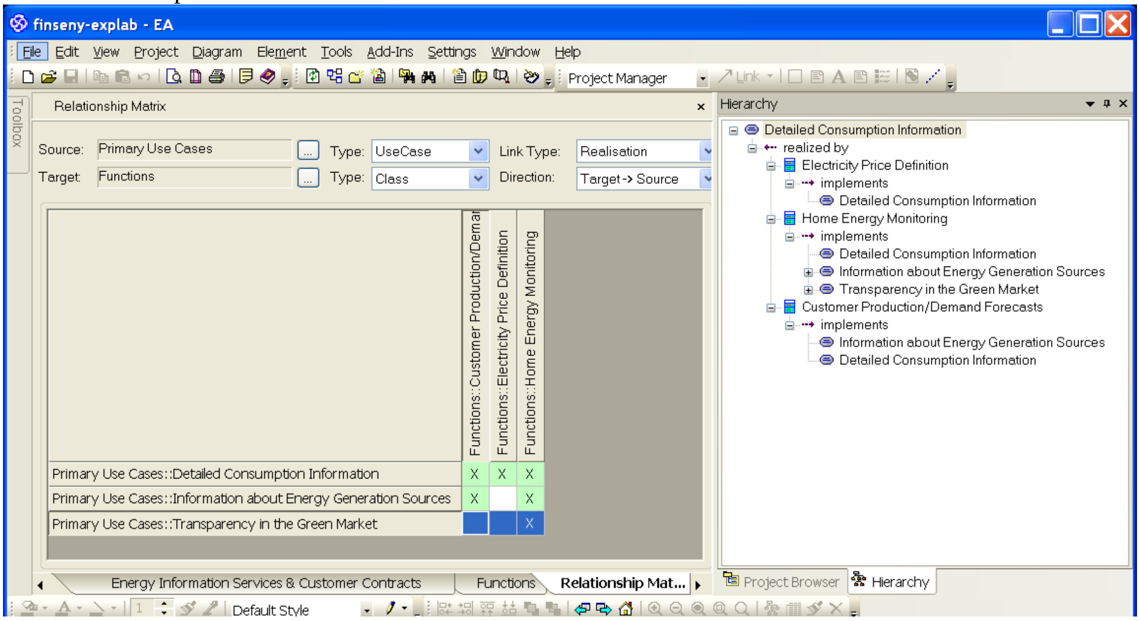


Figure 6: Traceability Matrix and Hierarchical View on Enterprise Architect

The traceability matrix allows to identify the most relevant functions supporting the scenarios. Those functions are consequently traced by the tool in the modelled capabilities (the ones from the existing capabilities).
In a similar way, the traceability process allows to identify the relationships among the requirements of the use case scenarios and the supporting functions, as shown in the table below.

	FUNCTIONS																
Reqs	Electricity Price Definition	Provide Schedules and Balance Reports	Customer Production/Demand Forecast	Contract Management	Final Customer Habits Profiling	Customer Energy Information Service	Monitor Electricity Network	Home Energy Monitoring	Energy usage optimization	Pre-paid Contract Services	Balance Energy	DSM Contract Management	Shape Demand	Market Mechanisms for Solving Grid Issues	Aggregated Production/Demand Forecast	Aggregated Energy Monitoring	Energy Trading
TRANSPARENCY IN THE GREEN MARKET																	
Info communication system for the Final Users						✓	✓										
Energy Management System				✓													
Format for information exchange has to be standardized																	
Need of HTTPS connection on IPV6 for wired approach						✓	✓										
Need of GSM/GPRS/HSDPA systems for mobile						✓	✓										

approach																	
Need of WSAN (Wireless Sensor & Actuator Network) for Smart Meter communications						✓	✓										
COLORED ETHICAL BID																	
Real time system to monitor the available kind of energy for the energy retailers	✓			✓	✓	✓						✓					
High portability applications for fixed and mobile services targeted to the final customer					✓	✓											
User Software Agent System					✓												
A HW/SW network of grid operators to offer different kind of energy						✓						✓					
TRADING FOR THE GOOD OF ALL																	
Weather Forecast Service										✓							
Energy information provider										✓							
Cloud computing for high speed data processing										✓							

Table 2: Traceability Matrix: eMarket4E Functions supporting Requirements

Enterprise Architect allows to support this traceability among Use Case Scenarios, ICT Requirements and Functions in UML. The traceability may be reported as a hierarchical tree or traceability matrix, as shown in the pictures below, which refer to the Transparency in the Green Market use case scenario.

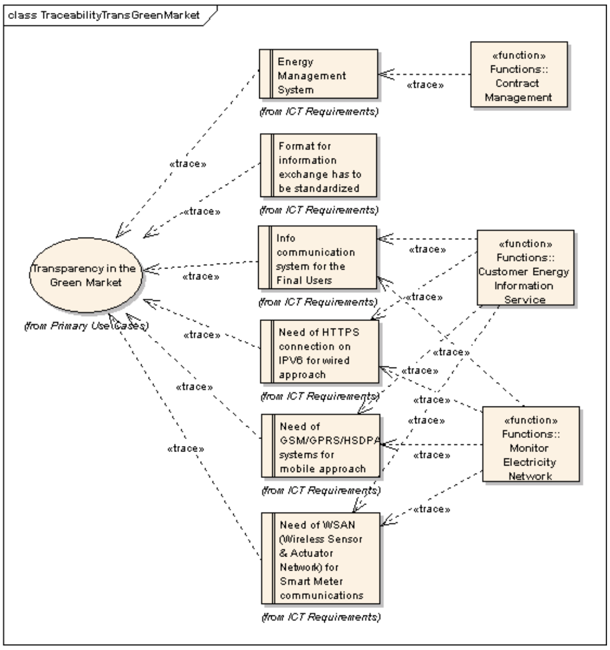


Figure 7: Tracing the relationships among use case scenarios, ICT requirements and functions

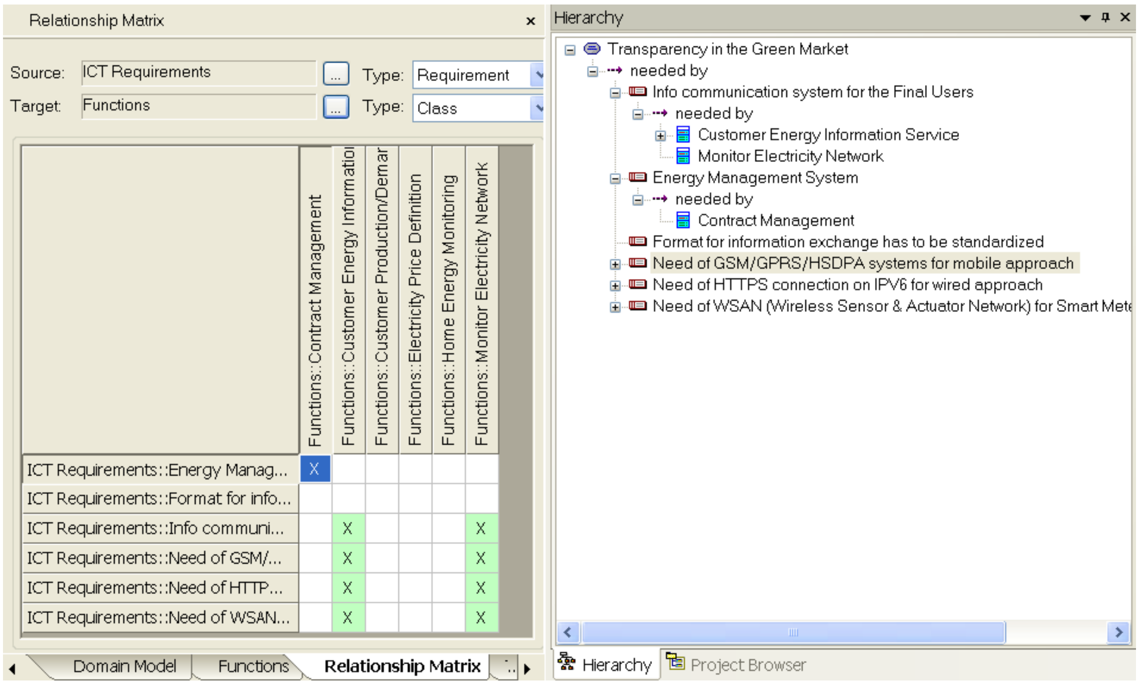


Figure 8: Traceability Matrix for Transparency in the Green Market use case scenario

In this perspective, the traceability matrix and modelling provided by Enterprise Architect allow the evaluator to identify the key parameters (through the “trace” relationship) for estimating the Relevance of a candidate enabler. Indeed, Requirements, Functions and Number of Scenarios are traced. In D6.2 an exercise for identifying candidate enablers and their preliminary Relevance has been done by considering the fitting functions of the FI-WARE Generic Enablers, as shown in the table below.

Functions	Enabler: FI-WARE Generic Enabler									
	Applications/Services Ecosystem and Delivery Framework		Security		Data / Context Management		Internet of Things (IoT) Services Enablement		Interface to Networks and Devices	
	USDL Service Descriptions	Marketplace	Identity management GE	Privacy GE	Query Broker	Publish / subscribe broker	Data handling	IoT Data handling	Connected Devices Interfacing (CDI)	Cloud Edge
Electricity Price Definition		✓						✓		
Provide Schedules and Balance Reports	✓									
Customer Production/Demand Forecast									✓	
Final customer Habits Profiling	✓									✓
Home Energy Monitoring	✓								✓	
Pre-paid Contract Services		✓								
Balance Energy								✓		
Energy Trading		✓								
Customer Energy Information Service			✓	✓	✓	✓	✓			
Energy Usage Optimisation										
Contract Management										
DSM Contract Management										
Market Mechanisms for Solving Grid Issues		✓				✓				
Monitor Electricity Network						✓				
Shape Demand										
Aggregated Energy Monitoring					✓	✓		✓		
Aggregated Production / Demand Forecast										
Preliminary Relevance	**41%**		**5%**		**23%**		**23%**		**18%**	

Table 3 – FI-WARE Generic Enablers fitting eMarket4E functions

Potential capability for supporting one of the scenarios is the FI-WARE Generic Enabler “Interface to Networks and Devices” which provides the “Home Energy Monitoring” function. Anyway, it could be reasonable to consider an alternative plan in case of delayed release of the function from FI-WARE. For that reason, it is considered the adoption of Energy@Home or BeAware capabilities which provide that function and, the adoption of the ACS RT-Simulator for experimenting the changes of load. Traceability matrix supports in this perspective also the evaluation of alternative solutions, by taking into account the requirements and the functions required by the scenarios.

Evaluation of relevance

In order to evaluate how well the traceability process retrieved topically relevant results, the relevance of the retrieved results must be quantified.

Relevance levels can be either binary (indicating that a result is/is not relevant) or graded.

In table 2 assigned relevance levels are binary and the measure of relevance has been expressed as the number of functions provided by the Generic Enablers which are estimated to be relevant over the total number of functions (identified in the Uses Cases scenarios within the WP6).
This methodology is based on the assumption that all the functions taken in to account as got the same weight of importance for the scenario those refer to. Then, the evaluator could either decide to consider some use cases that are more “important” for a specific scenario or to give a different weight to each function.

4.2 Methodology, Tools and Capabilities for Practicability

According to the definition of Practicability in section 4, it depends from the availability of existing capabilities to implement the expected functions for properly supporting the specific requirements of the scenario. Further, in case of existing capabilities, it is important to consider the order of evolvability/maintainability of the implemented functionalities. Last but not least, it is important to evaluate that the overall performances are satisfied.

As well as the evaluation of Relevance, the evaluation of Practicability is widely documented in the Software Engineering and it is strongly related to the Development Process, and in particular it is mentioned in the CMMI Verification and Validation Process Areas.

The Verification process area ensures that selected work products meet the specified requirements. The Verification process area expects that a verification strategy is developed to ensure adequate verification. This verification strategy should be highly integrated with the Technical Solution process area and the Product Integration process area. It is generally an incremental process starting with product component verification and usually concludes with verification of fully assembled products.

The Validation process area validates products against the customer’s needs. Validation may be performed in the operational environment or a simulated operational environment. Coordination with the customer on the validation requirements and the validation strategy is one of the most essential elements of this process area.

In practice, Validation and Verification can be performed in the operational environment and/or a simulated operation environment. In this perspective, FINSENY considers this best practice for supporting the validation and verification of the practicability.

In the field of real-time systems, particularly those deployed in remote or hostile environments, it is often challenging to debug software in the field.
Real-time systems have their own peculiarities and problems such as latencies caused by page-faults, file handling locks, shared memory issues etc.
Increasingly, ICT systems are being used in applications where a failure of the system can result in loss of life or catastrophic damage.

In many cases, those criticisms have to be considered, according to the preliminary requirements and scenarios identified in the FINSENY Use Case Scenarios.

Consequently, the need has arisen for a theoretical basis and practical methodology by which the correct design and construction of such systems can be achieved.

These systems are often referred as hard real-time systems, where real-time reflects the fact that they must directly interact with a changing physical environment and hard refers to the fact that at least some system functions must be performed within specific timing constraints.
Conventionally, hard real-time system simulators are individually designed for a specific architecture and application.

For hard real-time systems, the verification of the correct and safe functioning of the system, specifically the guaranteed meeting of timing requirements, is of paramount importance.

Past engineering practices have dictated that systems are verified on a testbed: arguments regarding the timeliness of the system are established by using test results. This approach is not ideal. Typically, systems fail, in a timing sense, in very constrained circumstances, with the average-case, as often seen on the testbed, performing adequately.

To counter this problem, much research has been concerned with developing design methodologies and feasibility theory to ensure timing requirements can be verified offline.

A common approach to the implementation of hard real-time systems is to use a cyclic executive, where tasks are executed according to a predefined schedule which is held in the system, and which is repeated indefinitely. The execution order is calculated off-line in advance.

Such an approach has a number of problems: schedules are inflexible and may become prohibitively long. More flexible approaches include:

- Static priority scheduling: all tasks are assigned (offline) a priority; with the highest priority runnable task executed at any time;
- Dynamic priority scheduling: tasks are assigned priorities at run-time, with those priorities able to change dynamically.

4.2.1 Tools for Supporting the Evaluation of Practicability

Over the last two decades, commercially available computer has become both increasingly powerful and increasingly affordable. This, in turn, has led to the emergence of highly sophisticated simulation software applications that not only enable high-fidelity simulation of dynamic systems and related controls, but also automatic code generation for implementation in industrial controllers.

Used in conjunction with the current generation of real-time simulators, these simulation software applications form the basis of the 'Model-Based Design' paradigm; a control design methodology that is centred on the use of reference system models. In the Model-Based Design (MBD) approach, initial modelling and requirements, early controller prototypes, production code generation, production controller testing and integration are all derived from reference models. The approach has the objective of accelerating the design cycle through the early detection of design flaws and other problems. In the automotive industry, in particular, the MBD method is a de-facto standard. MBD is also quickly gaining acceptance and being adopted by design engineers in a large number of industries.

Real-time simulation, based on automatic code generation, is used in many engineering field and applications such as: aircraft flight control design & validation, industrial motor drive design, complex robotic controller design and power grid statistical protection tests.

These applications benefit from the use of real-time simulators in a number of ways:

- First, real-time simulation produces a set of requirements and specifications that can be used by all disparate teams/subcontractors involved in a project.
- Secondly, it enables testing of simulated devices at or beyond their normal operating limits without the risks involved with testing of real devices, especially when high power levels are present.
- Third, it is easier and less risky to test fault responses on a simulated model. Finally, the simulation acceleration factor obtained by the use of compiled code (instead of the interpreted code used by most simulation tools) enables the realization of rapid batch simulations.

Furthermore, the Real-time simulators are used extensively in many engineering fields and in this perspective, the adoption of Real-time simulation can be considered a "de-facto" best practice in the development life-cycle.
In particular, the Real-time simulation is widely adopted for the refinement of requirements, specification of the scenarios and of their supporting functional architectures. So, this approach can provide benefit to the whole development process.

The above reasons strongly justify the choice of adopting Real Time Simulation for the evaluation of practicability of the scenarios.
In the next sections, some Real Time Simulators are identified and described.

4.2.2 How to apply the methodology and tools: a practical sample

In order to demonstrate how the methodology and tools of the Experimentation Lab may be applied for evaluating the practicability, the Smart Building use case scenarios are being considered.

There are a few generic context simulators available that can be used as frameworks and codebase for a building simulator.

In context simulators, emphasis is placed on the interaction between different entities. Context parameters change because of the influence by agents, and agents make adaptations to the context as well. There are some characteristics in common:

- Capability of modelling the whole context which include:
  - Physical entities, such as people, electrical appliance, room, building, etc;
  - Sensors/actuators;
  - Control systems;
  - Influence/interactions between entities;
  - Evolution in real time. As interactions will influence the behaviour of the context on the next moment, real-time is essential for this kind of simulations;
  - Result data can be exported to be exploited further.

According to the needs of the FINSENY project, the criteria for the choice of the simulator should be the following:

- Modelling of different types of entities in different complexity: from a simple electrical appliance (i.e. a hair-dryer) to a complex control system;
- Reaction in real time;
- Input/Output data are exploitable;
- Ideally, with a user-friendly GUI to describe the scenario and observe the results.

A simulator can be used at different levels, as represented here, with either a complete simulation integrating on the basis of physical component models, or a simplified simulation using abstract models of physical entities of the building.

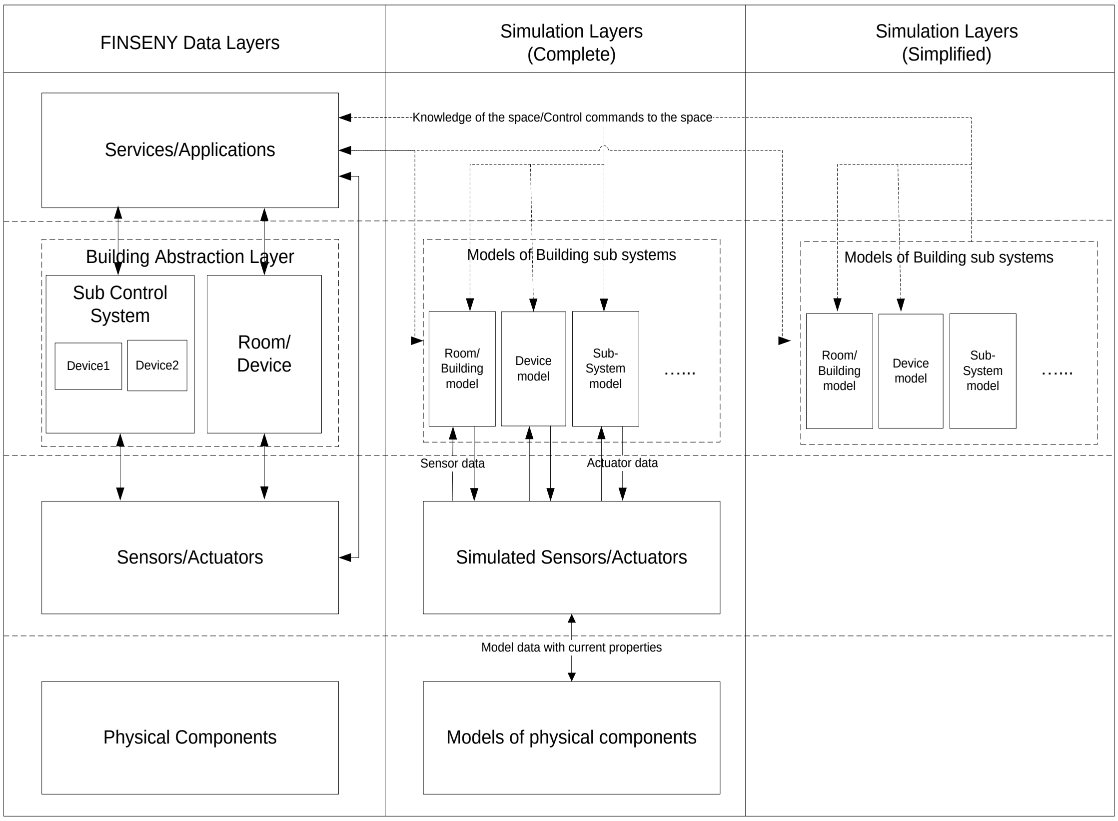


Figure 9: Schema of the Simulation Layers

A complete simulation begins at the lowest level of the data stack, where are implemented the specific models for each physical component, such as a human being, a desk, the room space, etc.
Simulated sensors/actuators are presented in the simulator as data interface between the specific models of physical components and those in the building abstraction layer.
In the higher level, models are of more abstraction and depend less on present physical components. The fact that a model can represent one category of physical components allows simulating a context only with the parameters which interest us. Impact of the interactions between different models is memorized as parameters in each model. The simulation at this level provides as output the information of the

context, such as the state of the building/room, the position of key objects in the space, etc. Services and applications take the information from the simulation and do further data processing.

The simulation does not need to take the whole data stack, which means in some cases, it will be sufficient to use some auto-generated data of the abstract models to run the simulation. An example of simulator is presented in the following:

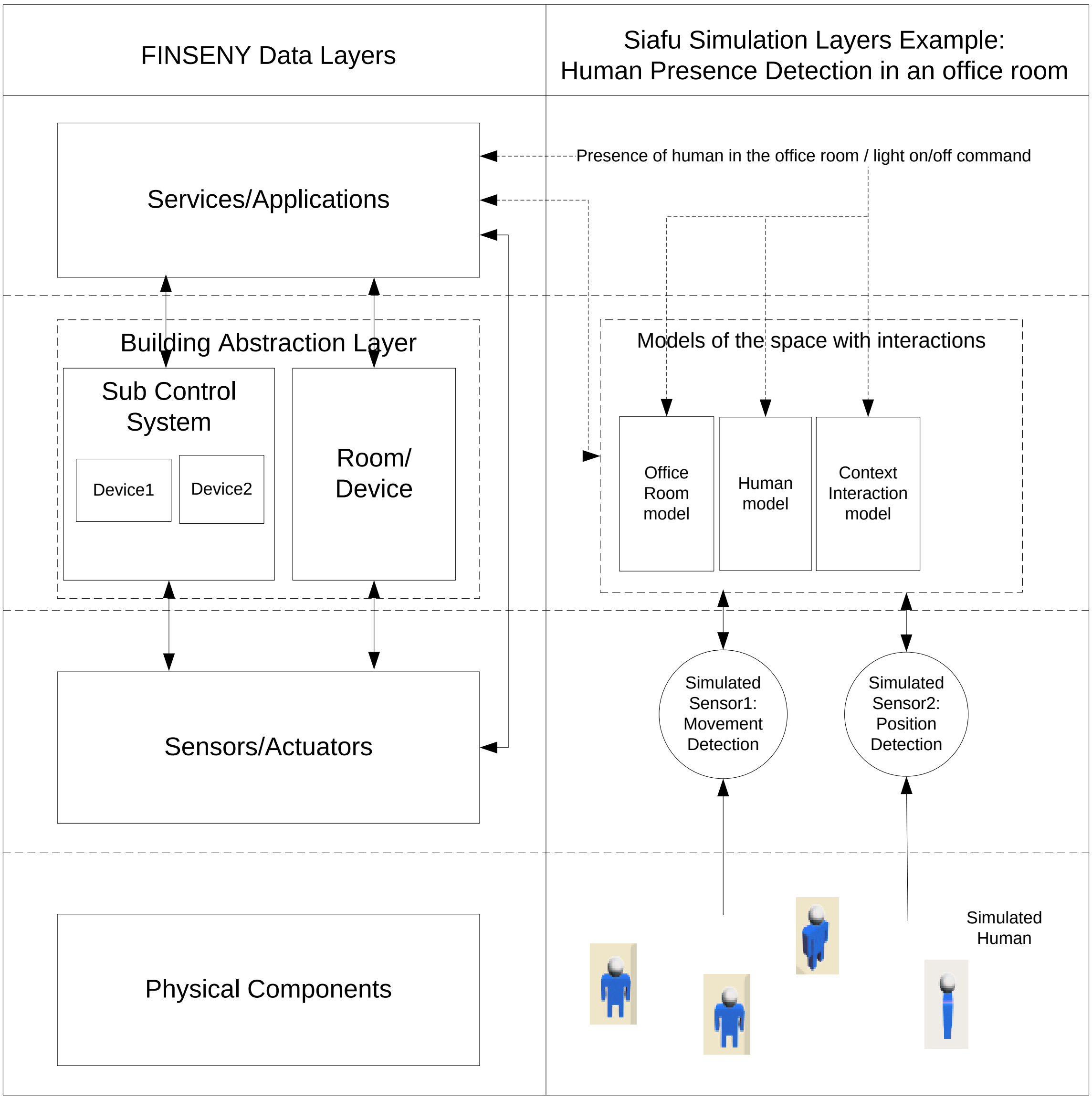


Figure 10: Siafu simulation layers

Siafu is an open source context simulator. It is capable to model agents in a context and simulate the whole system.
In this example, each present person is simulated by the specific model "human". Sensors and actuators detect presence and movement of these people and provide to the context simulation, where the office room and people are modelled more abstractly. From the received data, the models are capable to tell the presence of people in the office room. Based on this information of the room provided by the simulation in the lower level, services and applications are capable to give commands to the context, such as turn on/off the light.

Simulator and domain-specific enablers
The goal of WP8 is to demonstrate the feasibility of the domain specific enablers (DSE) towards large scale experimentation. While making choice of the simulator, the validation of these DSE should be taken into consideration.
Here is the list of DSEs in the Smart Building Domain:

- DSE;
- Non-application-specific supervisory control;
- Non-application-specific UI toolkit;
- Virtualization of building entities: One-to-many and many to one mappings;
- Security at the entity Level;
- Repository of models for physical entities;
- Optimization engine.

The main role of supervisory control is to coordinate individual controllers in order to satisfy the common goal of the global system. In the case of smart building or smart home, the supervisory control supervises the state of each entity (device, space, sub control system, etc), and give control commands based on the observed state and the pre-defined system constraints and properties.

Let’s take a room as an example. In one house, the electricity consummation cannot exceed 10A at anytime. If the user wish to use several appliances which would consume more than 10A if all are on, the supervisory controller will look at the current electricity usage, the state of the appliances and the their priority defined by the system. Based on the criteria mentioned above, the supervisory controller will coordinate the controller of each appliance, turn on some of them when the most priority ones finish in order to meet the system’s constraints.

The advantage of applying a supervisory controller is that the global system can adapt itself to the environment and to the interactions between sub-entities as soon as possible.
The validation of the supervisory control is a key step for the validation of the global Experimental Lab. The whole system is a very complex one which is the composite of many sub systems. The sub systems could be a simple controller (i.e. the controller of an appliance in room) or could be a complex one as well (i.e. the heating control system under the global building energy control system). The supervisory controller is above all the component controllers to make sure that they function together and respect the rules. If it is not validated, the system risks going out of its normal behaviour.

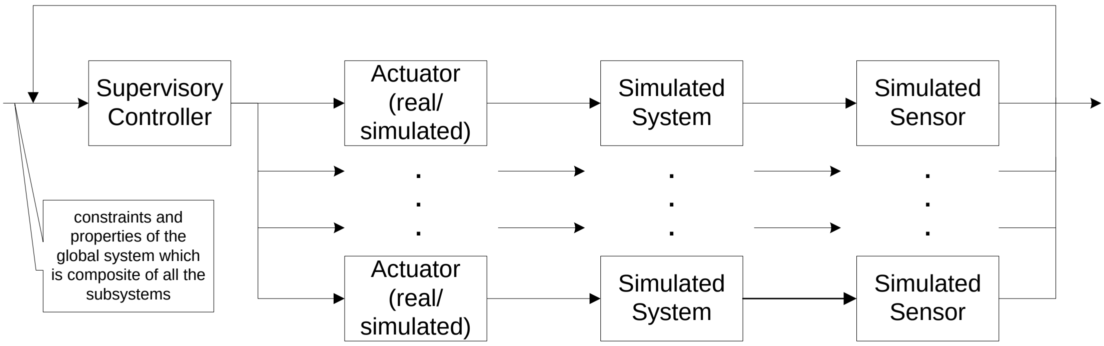


Figure 11: Schema for Validation of Supervisory Controller

The diagram above explains the principle of the validation of such a supervisory controller. The supervisory controller takes the outputs of simulated sensors and computes the commands for corresponding actuators which could be simulated as well or real ones. The simulated systems (the “subsystem” in the paragraph above) then perform as told by the actuator and output performance results to sensors. The validation is done at the output of the simulated sensors.
A simulator must represent the real system, that’s why the supervisory controller must be modelled and integrated in the simulator once it is validated. The supervisory controller could be mapped to the Building Abstraction or the Service layer of the FINSENY Data Stack according to further discussions.

4.3 Experimentation Lab Requirements

According to the above sections, here is reported a summary of the requirements for the Experimentation Lab. That allows to drive the selection of its capabilities.

ID	Requirement Title	Description	Priority
EXPLAB-REQ-001	Manage the requirements	The evaluator can manage the requirements of the use case scenarios in order to derive the domain specific enablers.	Mandatory
EXPLAB-REQ-002	Identification and Definition of the potential domain specific enablers	The evaluator can identify and define domain specific enablers for supporting the use case scenarios.	Mandatory
EXPLAB-REQ-003	Trace the enablers and the relative satisfied requirements	The evaluator can trace the enablers supporting the use case scenarios and the relative satisfied requirements.	Mandatory
EXPLAB-REQ-004	Normalisation of the information	The normalisation allows to obtain comparable information which may be analysed and evaluated by acceptance criteria: Relevance and the Practicability.	Mandatory
EXPLAB-REQ-005	Suitable methodology for evaluating the relevance of enablers	The evaluator needs to be supported by a methodology (based on an industry best practice) and development tools for analysing and evaluating the relevance of enablers. The identified methodology is the traceability.	Mandatory
EXPLAB-REQ-006	Suitable methodology for evaluating of practicability	The evaluator needs to be supported by methodology, tools and capabilities for analysing and evaluating the practicability of a scenario. The identified methodology is the real-time simulation and experimentation.	Mandatory

Table 4: Summary of the Experimentation Lab requirements for the selection of the capabilities

In section 4.1.1 has been identified the Enterprise Architect tool which supports the methodology Traceability and the Traceability Matrix.
Furthermore, the section 4.1.2 has clarified how the methodology and tool can be adopted for the analysis and evaluation of the Relevance of enablers.
For the analysis and evaluation of the practicability, it has been identified in section 4.2 the approach of experimentation and real-time simulation. In particular, that section has clarified that a set of capabilities are needed such as sensors/actuators, services, models/data from building and human behaviour. For those capabilities, the FINSENY project has selected existing solutions/capabilities. Those are reported in the next section.

5. Selected Existing Capabilities

This section describes the selected existing capabilities that provide particular services and satisfy the expected requirements of the Experimentation Lab, as reported in the above section.
In order to select the existing capabilities, the FINSENY project, as mentioned in DoW, has considered the experiences and results obtained in various other national and international research projects. For gathering and selecting the capabilities it has been adopted the template described in the above section.

In order to gather the information of the Existing Capabilities, FINSENY is adopting a template. That aims to gather in a structured format the information of the existing capabilities from the experiences of EU research projects and trials and labs of the partners. The capabilities are always selected by considering the scenario work packages and the requirements expected from the methodology for analysing/evaluating the acceptance criteria. Further, the template allows to provide a normalised information of the existing capabilities.

<table>
<tr><td>FINSENY Partner Commitment</td><td colspan="3"><Partner name + commitment statement></td></tr>
<tr><td colspan="4">Contacts & Main information</td></tr>
<tr><td>Acronym & Name</td><td colspan="3"></td></tr>
<tr><td>Leading Company or Authority</td><td colspan="3"></td></tr>
<tr><td>Involved FINSENY partner</td><td colspan="3"><short names of FINSENY involved partner></td></tr>
<tr><td>Type</td><td colspan="3"><Product? Research project? finance by?></td></tr>
<tr><td>Web site</td><td colspan="3"></td></tr>
<tr><td>Location/Countries affected</td><td colspan="3"></td></tr>
<tr><td>Running period</td><td colspan="3"></td></tr>
<tr><td colspan="4">Facilities description</td></tr>
<tr><td>Technology keywords</td><td colspan="3"><keywords about domain, technology used></td></tr>
<tr><td>Standards</td><td colspan="3"></td></tr>
<tr><td>Involved actors/user</td><td colspan="3"></td></tr>
<tr><td>Brief scenarios description</td><td colspan="3"></td></tr>
<tr><td>Running experimentation (if exists)</td><td colspan="3"><how the experimentation has been organized></td></tr>
<tr><td>Number of involved users during experimentation (if any)</td><td colspan="3"></td></tr>
<tr><td>Is any Experimentation Lab still running?</td><td><Yes? No?></td><td>Until?</td><td><Date></td></tr>
<tr><td colspan="4">Involved Technologies</td></tr>
<tr><td>Brief Technical description</td><td colspan="3"><how the involved technologies are deployed? ></td></tr>
<tr><td>Adopted technical solution (potentially domain specific enablers?)</td><td colspan="3"><describe on which hardware/software components the pilot relies>
…
…
…</td></tr>
<tr><td>Picture (Architecture, Interfaces, etc.)</td><td colspan="3"></td></tr>
<tr><td>Software availability</td><td colspan="3"><describe if software/hardware are available></td></tr>
</table>

Table 5: Template for facility and lab description

5.1 ACS Laboratory Infrastructure

FINSENY Partner Commitment	**RWTH Aachen University** **The institute for Automation of Complex Power Systems is committed to bring in the expertise and laboratory infrastructure for real-time simulation of the electrical grid in conjunction with communication systems for data exchange**		
Contacts & Main information			
Acronym & Name	ACS lab		
Leading Company or Authority	RWTH Aachen Unversity		
Involved FINSENY partner	RWTH Aachen University		
Type	Laboratory Equipment		
Web site	http://www.eonerc.rwth-aachen.de/aw/cms/website/zielgruppen/acs/research_aps/~svp/equipment_acs/?lang=en		
Location/Countries affected	Aachen, Germany		
Running period	Equipment available since 2009		
Facilities description			
Technology keywords	Real-time simulation of electrical network, monitoring, control, distributed control, home energy systems, hardware-in-the-loop, power-hardware-in-the-loop		
Standards	IEC 61850 GSE binary messaging, IEC 61850-9-2 sampled values, IEEE C37.118, DNP, OPC,		
Involved actors/user	• Components of electrical network; • Control platform.		
Brief scenarios description	• Real-time simulation of complex power system (RTDS®); • Real-time simulation of distributed system (e.g. simulation of wind farms and representation of wind field distribution) (DSP Cluster); • Real-time simulation of multi-physic system (e.g. home energy system) (PC cluster).		
Running experimentation (if exists)	• Simulation of electrical grid; • Power Hardware in the loop; • Distributed control approaches; • Communication emulation.		
Number of involved users during experimentation (if any)	Experimentation in laboratory and testbed facilities.		
Is any Experimentation Lab still running?	YES	*Until? No end date*	
Involved Technologies			

Brief Technical description

Figure 12: ACS Simulation and Testing Infrastructure

The main goal of the ACS laboratory infrastructure (RWTH Aachen, Simulation and Testing Infrastructure of institute for Automation of Complex Power Systems) is to support fast prototyping of control systems and de-risk new control architectures for application in Power System.
This goal is achieved through a set of real-time (RT) simulators and hardware interfaces.
Considering that the research focus is on complex systems and knowing that simulating different systems and situations may require different hardware and software solutions, it is impossible to choose one single platform for the real-time simulator that fulfils all the requirements. Thus ACS designed an own laboratory structure that accounts for all the scenarios of interest and satisfies all foreseeable needs of the institute.

The concept is to have three different simulation platforms in order to support:

- Real-time simulation of complex power system (RTDS®);
- Real-time simulation of distributed system (e.g. simulation of wind farms and representation of wind field distribution) (DSP Cluster);
- Real-time simulation of multi-physic system (e.g. home energy system) (PC cluster).

For what concerns the real time simulation of complex power system, the institute purchased eight racks of RTDS®, which is the state of the art in RT simulation of power systems. This installation represents a unique facility for the institute being the biggest in Europe and one of the biggest all around the world.

The RTDS® Simulator, which is a fully digital electromagnetic transient power system simulator, builds the core of the ACS simulation platform is. Of paramount importance is the fact that the simulator works in continuous, sustained real time. That is, it can solve the power system equations fast enough to continuously produce output conditions that realistically represent conditions in the real network. Because the solution is real time, the simulator can be connected directly to power system control and protective relay equipment.

The main field of use for the RTDS® system is real-time simulation of power systems at dynamic level. Usual time step for this system is 50 μs, and if it is needed, in particular situations, 2 μs can also be used. With one individual rack roughly up to 80 nodes of a power grid can be simulated. So in total the lab can simulate grids with a complexity up to about 650 nodes.

For the Real-time simulation of distributed systems, the development of hardware and software to build a simulator based on a cluster of DSP is ongoing. The simulator will represent a unique facility for the testing of distributed energy system. The first application will be the RT simulation of large off-shore wind farms. This platform, though, is suitable for simulating all kinds of distributed generation systems.
For what concerns RT simulation of multi physic systems, a shared memory PC cluster is used

	for which a solver for complex systems was developed that leverages on the computational power of multiprocessor machines. This platform is being used for different projects and is therefore continuously being expanded. To support Hardware in the Loop and Power Hardware in the Loop testing, the RT setup is connected to the E.ON ERC testing facility. For this purpose, considering the great industrial interest on this topic, ACS designed and built a FlePS (Flexible Power Simulator), the power interface to realize Power hardware in the Loop, which will be used in the laboratories for testing of small apparatus (less than twenty kW). Considering that communication infrastructures are expected to have a growing impact on performance of future power systems, a communication infrastructure was developed to allow emulating network behaviours inside the lab. The initial focus was on Ethernet based communication, using a WANem to emulate network behaviours. WANem is a Wide Area Network Emulator that allows for the setup of a transparent application gateway, which can be used to emulate WAN characteristics like time delay, Packet loss, Packet corruption, Disconnections, Packet re-ordering, Jitter, etc. In order to enable easy data exchange with external systems, an OPC server is available in the lab. The OPC server provides protocol conversion and by this allows for real-time data exchange supporting an industry standard for automation. For the potential case of interconnecting with other partner's laboratories, the pan-European research and education network GÉANT is considered as facilitator. Further concrete plans for the expansion of the real-time laboratory include the integration of the following components: • Real SCADA (Supervisory Control and Data Acquisition) system; • PMUs (Phasor Measurement Units); • DC Microgrid; • Smart Meters; • Smart Gateways/Home Energy Management Systems. With the integration of the listed components, the lab will represent the electrical grid in connection with real hardware based on state-of-the-art and advanced commercial technology.
Adopted technical solution (potentially domain specific enablers?)	• Real-time simulation of electrical grid, possible for various scenarios within several FINSENY work packages; • Communication emulation.
Picture (Architecture, Interfaces, etc.)	The structure of the laboratory is represented in Figure 13.

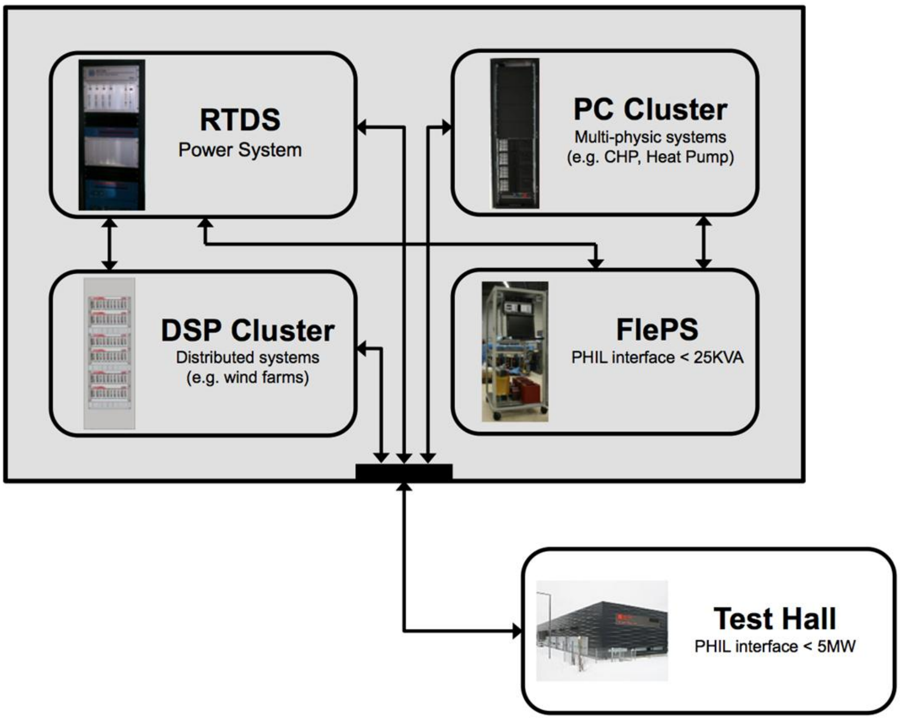


Figure 13: Testing capabilities

In Figure 14 the structure of the communication infrastructure of the laboratory is shown.

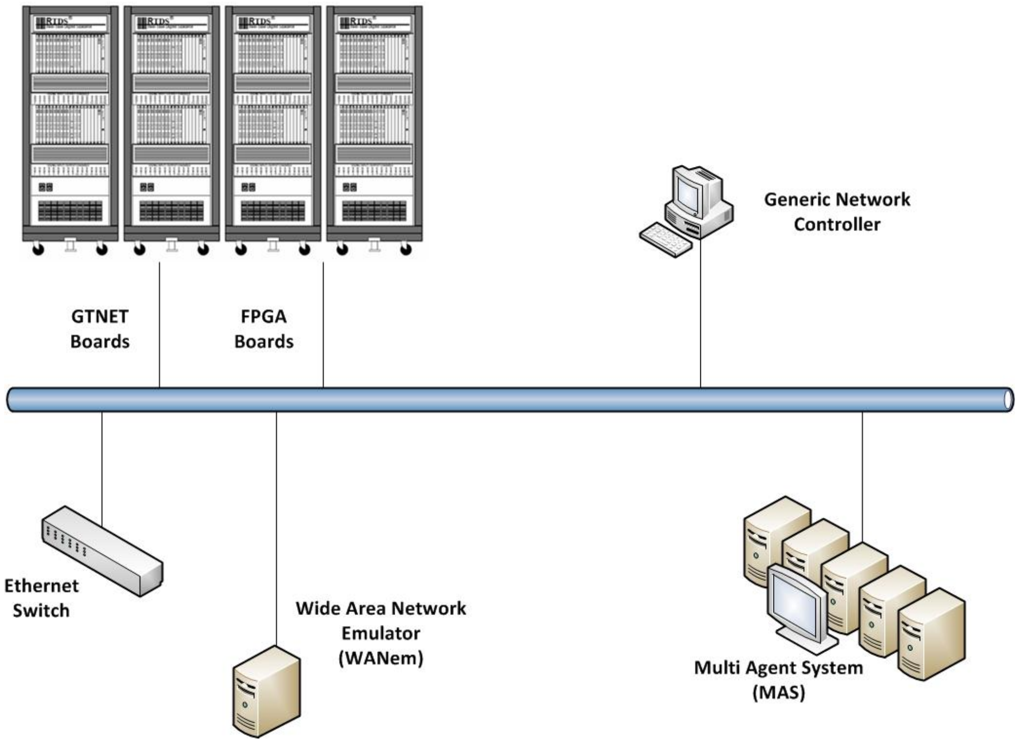


Figure 14: Communication Structure of the ACS Laboratory

Software availability	ACS has a complete set of software tools available to be used for different applications.

	Generic simulation, programming and control	**Multi-physic simulation**	**Power system simulation**	**Communication system simulation**
	• Matlab/Simulink • Visual Studio 2008, 2010	• SimulationX • VirtualTestBed2010	• Neplan	• OPNET
	• LabView 2010	• Open Modelica	• Power Factory	

5.2 Atlantic Labs

FINSENY Partner Commitment	**Atlantic Labs is a real world, integrated testbed between TSSG and Intune Networks and integrates electric vehicle charge point equipment provided by the ESB.** The partners endeavor to demonstrate the feasibility and scalability of using ICT to deliver rapid response load management functionality to the Grid and to provide a large scale, high capacity testbed for use in FI-PPP trial phase two.
Contacts & Main information	
Acronym & Name	Atlantic Labs
Leading Company or Authority	TSSG, Arclabs Research and Innovation Centre, Carriganore Campus, Waterford Institute of Technology, Ireland
Involved FINSENY partner	TSSG Intune Networks ESB Ecars
Type	Atlantic Labs is an integration of various existing research and industry test-bed facilities in Ireland and focuses on delivering a large, multi-functional, scalable test-bed to research and test future internet technologies. Testing can occur without the need to transmit data across the public internet. The testbed is required
Web site	www.atlanticlabs.ie
Location/Countries affected	Ireland
Running period	April 2013 – NA
Facilities description	
Technology keywords	fibre optic backhaul network, high capacity data centre, real electric vehicle charge points, intelligent control and monitoring of CPE, demand side management, dynamic network configuration
Standards	
Involved actors/user	TSSG and Intune Network. Both TSSG and Intune have connectivity the Irish research network (HEAnet) and, as a result, to the pan-European research network Géant. It is envisaged that this will facilitate ease-of-access for future research deployments on the testbed.
Brief scenarios description	The control of system frequency is a vital aspect of secure and stable power system operation. A continuous balance between active power generated and active power consumed by the load and losses is required to maintain frequency constant at nominal system frequency (50Hz). Any imbalance in active power will result in a frequency deviation. While precise instantaneous balancing of active power is not viable, frequency control ensures that the system frequency remains within acceptable frequency limits. Frequency control can be called upon for a variety of conditions ranging from a gradual change in load levels over time to a sudden loss of generation or step increase in demand.

<table>
<tr><td></td><td colspan="3">Of all the element of frequency control available to a Transmission Systems Operator (TSO), by far the most valuable is the availability of an autonomous 5s-15s response which constitutes Primary frequency control. Following that, a secondary (15s-900s) and tertiary (900s+) response period provides the operator with an opportunity to control additional resources in order to restore stability to the grid as well as to react to changes in the anticipated load pattern.

Atlantic Labs will make use of Web APIs and VPNs to dynamically configure high speed optical circuits to ensure that end-to-end control is available and that the testbed can scale to the required levels for future Smart Grid levels of operation (millions of sensors transmitting TBs of data).</td></tr>
<tr><td>Running experimentation (if exists)</td><td colspan="3">At this point (03/2012), the testbed is in initial development phase and will be running FINSENY phase one experiments until 04/2013.</td></tr>
<tr><td>Number of involved users during experimentation (if any)</td><td colspan="3">TSSG, Intune Network and ESB</td></tr>
<tr><td>Is any Experimentation Lab still running?</td><td>NO</td><td>Until?</td><td></td></tr>
<tr><td colspan="4">Involved Technologies</td></tr>
<tr><td>Brief Technical description</td><td colspan="3">Atlantic labs is an integration of all of the following test-bed Infrastructures:
• Intune and Irish Government Exemplar Network advanced distributed telecoms testbed;
• TSSG – Datacentre and eCar charging facilities;
• ESB eCars and supporting infrastructure.

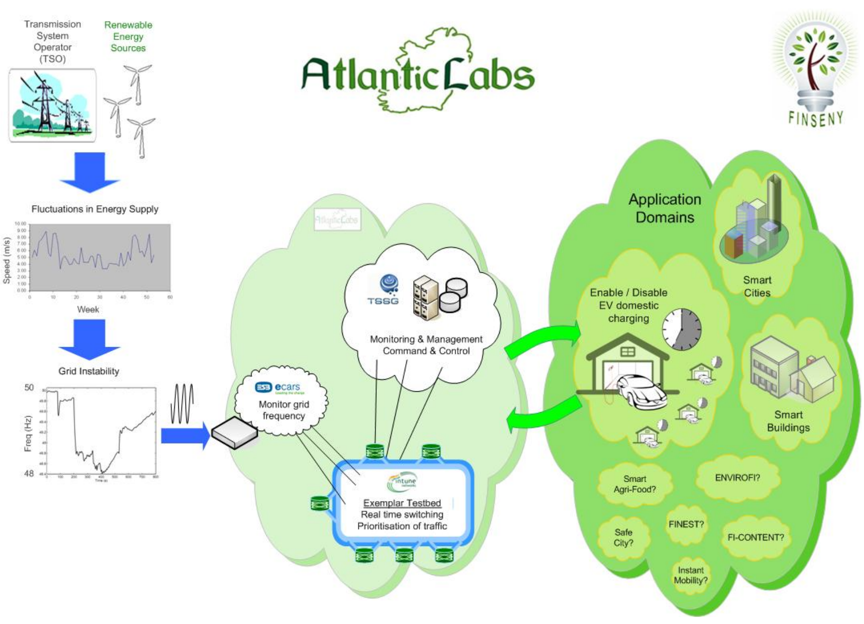
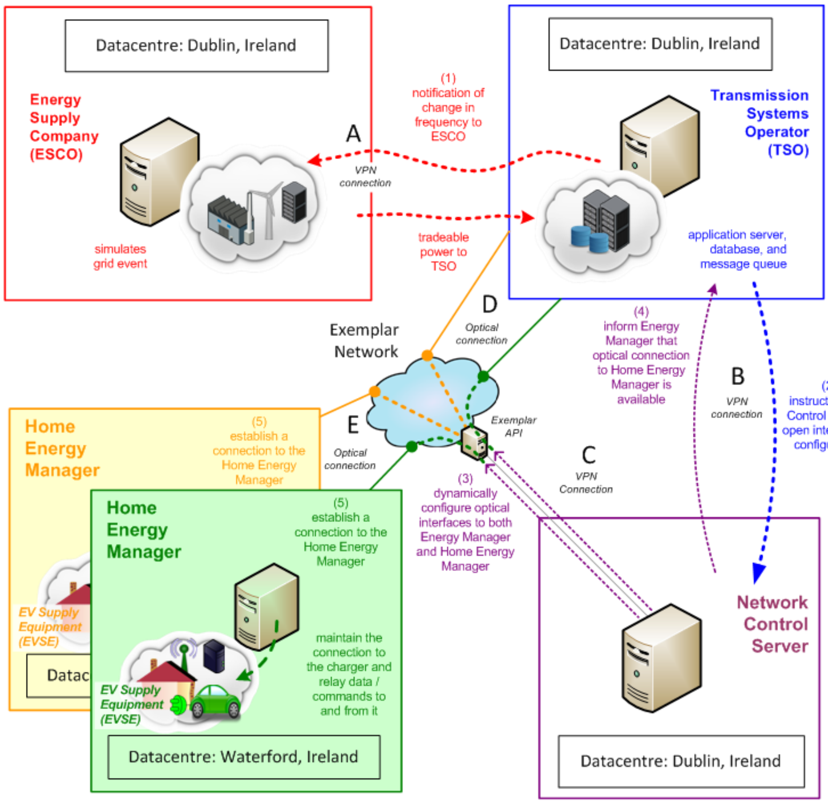


Figure 15: Storyboard for ICT enabled DSM as part of Atlantic Labs

Figure 15 shows a story board for an integrated testbed between TSSG and Intune Networks ensuring that testing can occur without the need to transmit data across the public internet. The testbed is required to demonstrate the feasibility and scalability of using ICT to deliver rapid response load management functionality to the Grid.

Essentially, it uses ICT to provide a smarter, more efficient, autonomous demand side management system. In WP5, the focus is to manage the load being drawn from Electric Vehicles but, as illustrated by the “Application Domain” in the above figure, there are other relevant applications of this concept to control various other sources of load on the grid.</td></tr>
</table>

	.
Adopted technical solution (potentially domain specific enablers?)	Unknown at this point
Picture (Architecture, Interfaces, etc.)	The testbed will comprise both simulated and real-world elements. The fluctuation in grid frequency will be simulated. Once this is initiated there will be instability in the grid and this Grid Event will be recognised and acted upon by the TSO. Following that, real-world management and control of the load on the network (electric vehicles in this case) will allow the TSO to stabilise the grid and demonstrate the feasibility of the use case and testbed. Figure 16 and Figure 17 provide more detail about the envisaged interactions of the components of the testbed.  **Figure 16: Connectivity diagram for Atlantic Labs testbed** As illustrated in Figure 16 above, Atlantic Labs will make use of software VPNs to dynamically configure high speed optical circuits to ensure that end-to-end control is available and that the testbed can scale to the required levels for future Smart Grid levels of operation (millions of sensors transmitting TBs of data). The interfaces are: A. **Software VPN from *ESCO* to the *TSO*** – to allow a grid event simulation to occur; B. **Software VPN from *TSO* to *Network Control Server*** – to allow the Energy Manager to instruct the Network Control Server to initialise connectivity to the Exemplar network; C. **Software VPN from *Network Control Server* to *Exemplar API*** – enabling real-time dynamic configuration of interfaces to the Energy Manager and Home Energy Manager; D. **Optical connection to the *TSO*** – allowing for connections to be made to multiple Energy Managers if required; E. **Optical connection to the *Home Energy Manager* (which manages the *EVSE*)** – allowing for connections to be made to multiple Home Energy Managers. Once **D** and **E** are established, an end-to-end optical connection is available, providing high

<table>
<tr><td></td><td>
speed control and allowing a DSO to control multiple Home Energy Managers and, as a result, control multiple EVSE's. As such, the testbed becomes dynamic and robust and, significantly, is extremely scalable.

The figure below provides a message sequence diagram for controlling the EV charge point in the event of a change in grid frequency.

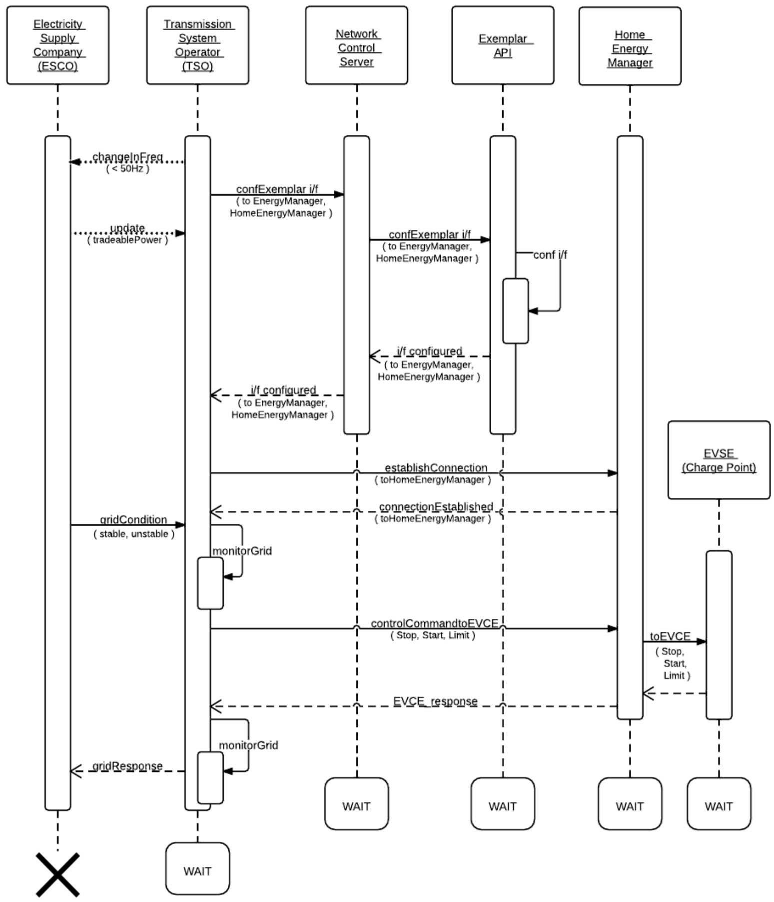
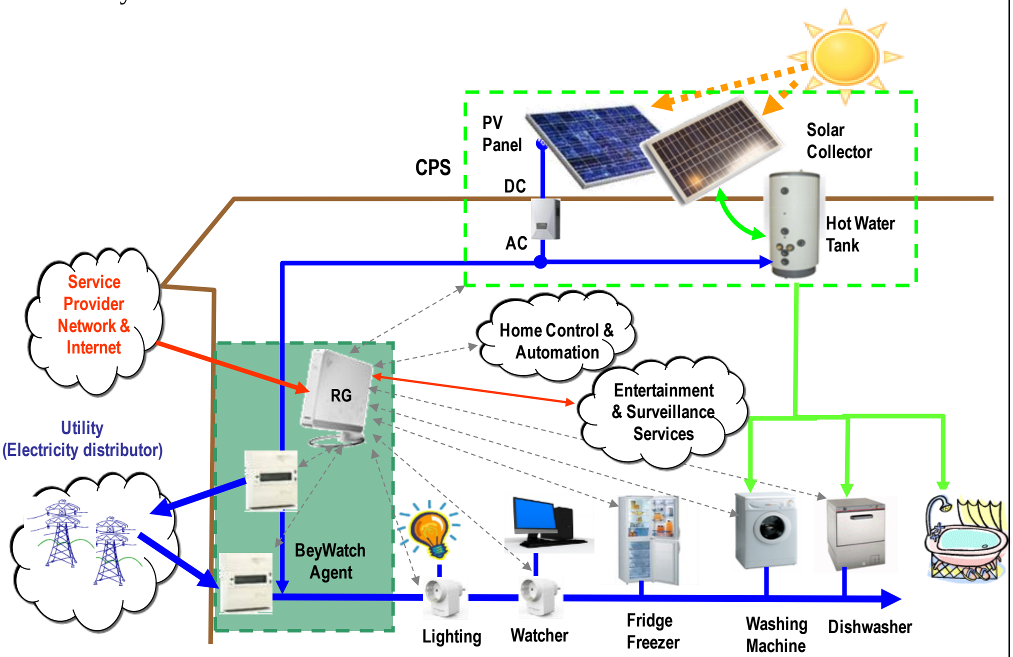


Figure 17: ICT_DSM sequence diagram for Atlantic Labs
</td></tr>
<tr><td>Software availability</td><td></td></tr>
</table>

5.3 BeyWatch

<table>
<tr><td>FINSENY Partner Commitment</td><td>Telefonica I+D (TID) and Synelixis (SYN)

TID led BeyWatch project at administrative level and SYN has been the Technical Coordinator of the project. Together TID and SYN, supported by EDF, have led the Integration & Validation work package at technical level. TID and SYN will contribute to FINSENY WP8 activities and TID will lead the experimentation & validation task.</td></tr>
<tr><td colspan="2">Contacts & Main information</td></tr>
<tr><td>Acronym & Name</td><td>BeyWatch – Building EnergY Watcher</td></tr>
<tr><td>Leading Company or Authority</td><td>Telefonica I+D (TID) – Spain</td></tr>
</table>

Involved FINSENY partner	TID, Synelixis, EDF
Type	FP7 project, ICT-2007.6.3 ICT for environmental management and energy efficiency
Web site	http://www.beywatch.eu
Location/Countries affected	Spain, France, Greece, Italy, UK, Slovenia
Running period	2008-2011
Facilities description	
Technology keywords	Energy efficiency, smart metering, energy watcher, control and monitoring, scheduled control for expense optimization, data model, graphical user interface, home user, utility user, demand side management, customer insights, low cost energy aware product integration, combined photovoltaic & solar thermal system, simulated home appliances.
Standards	WiFi, Zigbee
Involved actors/user	Residential dweller, utility operator
Brief scenarios description	A residential dweller can: • **Monitor** how much their appliances are consuming: instant power demand and accumulated energy consumption; • Check historical consumption of each appliance; • **Control** the intelligent home appliances (dishwasher, washing machine, fridge/freezer, and legacy appliances plugged in energy-aware smart plugs), including also the control of a combined photovoltaic & solar thermal system (which produces electricity and hot water for domestic use); • Set preferences for scheduling appliances operation and optimizing the total cost of electricity for the user.  **Figure 18: BeyWatch Home reference configuration** and an utility operator can: • Smooth energy control; • Load balancing control at neighbour level; • Incentive/contra-incentive electricity consumption at different times.

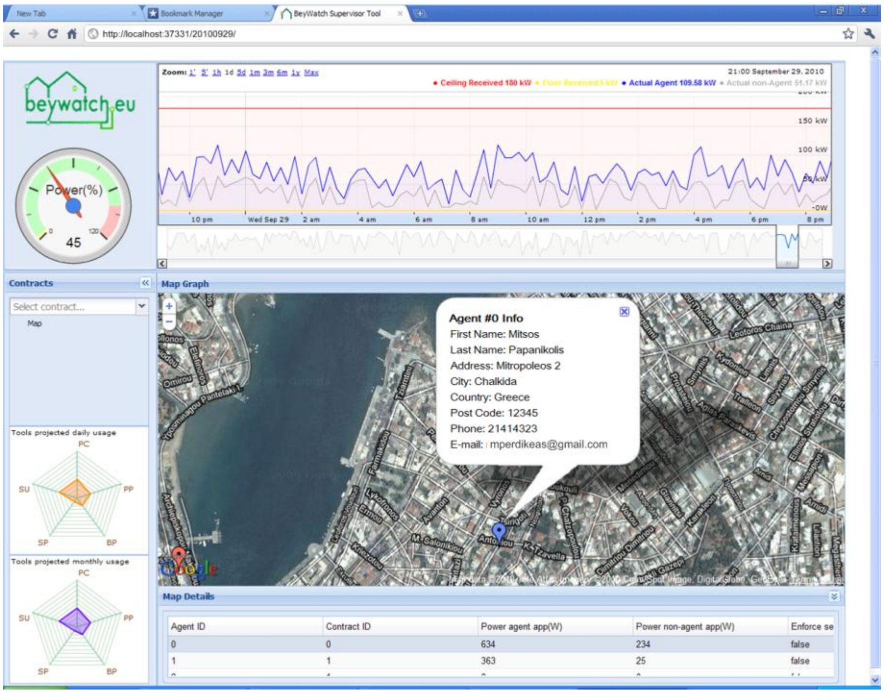
<table>
<tr><td></td><td colspan="3">

Figure 19: BeyWatch Utility Operator User Interface</td></tr>
<tr><td>Running experimentation (if exists)</td><td colspan="3">BeyWatch experimentation has been run during 2010-2011 in testbeds and laboratories in Greece, Italy, France and Spain. An extensive testbed run in EDF R&D site in Les Renardières.

Figure 20: BeyWatch “Les Renardières” EDF R&D site – “SmartHome” Lab</td></tr>
<tr><td>Number of involved users during experimentation (if any)</td><td colspan="3">Experimentation in laboratory and testbed facilities. Real users were involved only in the validation/ user acceptance phase of the main functionalities related to energy efficiency services conceptualized in BeyWatch.
Additionally simulations took place based on off the self energy consumption simulators.</td></tr>
<tr><td>Is any Experimentation Lab still running?</td><td>No</td><td>Until?</td><td>End of 2011</td></tr>
<tr><td colspan="4">Involved Technologies</td></tr>
<tr><td>Brief Technical description</td><td colspan="3">The project has run an integration and performance evaluation cycle at the end of the project, for a time period of 3 months. The objective was to run the integrated BeyWatch system and measure the different gains in energy efficiency per appliance and for the whole integrated system (applying demand side management measures driven from the BeyWatch Supervisor, and delegating to the BeyWatch Agent the responsibility for consumption optimization at home level, collecting user requirements together to utility incentives/counter-incentives)</td></tr>
<tr><td>Adopted technical solution (potentially domain specific enablers?)</td><td colspan="3">At home level:
Management from an Energy Gateway (BeyWatch Agent) of the different home appliances. Appliances use ZigBee and WiFi to communicate with the Gateway
The BeyWatch Agent offers:
A module for communicating with the Supervisor (utility/Energy Service Provider tool)</td></tr>
</table>

- A portal for the user, where he/she specifies the preferences related to white goods operation
- An energy optimization and scheduler module, which compiles the different requirements to come up with the schedule for operation of the appliances that minimizes the cost for the user.
- A monitoring and control module

At grid level:

- The BeyWatch Supervisor:
  - Has a web interface for displaying aggregated demand, actual and forecasted and to allow the operator to send incentives and counter-incentives to the supervised Agents to smooth the aggregated demand curve.
  - The BeyWatch Supervisor and Agent communicates through web services

The results achieved with such a system are:

- For the white appliances (dishwasher, fridge/freezer and washing machine), the gains in standalone mode have reached up to 30% less consumption. The use of hot water coming from the CPS in the appliances has shown to be a measure that can save up to 50% of the normal power consumption;
- The ICT enabled fridge/freezer can be used for “load shaping” by storing cold;
- The ICT enabled CPS (Combined Photovoltaic System) is a very good tool that in combination with the agent can help reduce the energy bills of the customers up to a 59%;
- The Agent software running at the home level has shown that is able to control a high number of devices/islands at home to reduce the energy consumption, taking as inputs the user preferences and complex tariff schemes. It can be used very effectively for Demand Side Management and peak-shaving or load shifting.

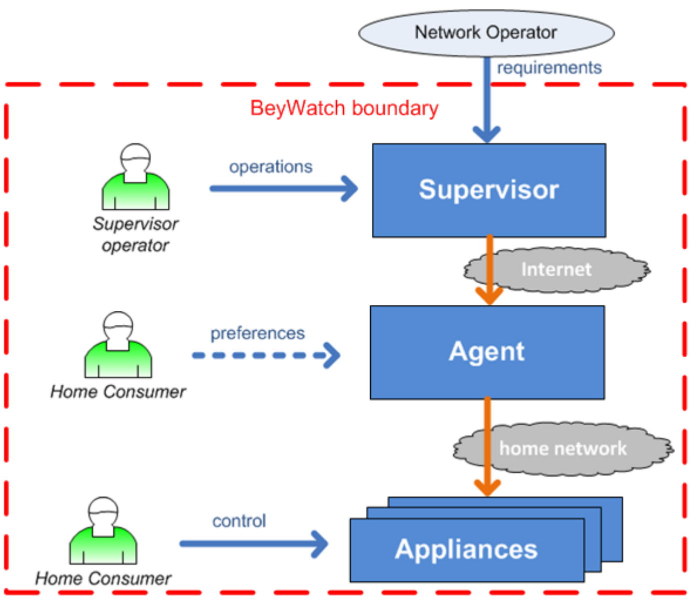


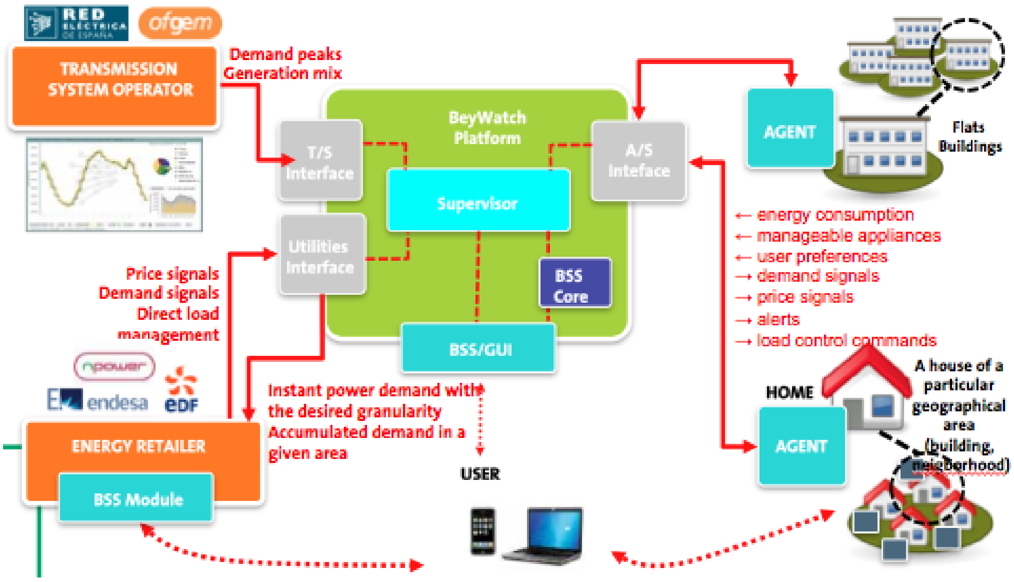
Figure 21: BeyWatch modules interrelation

Possible Improvements:

- M2M communications at Home are still suffering from the lack of standardization and from the high level of fragmentation in the Home Area Network landscape;
- ZigBee profiles need further development in order to suite Home Automation and energy efficiency needs in parallel;
- Smart metering combined with ZigBee sometimes is difficult to implement because of the distance between the meter and the in-home equipment. So networking reliability should be improved.

FINSENY perspectives:

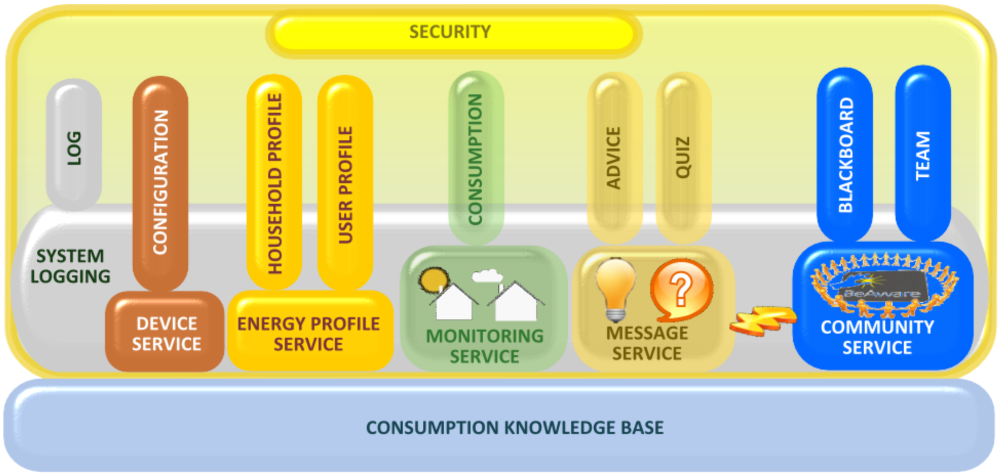
- With FINSENY, BeyWatch can be upgraded to support not only residential buildings but also commercial buildings, offices, data centers, etc.;
- Could be extended to support the microGrid scenarios of FINSENY;
- Higher integration with the energy market place will pave the way to provide to the “agent” system with more information that can induce better energy efficiency. Also the

	users will be able to get information services that will for sure raise user awareness. Through the FIWARE enablers, BeyWatch provisioning could be easier and rapidly instantiated for very distinct markets.
Picture (Architecture, Interfaces, etc.)	 **Figure 22: BeyWatch high level architecture**
Software availability	The complete Agent and Supervisor software suites are available from the FINSENY partners Synelixis and TID.

5.4 BeAware

FINSENY Partner Commitment	**Engineering Ingegneria Informatica S.p.A. [ENG]** **ENG lead the development of Service Layer in the BeAware project and will support FINSENY during the WP8 activities about prototyping and experimentation.**
Contacts & Main information	
Acronym & Name	BeAware - Boosting Energy Awareness with Adaptive Real-time Environments
Leading Company or Authority	Aalto University of Helsinki, Finland (FI)
Involved FINSENY partner	ENG, ENEL.SI
Type	FP7 project, ICT-2007.6.3 ICT for environmental management and energy efficiency
Web site	http://www.energyawareness.eu
Location/Countries affected	Sweden, Italy, Finland
Running period	2008-2011
Facilities description	
Technology keywords	Smart Metering, Energy Saving, Energy Awareness, Smart Home
Standards	--
Involved actors/user	Residential dweller

Brief scenarios description	Through a smart phone interface, residential dweller can: • Monitor how much their appliances are consuming; • Check historical consumption of each appliance; • Check if some appliance is consuming more with respect to the last seven days; • Play a pervasive game by reading tips and answering quiz tailored to their energy habits in a context-aware way with the objective of improving their awareness about energy consumption in the household context; • Exchange messages and share good practices within BeAware social network where all BeAware users join; • Check a pie chart about energy consumption household breakdown.		
Running experimentation (if exists)	BeAware experimentation has been assessed via a two phase six months trial during 2009-2011 by involving several household in Sweden, Italy and Finland		
Number of involved users during experimentation (if any)	Around 3 people per household => 10 household per phase => 50-60 people in total		
Is any Experimentation Lab still running?	YES	*Until?*	End of 2011
Involved Technologies			
Brief Technical description	Each household has installed a set of sensors connected to each appliance. A basestation gateway (installed in the household too) gathers all consumption data coming from sensors via wireless. Gathered data are sent to a centralized Sensing Layer that makes a pre-processing analysis, stores them and provides data to the Web Service Platform. Home dwellers access their smart phone to check consumption data by using Web Service Platform.		
Adopted technical solution (potentially domain specific enablers?)	• Smart Sensor – developed by Aalto University of Helsinki (http://www.aalto.fi/); • Sensing Layer – platform for consumption data management and elaboration developed by BASEN (https://www.basen.net/); • Basestation – gateway deployed in each household, running on Ubuntu Linux, developed by BASEN (https://www.basen.net/); • Energy Management Web Service Platform (Service Layer in architecture picture) - lead by Engineering Ingegneria Informatica S.p.A (http://www.eng.it) – JAX-WS service platform with JSON binding for building final user application; • EnergyLife (Application Layer in architecture picture) – web interface for iPhone tailored to home dweller – developed by Aalto University of Helsinki (http://www.aalto.fi/).		

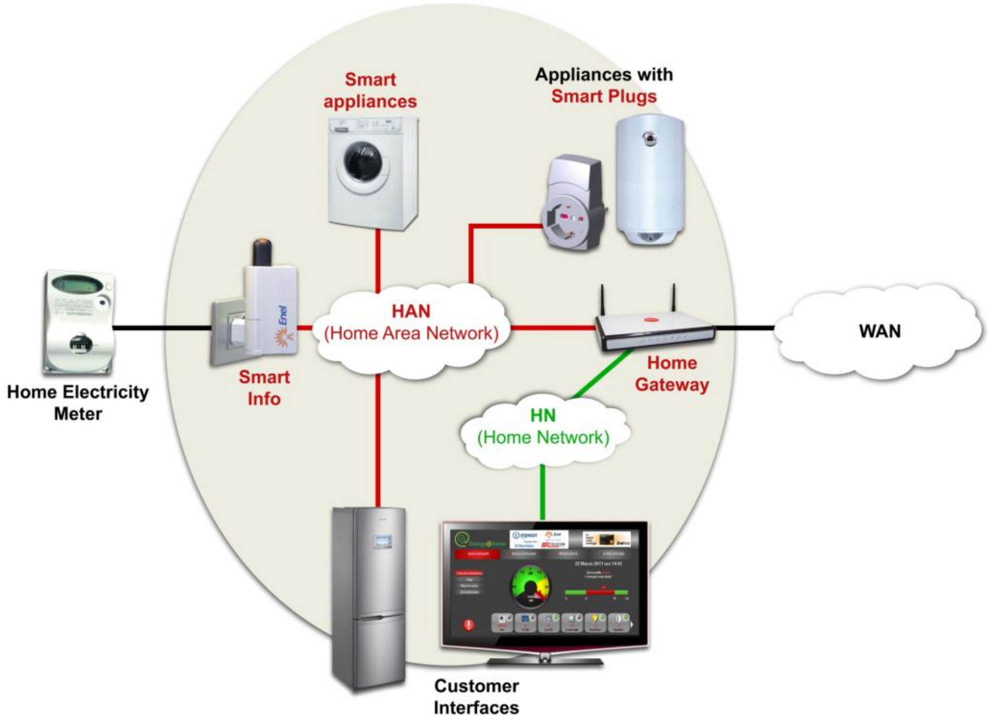
Picture (Architecture, Interfaces, etc.)	 **Figure 23: BeAware Big Picture**  **Figure 24: BeAware Service Layer**
Software availability	http://sourceforge.net/projects/beaware/

5.5 Energy@home

FINSENY Partner Commitment	**Telecom Italia (TI)** **TI leads the development of Energy@home and will support FINSENY during the WP8 activities about prototyping and experimentation.**
Contacts & Main information	
Acronym & Name	E@H - Energy@home
Leading Company or Authority	Electrolux, ENEL, Indesit Company and Telecom Italia

Involved FINSENY partner	Telecom Italia		
Type	Research project self-funded by the partners. A solution for Energy management System for monitoring and controlling smart appliances		
Web site	www.energy-home.it (*English version available*)		
Location/Countries affected	Project started in Italy extensible in other European countries		
Running period	Since 2009 and actually in progress		
Facilities description			
Technology keywords	Smart Grids, Smart Metering, Energy Saving, Energy Awareness, Smart Homes, Connected Homes, Home Automation Profile, Smart Energy Profile, Internet of things, Smart Appliances		
Standards	*Internet of Things – under standardization* *Cloud computing – under development* *IEEE 802.15.4 wireless communication – stable* *IEEE 802.11 WiFi communication – stable* *xDSL broadband communication – stable* *power line communication – stable*		
Involved actors/user	Residential dweller, Energy Retailer, Energy Distributors, Telcos, Appliances vendor		
Brief scenarios description	• Measure global energy consumption to improve user's energy awareness; • Optimize the energy bill implementing automatic scheduling; • Increase user's comfort by avoiding power overload; • Measure per-device energy consumption. In particular for the washing machine the energy consumption and usage patterns of the user will be measured and compared against legacy washing machine and connected washing machine; • Measure the usage patterns of the graphical tool; • To monitor how and how much the user moves its consumption in off-peak timeslots; • Optimize Home Energy Globally; • Optimize home energy locally; • Activation of smart appliances leveraging on the Photo Voltaic generation.		
Running experimentation (if exists)	Energy@Home experimentation has been run during 2010-2011 in test beds and laboratories in Italy. Field trials started on 2012.		
Number of involved users during experimentation (if any)	10 houses already connected		
Is any Experimentation Lab still running?	Yes	*Until?*	Estimated end 2014
Involved Technologies			
Brief Technical description	The system is installed into a Living Lab, the Innovation Lab of Telecom Italia in Torino. On average it is visited by 2000 people/year with about one visit/day. At the time of writing, the trial is active in 10 private houses. By 2012 it is planned an extension to 100 houses. Each private house is equipped as follows: • Smart meter; • SmartInfo device of Enel that bridges the Smart Meter communication with the Home Area Network; • Energy Box which is also the HAN controller. It is a device of Telecom Italia with OSGi (Open Service Gateway initiative) framework and HAN wireless communication capability; • 5 smart plugs with a local meter, a switch, and a radio communication; • 1 smart appliance with embedded radio communication. The system is open and fully controlled by FINSENY partners. It can be easily extended to include communication with other devices, to implement new functionalities and to interact with other FINSENY trials domains. The system in the demo room can be easily extended.		

	The trial in private houses can be extended in time under some constraints to be verified.
Adopted technical solution (potentially domain specific enablers?)	The adopted *Energy@home* technical solution are reported in the following: ***A. Smart Appliances*** Smart Appliances represent an evolution of the current standard white goods: the new smart devices feature connectivity towards both the home environment and the smart grid, and embed an enhanced local intelligence in order to manage innovative services. Hereunder, some of their possible new functionalities are reported: • Display to the customer information on their energy consumptions (e.g. used energy, instant power, etc.); • Dispatch in the HAN information on their energy consumptions (e.g. used energy, instant power, etc.); • Autonomously adapt their behavior according to information on energy consumptions coming from the house. (e.g. reduce their load when global house consumption goes beyond a threshold, etc.); • Cooperatively operate with other entities in order to optimize the energy usage through load shifting and load shedding. In any case, the load control operations, either performed autonomously or under an external supervision, shall be performed under the complete control of the appliance, which assures the correct execution of its working procedure, its results and performance. For example, a smart washing machine, when requested to modify its consumption behavior, shall assure the result of the washing cycle. ***B. Smart Plugs*** The Smart Plugs can actively participate to in home monitoring and control activities. To this purpose, smart plugs are able to collect metering data and implement on/off control on simple plugged energy loads, other than Smart Appliances (e.g. lighting, A/V devices, etc.). ***C. Customer Interfaces*** The customer interfaces could: • display information on energy usage like instant power, historical data, contractual information and similar, from the whole house (coming from the Smart Info) and from every single smart appliance. The level of details and graphical layout of their user interface is freely defined by every device; • transmit control message to Smart Appliances to request a modification of their behaviour; • configure Smart Appliances to modify their power consumption profile (e.g. a personal computer used to configure a thermostat to activate the controlled load only in certain time slots). The Customer Interface, from this perspective, is connected in both the HN and HAN. It is foreseen the possibility to have Customer Interfaces accessing the house from the WAN through a specific interface, but the definition of this interface is out of the scope of the *Energy@home* project as previously stated. Typical Customer Interfaces are personal computers, smart phones, PDAs, *ad-hoc* displays, entertainment systems, in-house monitor and similar. The software application, which implements the user interface, could be local in the device or remotely hosted in another device (e.g. the Home Gateway) and accessed through web-services. ***D. Home Gateway*** The Home Gateway represents the link between the HAN, the HN and the WAN (e.g. internet). It is able to interface Smart Appliances and other user's devices (e.g. PC) through the communication protocol(s) used in the HAN (e.g. ZigBee) and in the HN (e.g. IP/HTTP) and to provide a broadband connection to internet (usually via a standard ADSL connection). Moreover, the gateway is able to collect energy data from the Smart Info and additional information from Smart Appliances, publish them in the HAN and in the HN and use all collected data to control Smart Appliances and optimize their behavior. Finally, the gateway can offer a web user interface and provide an execution environment (e.g. Java OSGi framework) to host third-party application (e.g. a SW component implementing the algorithm to calculate the energy price at a given time, provided by the energy retailer).

	E. Smart Info The Smart Info is the element, provided by the DSO (*Distribution System Operator*), which dispatches energy related information into the HAN. Published data are a sub-set of those already available inside the home electricity meter, hence the Smart Info acts like a proxy of the meter. Additional data could be possibly generated by the Smart Info itself. Noticeably, near real-time instant power (sampled at about 1 Hz frequency or higher) should be acquired by another metering device, likely embedded into the Smart Info. Additional elements (SI') can also be provided by third parties and used to dispatch data generated by other meters into the HAN. Outstanding components outside the Home Domain are: • HAP: Home Automation Platform: it manages, together with the Home Gateway, the HAN devices and provides service oriented interfaces for the development of third-party applications; • For an effective use of the energy, the smart sustainable appliances must have an active role in the energy management automatic systems: o Being able to completely control the processes as they are fully responsible for the final result; o Offering, thanks to an active dialog with the customer and the energy sources, a valuable flexibility in terms of time and energy profile (best tariff).
Picture (Architecture, Interfaces, etc.)	 **Figure 25: Energy@home Architecture**
Software availability	Hardware and software for the products are ready to be used in field trials

5.6 GrenobleINP Experimentation Lab

FINSENY Partner Commitment	**Grenoble Institute of Technology** **The role of Grenoble Institute of Technology in the FI-PPP project is to provide its completing electrical engineering, distributed control and optimization competences specifically focused on the electrical grid and the building to contribute to develop the concept of smart energy management on the basis of ICT tools and particularly the future Internet. It could provide about the advanced functionalities (as congestion management, fault detection, isolation and restoration, voltage regulation also named Advanced Distribution Automation Function), and about the reconfigurable electrical network necessary for a smarter energy management in the future generation of the electrical grid. It will also provide the ability of smart buildings to provide services to the grid. Then a transverse point of view can be achieved.**		
Contacts & Main information			
Acronym & Name	PREDIS		
Leading Company or Authority	Grenoble INP		
Involved FINSENY partner	Grenoble INP		
Type	Laboratory Equipment		
Web site	http://www.grenoble-inp.fr/le-groupe/predis-7154.kjsp		
Location/Countries affected	Saint Martin d'Hères, France		
Running period	Equipment available since 2006		
Facilities description			
Technology keywords	Reduced scale distribution network, ICT infrastructure, Advanced Distribution Automation functions, emulation of real industrial automation systems and of its external ICT system		
Standards	French "Substation Numerical Control Command" with advanced features IEC 61850, SCADA		
Involved actors/user			
Brief scenarios description	A dispatching centre which consists of an open SCADA system upgraded with observability and dispatches functions where a self healing function was implemented. A connection to the main grid enables the supply security and the interaction with the market (selling/buying power depending on the price of the energy) A real industrial SCADA application was developed in order to assess the interaction between the distributed control and the centralized control that exists. To be able to evaluate the ICT performances requested by the self healing agent, different layers of communication (RTU to SCADA/intra substation/PAC to SCADA/RTU to agent to SCADA) and associated monitoring have been developed.		
Running experimentation (if exists)	The Experimentation lab has already been used during the INTEGRAL STREP European project		
Number of involved users during experimentation (if any)	Experimentation in laboratory and testbed facilities.		
Is any Experimentation Lab still running?	no		
Involved Technologies			
Brief Technical description	The site consists of: • Generations facilities: microturbine and co-generation, PV cells, hydro turbine and fuel cells system; • Controllable loads: electrical and thermal loads, low consumption loads, smart appliances;		

- Physical electrical and heat network: protection, switches, metering, cables;
- Control room and dispatching center;
- Interconnection facilities to the main distribution grid.

This demonstration site has a total of about 30 kW electric loads whose objectives are multiple:

- Multi technology proof – demonstrating the ability of multi DG technology operation with various control capabilities;
- Electricity and heating local supply (ENSE3);
- Research facility where different innovations issued from research effort of the associated research laboratories can be implemented and tested. As an example, the dispatching centre consists of an open SCADA system upgraded with observability and dispatches functions where a self healing function was implemented during the INTEGRAL STREP European project;
- Initial training center dedicated mainly for the graduate school; of electrical engineering but also for the other students of the University of Grenoble. The center offers custom-built experiments but research projects for educational purposes;
- Continuous training for professional development in advanced electrical technology;
- Connected to the main grid for the supply security and the interaction with the market (selling/buying power depending on the price of the energy).

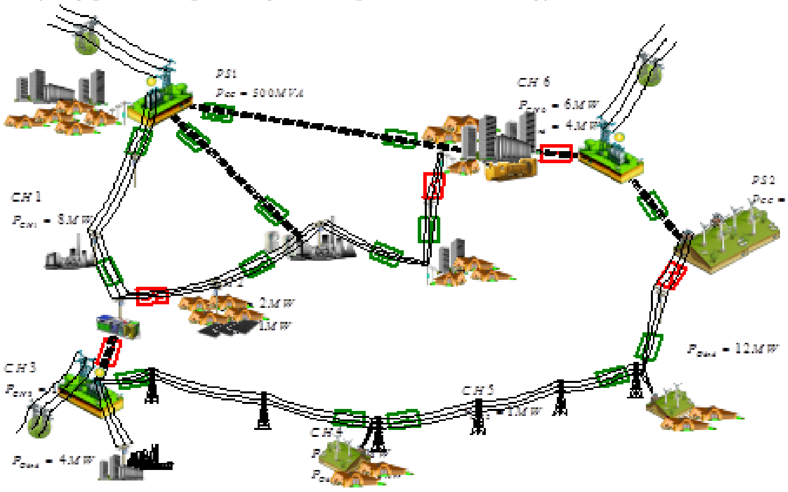


Figure 26: Network structure extracted from a real French Distribution Network

This network is a reduced scale (20kV → 400V ; 30 MVA → 30 kW) from a real French Distribution Network with 3 substations (63/20kV), seven emulations of Distributed Generators (synchronous generators with active and reactive control) and nine controllable loads (induction machines and dimmers). The network components such as lines and cables are represented with their equivalent values. Substations are emulated with internal electrotechnical components (transformers, breakers) and substation automation, see Figure 26. In one of those substations, the automation system was fully derived from French “Substation Numerical Control Command” with advanced features compliant with IEC-61850. Indeed few computer based automation systems are interconnected with their internal ICT network. These computers emulate real industrial automation systems linked with internal communicative protective relays.

A real industrial SCADA (Figure 27) application was developed within the INTEGRAL project in order to assess the interaction between the distributed control and the centralized control that exists.

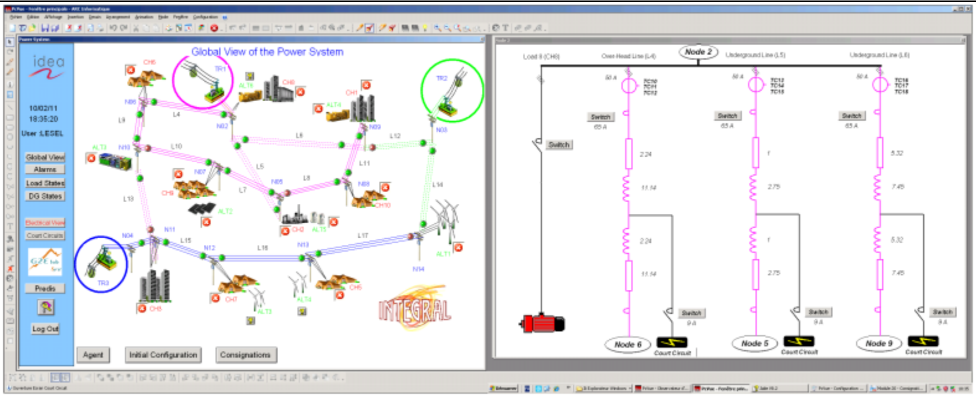


Figure 27: SCADA for DNO operator

To be able to evaluate the ICT performances requested by the self healing agent, different layers of communication (RTU to SCADA/intra substation/PAC to SCADA/RTU to agent to SCADA) and associated monitoring have been developed

Adopted technical solution (potentially domain specific enablers?)	• Real-time simulation of electrical grid, possible for various scenarios within several FINSENY work packages; • Communication emulation.
Picture (Architecture, Interfaces, etc.)	See above, in the Brief Technical description.
Software availability	(see table below)

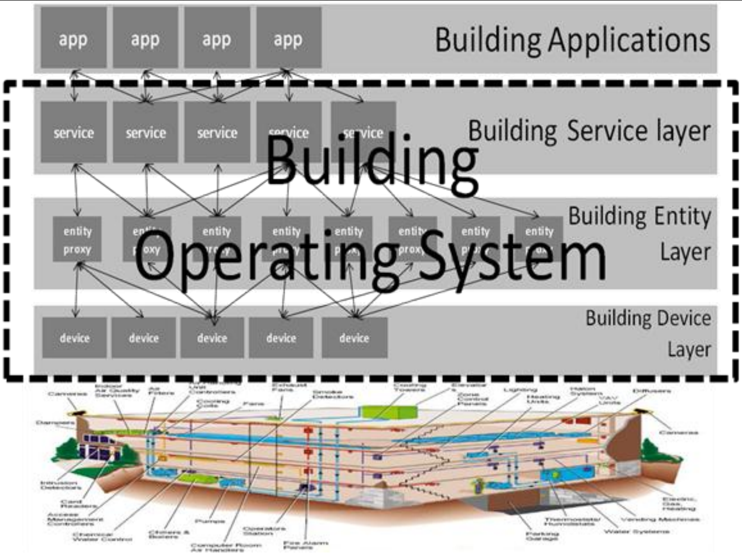
OS	Application	Software
Windows XP	PLC	Utility
	SCADA	PcVue
	Visual Display	Pyhon
	Distributed Database	OFS
	Fault Recorders	LabVIEW
	Self Healing Agent Modelling	Matlab
	Communication between Agent/Outside	Matlab OPC toolbox
FreeBSD	Emulated communication network	
	Emulated telecommunication nodes	Open source software

5.7 Building Abstraction Layer (integrated in BOS - Building Operating System)

FINSENY Partner Commitment	**Orange Labs proposes a shared building platform that can play for buildings a role similar to that of an operating system for a computer: provide a unified interface for applications that hides away the low-level peculiarities of the hardware involved. This is an application, at the building scale, of the FI-PPP approach of sharing and horizontalizing the infrastructure as much as possible**
Contacts & Main information	
Acronym & Name	Building Operating System
Leading Company or Authority	Orange Labs
Involved FINSENY partner	OLF

Type	Research project		
Web site			
Location/Countries affected	Not applicable (as it is a software platform)		
Running period	Not limited		
Facilities description			
Technology keywords	*Smart Environment, Building Abstraction Layer, Internet of Things*		
Standards			
Involved actors/user			
Brief scenarios description	Self-identification and self-configuration of home entities in the Home Abstraction Layer, electrical appliances and legacy non-connected equipment. An interface to these entities is made available to applications for both monitoring and control through a generic and open “Home Abstraction Layer” service.		
Running experimentation (if exists)	Configuration of hardware appliances in home instance of building operating system		
Number of involved users during experimentation (if any)			
Is any Experimentation Lab still running?	Yes	*Until?*	*<Date>*
Involved Technologies OSGI			

Brief Technical description	**Overall Architecture** Sensors and actuators are supposed to be shared between all building applications and made available as a pool when they are individually identifiable and addressable Black box legacy systems that do not give access to their individual sensors and actuators will be dealt with through their own interfaces. Building energy management and building automation applications in general are not interested in sensors and actuators themselves, but in what is being sensed by the sensors, or acted upon by actuators. The relevant level of abstraction for information pooling should thus be at the level of the physical entities that are being sensed by sensors and acted upon by actuators, which can be pieces of equipment, appliances, people, rooms of a building, or more generally any relevant self-contained subsystems of the building. These entities are generic, intrinsic to the building environment and not tied to any specific building automation application. A set of models and corresponding software components for these entities make up a “Building abstraction layer”, in a way similar to a hardware abstraction layer for a computer platform. An additional service layer, corresponding to software enablers that span several entities or entity categories, may be provided to applications on top of the building abstraction layer to make up the building operating system. Absent such services, the interfaces that are exposed to applications from this building operating system may correspond directly to the states and associated attributes of relevant physical entities of the building. Further, absent the models representing such entities, interfaces exposed to applications from this building operating system may correspond directly to sensor readings and actuator control parameters. Taking a room of a building as an example such entity, the state of a room could be whether it is occupied, the type of activity going on, and the corresponding attributes could be its temperature, the number of persons present, etc. For control purposes, an application can change the state of an entity to another state, if admissible, or change associated attributes. In the examples below, the state of a room could be changed to dark by sending coordinated commands to individual actuators, such as those controlling shades and light fixtures. The use of this level of information abstraction as a pivotal intermediary layer is in line with the Internet of Things and Context Management enablers provided for the Future Internet platform by the FI-WARE project. **OSGi-based implementation** The OSGi specifications (OSGi, 2009) define a standardized, component-oriented computing environment for networked services that is the foundation of enhanced service oriented architectures. Using the OSGi framework for the building operating system adds the capability to manage the lifecycle of software components. They can be installed, updated, or removed on the fly without ever having to disrupt the operation of the HAB. Core features of OSGi are based on an original Java class loader architecture that allows code sharing and isolation between modules called bundles. A bundle contains Java classes that implement zero or more services. Bundles are deployed in an OSGi service platform to provide application functions to other bundles or users. Our purpose in choosing OSGi is to enable BOS application reconfiguration at runtime. So, each SIMC functional module is modeled as a bundle, and its application functions are packaged in the bundle as services.
Adopted technical solution (potentially domain specific enablers?)	

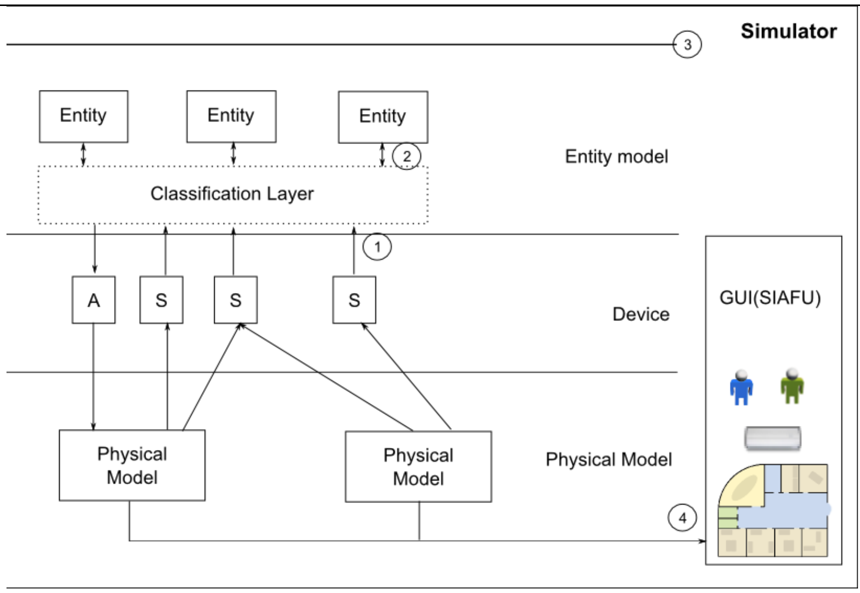
Picture (Architecture, Interfaces, etc.)	 **Figure 28: Building abstraction layer (integrated in BOS Building Operating System)**
Software availability	Software object code available

5.8 MILESENS (MultI LEvel Smart ENvironment Simulator)

FINSENY Partner Commitment	**Orange Labs**
Contacts & Main information	
Acronym & Name	MILESENS (MultILEvel Smart ENvironment Simulator)
Leading Company or Authority	Orange Labs
Involved FINSENY partner	OLF
Type	Research project
Web site	
Location/Countries affected	
Running period	Software
Facilities description	
Technology keywords	Middleware, discrete-state model, supervisory controller, entity monitoring and configuration, multi layer, real-time and accelerated software simulation
Standards	ETSI M2M release 2 (for entity/thing)
Involved actors/user	
Brief scenarios description	- Services can apply their control rules over the entity models in the simulator and the effects will be visible over the behaviour of the physical models (Ref. Architecture figure) - Different controllers in the superior layer of the simulator will have different priorities when they apply their rules on one entity. - Entity models are organized in a tree structure according to their heritage relationship. A son node can be used with its father node's model if the latter satisfies the control goal. - An entity model can have more than one model with extra states for some control goal, e.g. a room may only have occupied and empty state in an energy optimization control goal but it will have an extra state "will be occupied in 10 mins" if the comfort goal need it to pre-heat the room.
Running experimentation (if exists)	Software Simulation

Number of involved users during experimentation (if any)			
Is any Experimentation Lab still running?	Yes	*Until?*	*<Date>*
Involved Technologies			

Brief Technical description	Though a complete simulator would start from a complete physical model corresponding to its bottommost layer, up to abstract models at its uppermost layers, a practical simulation doesn't need to include all relevant layers. With the interfaces provided between every two layers, every simulation layer is independent from each other and can work alone. For example, it can simulate only the Entity Model level by directly taking entity information from the external BAL (Building Abstraction Layer) service and generating and pushing actuator data as output value. 1. Layers & Components • **Physical Model** is the software representation of physical entities in the real world. They are continuous-time and continuous-space models, using e.g. differential equations. The simulator can interface to a library of ready-to-use physical models if needed and relevant. Otherwise the models are simple abstractions of physical reality (like the movement of human beings) that are sufficient to provide the necessary stimuli to the upper layers. • **Device Layer** is where networked entities exist, including **sensors, actuators**. Sensors and actuators are intermediate between the legacy or non-networked entities and their abstract model which can be integrated into the system. Simulated sensors and actuators are independent instances of functions representing the behaviour of the real sensors and actuators. Some of them act only as interface between physical models and while some of them have their own behaviour functions which generate contextual data alone. Results of this layer are shared by all the entities in the upper layer. • **Entity Model Layer** contains two essential components: o Entity Model (Entity in the schema) is the basic simple model of building identified physical entity. It is modelled as automaton which only contains the intrinsic states of the entity. It can contain continuous variables of the entity as well. The simulator will provide a library of basic entity models with specified input/output value type, continuous variable type, states and transition events. o Classification Layer is the part of the functionality of the service (Ref the next section) which takes in charge of identification of entities and their current states. According to available data from sensors, it identifies events in the context and makes necessary state changes on target entities. BAL is an independent service which runs on OSGi platform and provides interfaces of different types: socket, REST, etc. 2. Interfaces and communication protocols • **Device/Classification Layer** (①). This is the output of sensors and input of actuators. As the entry interface of BAL service is already defined (data in XML format, refer to BAL specifications), the output of simulated sensors should respect this format. Data structure of exchanged actuator information will be discussed in the near future. At present, as a RESTful service, it accepts an HTTP POST request to receive sensor data. It can also go to read periodically an XML file in a specific containing output data of sensors • **Classification Layer/Entity Model (②).** The output of the Classification Layer contains information about the identified entity such as: type of the entity, its current state, value of its attributes, in an XML format exposed as a REST interface. • **Entity Model/Service (③).** This is the top interface of the simulator which will be used directly by the services to be validated. Both basic entity model and derived entity model are available here as well as some environment variables to be provided to services. Services will need to know the automata in details such as their input/output variable's type, states and transitions, and they should be able to give input to entity models and trigger transition events.

	• **Communication protocols appropriate for RESTFul Web Service.** We choose the REST style for our web service because it is based on the universally understood protocol HTTP and impose few constrains for design and development. By using the basic HTTP verbs, there is no need for developing other special interfaces which saves a lot of work and makes it easy to access. Other components of the simulator can query it to get or put data. A Publish-Subscribe style may be appropriate for the system/environment to simulate which is usually event-driven. • **GUI: MILESENS** based on an open source context simulator SIAFU will mainly provide animation of physical entities such as people, room space. Knowing the position of a moving person on the graphical background, it can generate as output data to simulated sensors the presence of a person in a given space, etc. It provides also the functionality of planning a person's itinerary during the simulation by interaction with user, e.g. mouse click on the room's layout. Physical models are presented on the MileSENS via the interface ④.
Adopted technical solution (potentially domain specific enablers?)	This component will depends on the Building Abstraction layer which is presented as a separate experimentation capability
Picture (Architecture, Interfaces, etc.)	 **Figure 29: MILESENS - Simulator Architecture**
Software availability	Under development

5.9 Conclusions

Based on the Requirements identified in section 4.3, and the description of the tools and capabilities, it is possible to trace the satisfied requirements, the functions provided by the capabilities, and the capabilities themselves. This theoretical approach supports the set-up of an experimentation lab by identifying the existing capabilities belong a needed function.

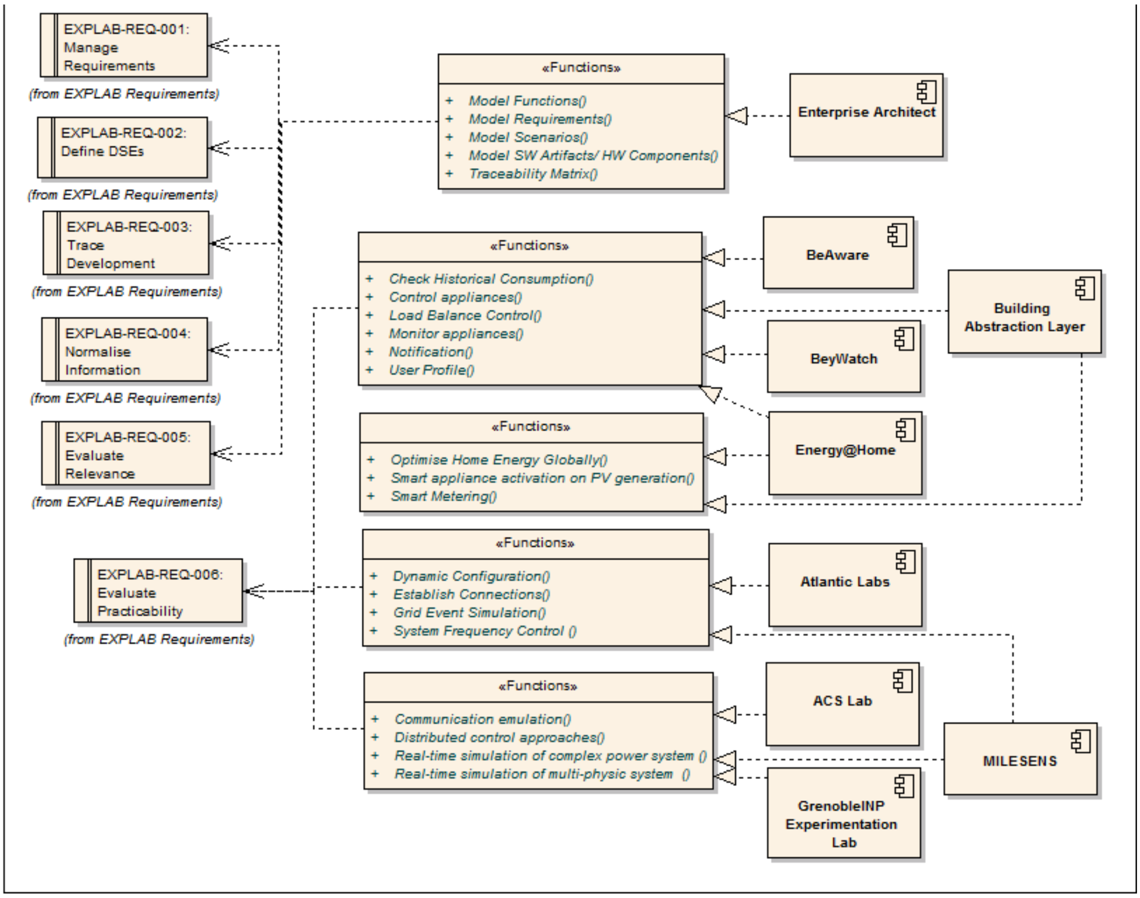


Figure 30: Tracing ExpLab Requirements, Functions and Existing Capabilities

6. FINSENY Experimentation Lab

This section describes the FINSENY Experimentation Lab architecture, its setup and configuration, intended experiments and expected results and, finally, the value that the lab adds to the FINSENY project.
In particular, the configuration depends on scenarios and/or domain specific enablers. For that reason this section provides a useful how-to for supporting the evaluation and validation phase. The how-to is based on selected sample scenarios (prominent FINSENY scenarios) and shows how to apply the common framework (i.e. methodology and tools) supported by the FINSENY Experimentation Lab and its specific configuration.

6.1 Architecture

According to the requirements described in section 3 and the capabilities which can satisfy them, it is possible to identify the architecture of the Experimentation Lab.
The experimentation in FINSENY phase1 is performed on the entities of the selected existing capabilities as described in chapter 5. Depending on the experimentation needed for the evaluation of the different domain specific enablers different configurations of the existing capabilities are needed.
As the FINSENY experimentation lab is used as one of instrument of the common framework for the evaluation of the domain specific enablers the following aspects have to be considered:

- Each domain specific enabler will consist of one or more entities
- Depending on the position of the these entities in the ICT and smart energy infrastructures, the role of these entities and the requirements on these entities will be different

Due to these aspects an appropriate architecture for the FINSENY experimentation lab has to be selected to order the entities of the selected existing capabilities according to their role in the ICT and smart energy infrastructure in the right position and make them comparable and more understandable. For this purpose a layered architecture has been defined for the FINSENY experimentation lab in phase1 as depicted in Figure 31, for common understanding the terminologies of the SGAM framework are adopted considering that in this phase no compatibility can be given to the existing requirements on the SGAM framework.
Using this layered approach facilitates also the evaluation of the gaps between the entities of the existing capabilities and the target domain specific enablers, which will be realised in the next phases of the FI-PPP project for the smart energy domain.

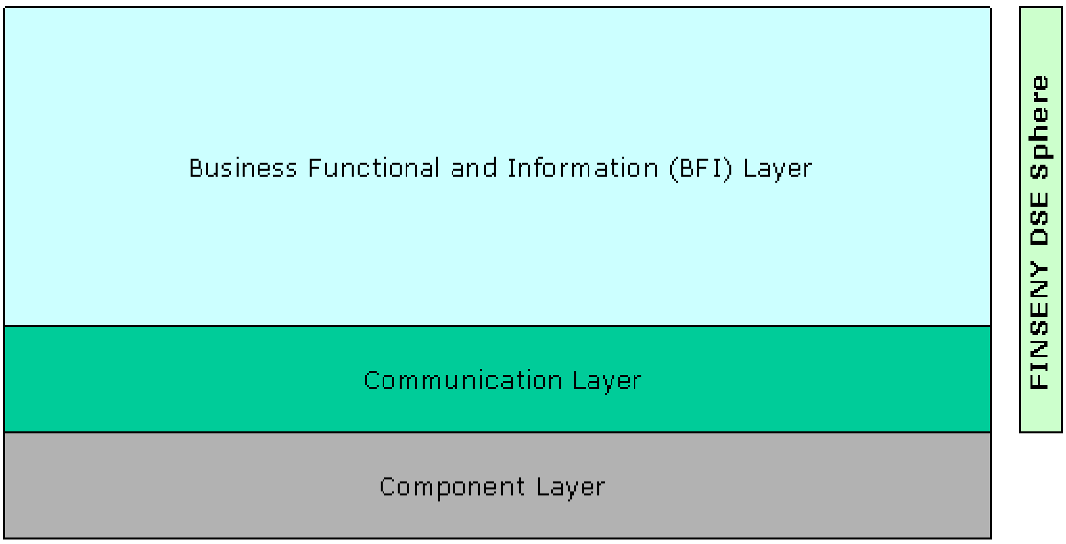


Figure 31: FINSENY Experimentation Lab Layered Architecture

BFI Layer:
The BFI layer represents the entities of the selected existing capabilities dealing with:

- The information exchange and services related to the market partner like BeyWatch energy retailer and Energy Customer Interface;

- The control and supervision of the smart energy components like GrenobleINP DSO SCADA and BeyWatch platform and GUI;

- The information model used by the business and functional entities like the consumption knowledge base of the BeAware service layer.

Communication Layer:
The Communication layer presents the entities of the selected existing capabilities dealing with HAN and WAN communication or enabling the energy components to smart one, like Energy@Home home gateway or GrenobleINP ICT enabler.

Component Layer:
The component layer is intended to show the entities of the selected existing capabilities, which are the real smart energy components like Energy@Home home appliances and GrenobleINP smart grid generation facilities or entities simulating these components like ACS laboratory RTDS and PC cluster.

The following figure shows a "*Mapping of the entities of the selected existing capabilities to the FINSENY Experimentation lab layered Architecture*"

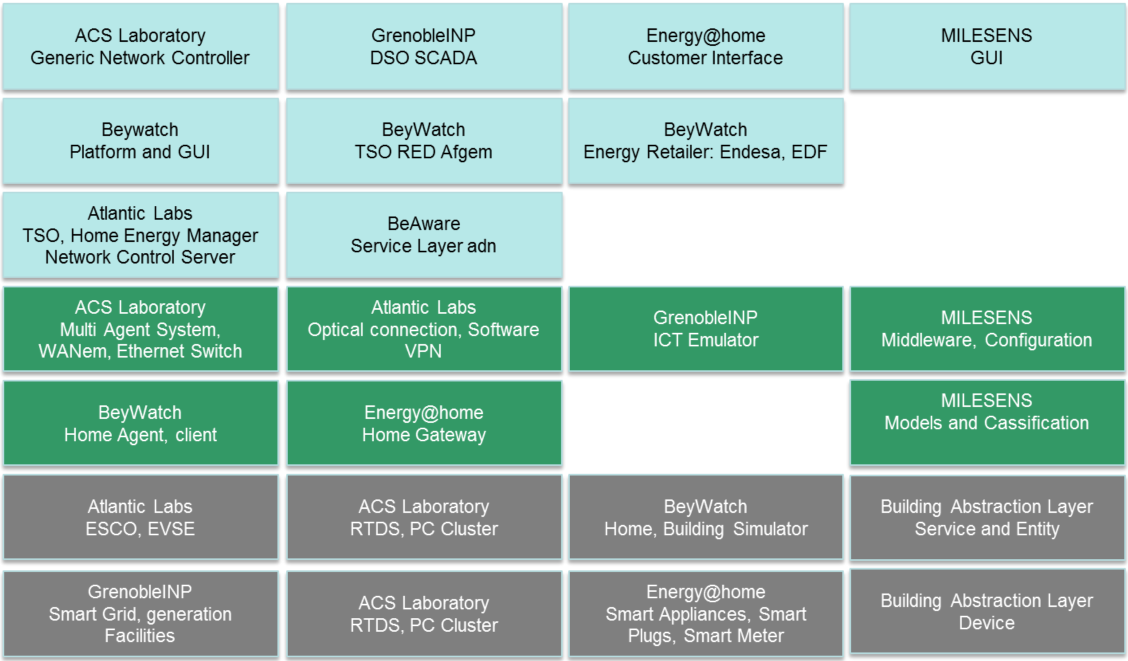


Figure 32: Mapping of the entities of the selected existing capabilities to the FINSENY Experimentation lab layered Architecture

The picture shows the available capabilities into the different SGAM layers. For each layer, each capability represents a potential alternative solution that can be adopted and combined through a proper configuration. The next sections describe some sample scenarios and show the specific configuration of the FINSENY experimentation set-up with the presentation of the entities in layered approach.

6.2 Set-up for Load Balancing by Using of Charging Stations

6.2.1 Rationale

Here below it is summarised a prominent scenario which is common for FINSENY WP2 and WP5.

The key components for a classical charging infrastructure are beside the energy generation the distribution grid and respectively the distribution grid operator (DSO), the secondary substations and the charging stations (CS). These components are more or less decoupled from a single operator. Especially the CS will be operated by a different operator as the DSO. The following figure shows this architecture.

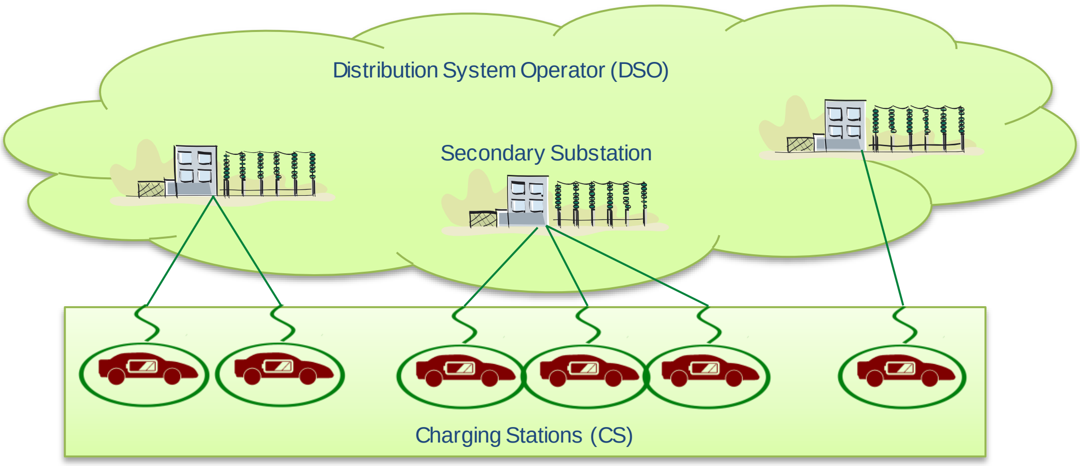


Figure 33: DSO-CS Architecture

New challenges for the secondary substations are the charging stations. These stations generate a high and unpredictable load, especially by parallel access. Furthermore the load depends on the season. This is a problem during the winter. In this season already the overall energy consumption is higher. Additional the electric vehicles consume more energy because the battery efficiency is lower compared to the summer. Thus the need for recharging is higher.

A comparable situation is the difference between the business days and the weekend. The load profile of the electric vehicles and the thus the load for the secondary substations is different. Additional examples for this not constant load for the substations are the current weather, vacation times, accidents etc. All these events or environmental conditions produce peaks in the energy grid. The dimensioning of the energy grid and the generation of energy on the basis of these peaks is not a common way. Consequently intelligent algorithms and methods or ICT enabler are necessary to balance the power grid.

One approach is the foresighted energy generation. For that ICT functions are necessary which collecting, analyzing and preparing the recharging behaviour of the electric vehicles. An assumption about potential energy hotspot can provided by means of these information. The generation of the needed energy can be take place on time and a wide range of energy sources (power plants, wind farms, photovoltaic etc.) can be used. Thus it is not necessary to use the energy from peak-load power stations.

Another approach is the controlling of the charging stations, especially the controlling of the charging speed. The overall load per secondary substation can be reduced by reducing the charging speed. For that the charging stations and the charging process must be included in the future internet and new ICT functions and ICT services must be developed. The execution of these new functions or enablers can be done via cloud computing and the charging stations must be extend with an ICT interface or an interface to the future internet. In this context the grouping of charging station should be also considered because the controlling of all single CS is a big challenge and may be impossible for the DSO. The grouping can be done via aggregators (operators of charging stations). The aggregation can be conducted per substation, over multiple secondary substations (see Figure 34) or between multiple secondary substations (see Figure 35).

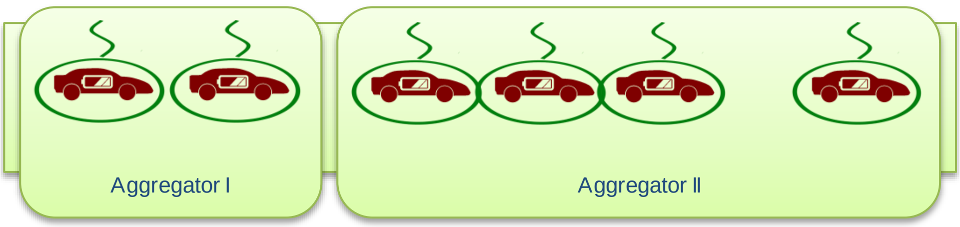


Figure 34: Aggregation over multiple secondary substations

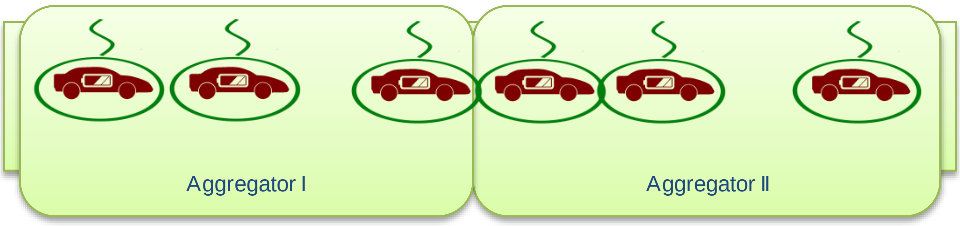


Figure 35: Aggregation between multiple secondary substations

The routing of the electric vehicles to dedicated charging stations is another load balancing approach beside the charging speed. This approach can be combined with foresighted energy generation. Intelligent ICT functions can navigate the electric vehicles to alternative charging stations if the load of a secondary substation or an aggregator is too high or will rise abruptly and exceedingly. Furthermore forecasting can be used to avoid such situation and the vehicle will be navigated directly to an alternative charging station, maybe to the next secondary substation.

The market should be also considered beside all the technical and infrastructure questions. Load peaks can be avoided or considered by means of special contracts between the DSOs and the aggregators. Special prices can be used to encourage the electric vehicles users to recharging at certain times. Or low prices on charging stations of a under loaded substation can be motivate electric vehicles users to switch from an over loaded substation.

The described load balancing methods via the using of charging stations are only basic methods. A mix of these methods is also imaginable. Beside the charging stations also complete micro grids can be used to balance the grid. Depending on the components of the micro smart grid the overall energy consumption of this grid can be reduced. Also these grids can be used to store energy in case of energy peaks in the grid. Possible components for that are for instance refrigerators or heat storages.

Finally the avoidance of new load peaks in the distribution network is important. Desirable is also the smoothing of fluctuation in the grid by an intelligent using of the charging stations and other storage elements. These imply the development of new ICT functions and the using of future internet technologies.

6.2.2 Experimentation Lab Configuration for Load Balancing

The load balancing by using of charging stations scenario or part of this scenario can be simulated with help of the ACS laboratory infrastructure. Figure 36 shows the principal set-up of the environment.

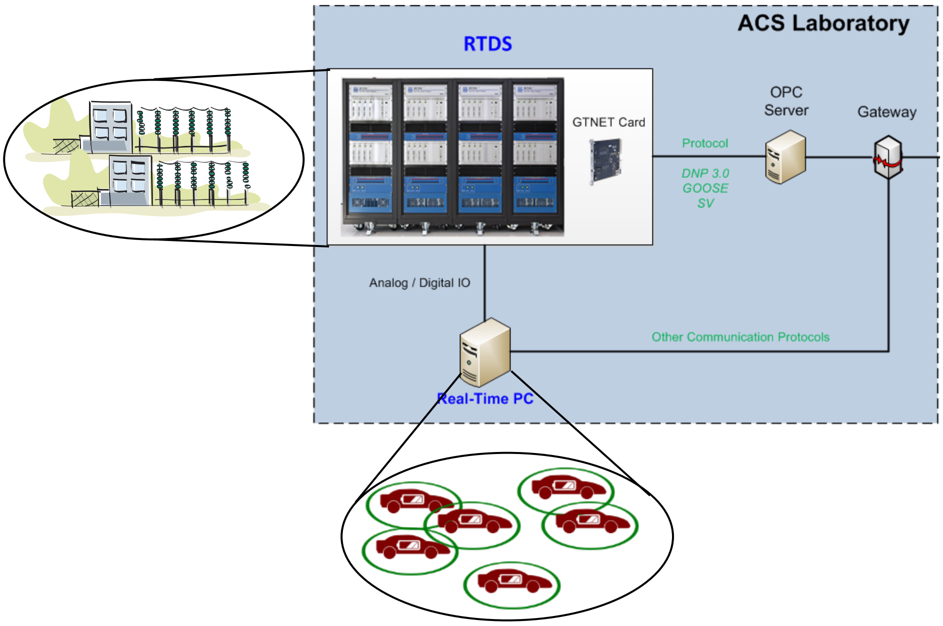


Figure 36: Set-up for Load Balancing by Using of Charging Station

The CS or the cars connected to the CS can be simplified to batteries. These batteries or the loads can be modelled or simulated by the real-time PCs or directly in RTDS. The example electrical network to be used in the experimentation can be simulated by the Real-Time Digital Simulator (RTDS). By generating different load profiles (that means different battery charges) the affects to the electrical network including the system frequency can be studied. The figure below shows several components of the ACS laboratory.

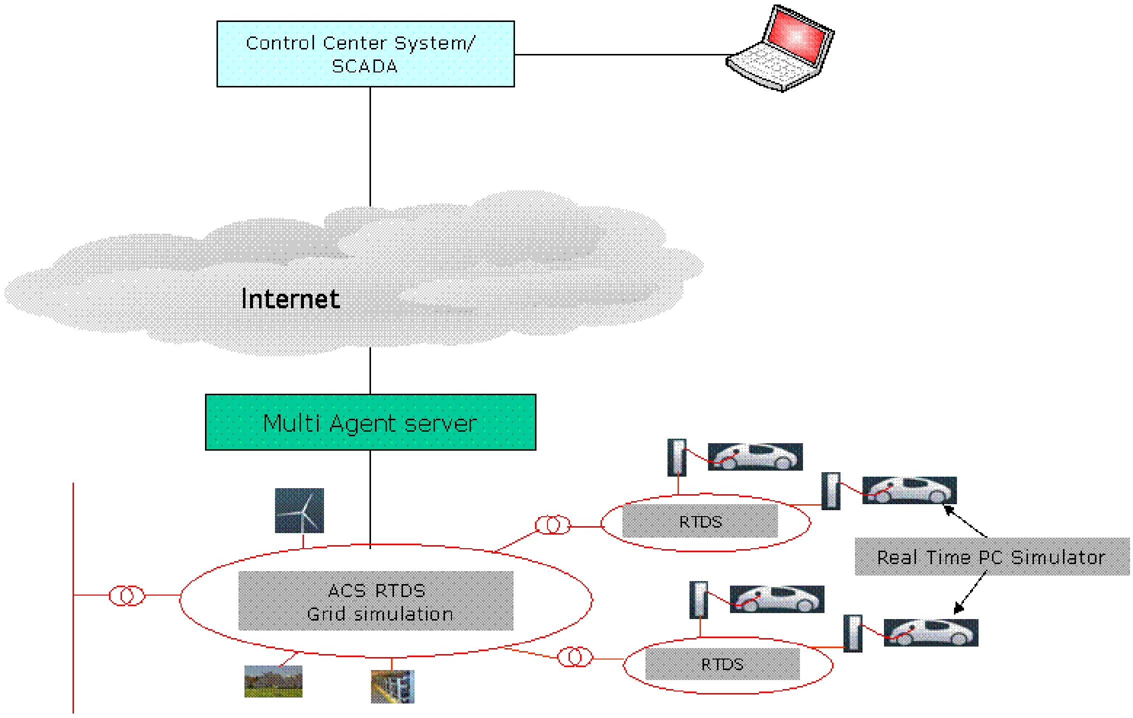


Figure 37: Components of Lab Configuration

6.3 Set-up for ICT Enabled Demand Side Management

Here below it is summarised a prominent scenario which is common for FINSENY WP5.

6.3.1 Rationale

The control of system frequency is a vital aspect of secure and stable power system operation. A continuous balance between active power generated and active power consumed by the load and losses is required to maintain frequency constant at nominal system frequency (50Hz). Any imbalance in active power will result in a frequency deviation. While precise instantaneous balancing of active power is not viable, frequency control ensures that the system frequency remains within acceptable frequency limits.
Frequency control can be called upon for a variety of conditions ranging from a gradual change in load levels over time to a sudden loss of generation or step increase in demand.

Of all the element of frequency control available to a Transmission Systems Operator (TSO), by far the most valuable is the availability of an autonomous 5s-15s response which constitutes *Primary* frequency control. Following that, a secondary (15s-900s) and tertiary (900s+) response period provides the operator with an opportunity to control additional resources in order to restore stability to the grid as well as to react to changes in the anticipated load pattern.

In conclusion, all methods available to increase the availability of this type of response in any grid are extremely valuable.

Background
A Fossil Fuel generator experiences a load shedding event:

- In day to day operation, generating stations will experience protective load shedding events;
- These events are protection actions which de-rate the generated power of the turbine until a stable operating condition is reached;
- The load shedding takes place under a pre determined ramp rate (Typically 3MW/sec);

- Lost generation requires balancing loads to stabilise the grid.

Controlling the Loads:

- For every 1000 cars charging on the grid, the load drawn is approx 2.75MW;
- By reducing the charge rates of EV's on the grid by just 50% during the initial minutes of the disturbance, the grid can be stabilised while spinning reserve is ramped up to take up the slack;
- The effect on EV charging is negligible.

Speed of reaction:

- Direct Unit Trips can drop generation by 100's of MW instantly;
- Wind turbines can shut down in blocks due to gusting winds, dropping 10's of MW generation from the grid instantly;
- This requires a fast reaction.

Generation Feedback

- Generators who experience Protective Load Shedding send a signal to the grid controller;
- These signals notify of imminent events;
- Instantaneous disturbances will first be identified by a frequency drop. (Nominally 50Hz for Europe).

Response Considerations

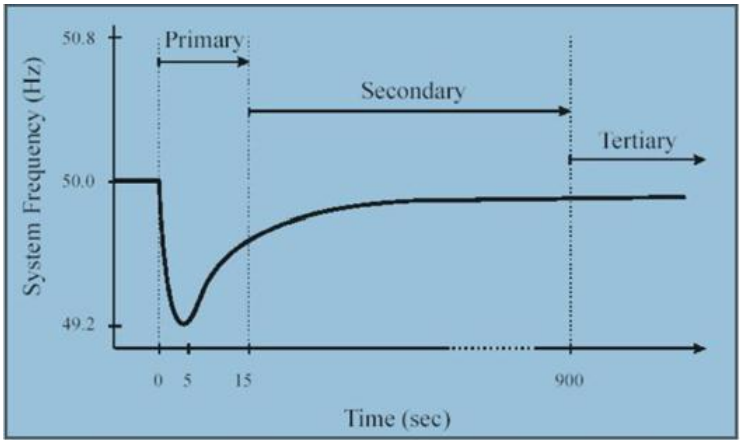


Figure 38 - Example frequency response diagram

Sliding scale of response based on Frequency deviation as illustrated in Figure 38:

- 50 to 49.8Hz – No reaction;
- 49.8 to 49.5hz – Reduce current 25% (5 min);
- 49.5 to 49.3Hz – Reduce current 50% (5 min);
- Below 49.3Hz – Stop charging (10 min);
- Ramp back for 15 minutes.

a) Primary control:

- Provided in a way of solidarity by all synchronously connected TSOs inside the UCTE area;
- To be activated within 30 sec;
- Time period per single incident 0 < t < 15 min.

b) Secondary control:

- Direct and automatic activation by the affected TSO;
- To be activated within 5 min;
- Time period per single incident 30 s < t < 15 min.

c) Tertiary control (minutes reserve):

- Telephonic and schedule-based request of the affected TSO at the respective suppliers;
- Time period per single incident t < 15 min up to 4 quarter hours or up to several hours in the event of several disturbances;
- Manually activated according to the 15 min schedule time frame or rather within 15 min.

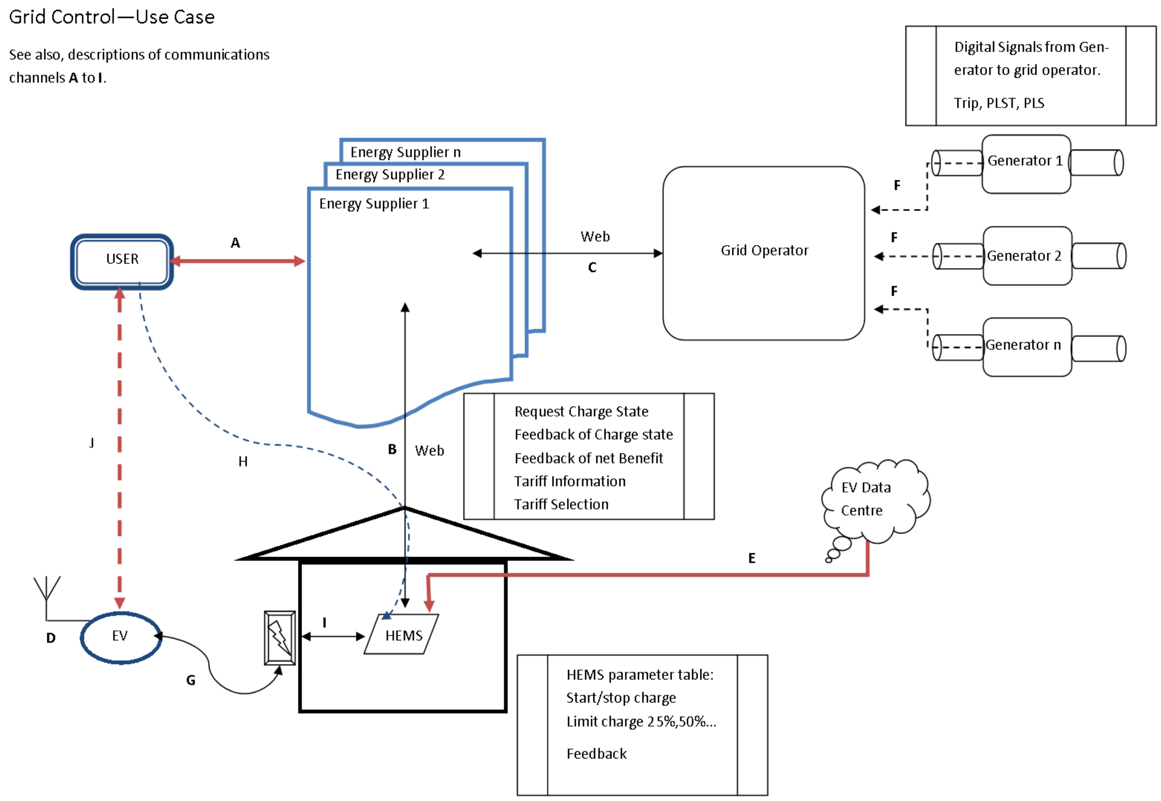


Figure 39: End to end diagram from energy generation to EV charge point

Description of Communications Channels in Figure 39:

A. The User communication to the Energy Supplier can be in multiple formats. He/she can change Energy Suppliers at any time. The communications can be via telephone, mail or internet.
B. The Energy Supplier sends tariffs, instructions to adjust charge status and credit/benefit statements to the Home Energy Management System HEMS. The HEMS sends availability, current charge level, net reduction and acknowledgements (on request) to the Energy Supplier. The communication takes place over the internet.
C. The Energy Supplier trades on the Energy Market. Included in this trading is the sale of Spinning Reserve. The Quantity of Spinning Reserve available is sent to the Grid Operator. The Grid Operator sends reaction requests to the Energy Supplier. Communication is via the Internet.
D. The EV communicates via GPRS to the Vehicle data Centre. Amongst the data transferred is the Vehicle State of Charge (SoC).
E. The HEMS will request the vehicle SoC data from the Vehicle Data Centre via the internet.
F. The Generator automatically sends the unit status to the Grid operator. These signals are hardwired digital signals, indicating when the generator is experiencing a Trip, A Protective Load Shedding event or a Protective Load Shedding Trip event. (Trip is an instantaneous event, PLS is a predetermined ramp down until stable and PLST is a predetermined ramp down that terminates with the generator off the grid)
G. The EV communicates with the charger via Pulse Width Modulation PWM to control the charge current to the battery from the charger.
H. The User sets preferences or views status on the HEMS. This is via touch screen or keyboard.
I. The HEMS will tell the charger to start, stop or limit the charge according to Grid requirements. This is done by Wi-Fi.
J. The User may be able to communicate with the EV to set charge timers or climate control timers via the on board computer.

In all cases consideration for user identification, data security and verification will need to be considered. Transfer of Net benefit data and loyalty benefits will also need to be exchanged and redeemable. All quantities such as kW's and credits will require to be validated for audit and transparency purposes. In conclusion, all methods available to increase the availability of this type of response in any grid are extremely valuable.

6.3.2 Innovation

Benefits

The Consumer	The Grid
• Customer incentive / Tariffs; • Loyalty Card system (free toaster); • kWh Credits; • Reduced kWh tariffs (while interruptible).	• Transfer of grid penalties to loyal consumers; • Greater acceptance on renewable generation.

6.3.3 Experimentation Lab Configuration for Demand Side Management

6.3.3.1 Atlantic labs (Ireland)

In WP5, one of the focuses is to manage the load being drawn from Electric Vehicles but, as illustrated by the "Application Domain" in the figure below, there are other relevant applications of this concept to control various other sources of load on the grid (see details in section 5.2). The initial experiment will be realised locally in TSSG (Waterford) and can be distributed across multiple sites as illustrated in the figure below.

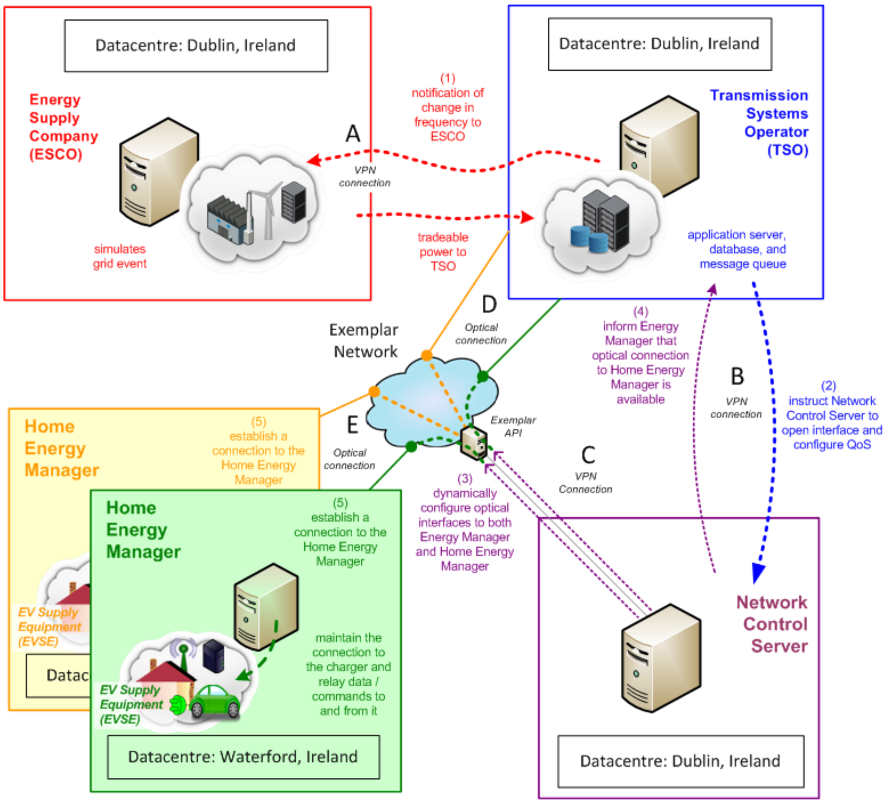


Figure 40: Connectivity diagram for Atlantic Labs testbed

The figure below shows the specific configuration of the FINSENY experimentation set-up with the presentation of the entities in layered approach for DSM scenario using the Atlantic lab.

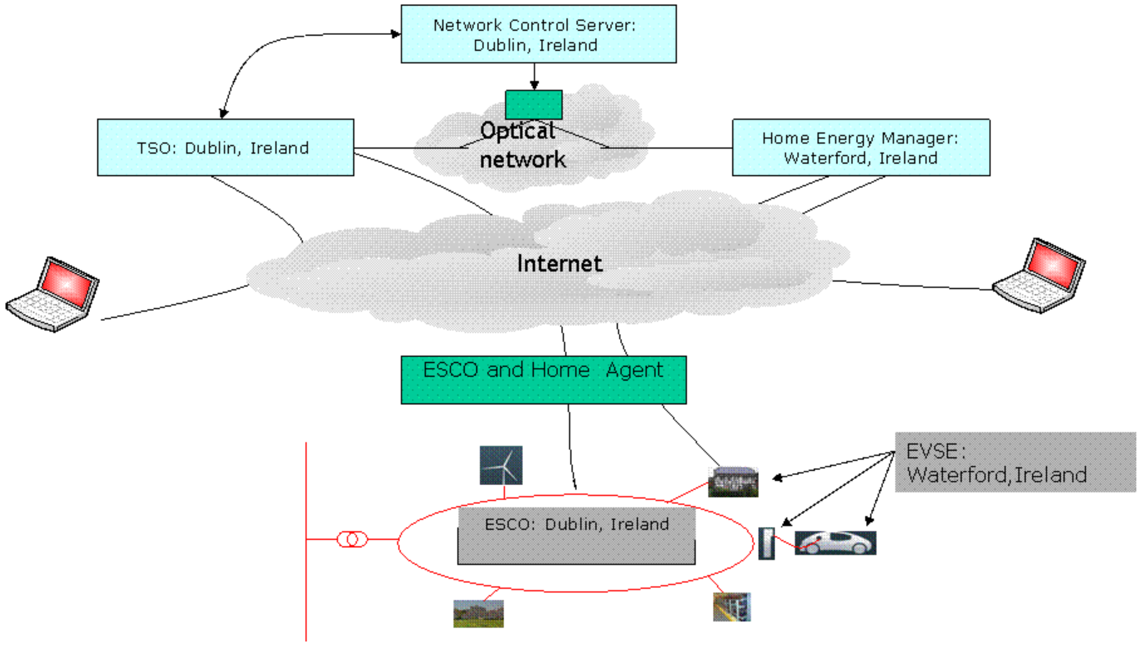


Figure 41: FINSENY Experimentation Lab Set-up Configuration for DSM with Atlantic Lab

6.3.3.2 Grid Simulation (RWTH Aachen, Germany)

In addition to the described lab facilities *Atlantic Labs* in Ireland, the electrical grid can be simulated at the real-time simulation lab of RWTH Aachen, Germany, run by the Institute for Automation of Complex Power Systems (ACS).

The core of the ACS simulation platform is the RTDS® (Real-Time Digital Simulator) Simulator, which is a fully digital electromagnetic transient power system simulator. The simulator works in continuous, sustained real time. That is, it can solve the power system equations fast enough to continuously produce output conditions that realistically represent conditions in the real network. Because the solution is real time, the simulator can be connected directly to power system control and protective relay equipment.

Within the proposed use case realization, the real-time simulation of the electrical grid will be an important step in the validation of the concept. Within the FINSENY experimentation, a proof-of-concept is planned with the proposed simulation setup.

The figure below shows a possible test case environment setup with the main units RTDS, real-time PC and Control Center System:

- RTDS provides real-time simulation capabilities of the electrical grid in a realistic continuous mode;
- The real-time PC enables the modelling of different loads connected to the grid;
- The Control Center System represents a monitoring and load shedding control unit.

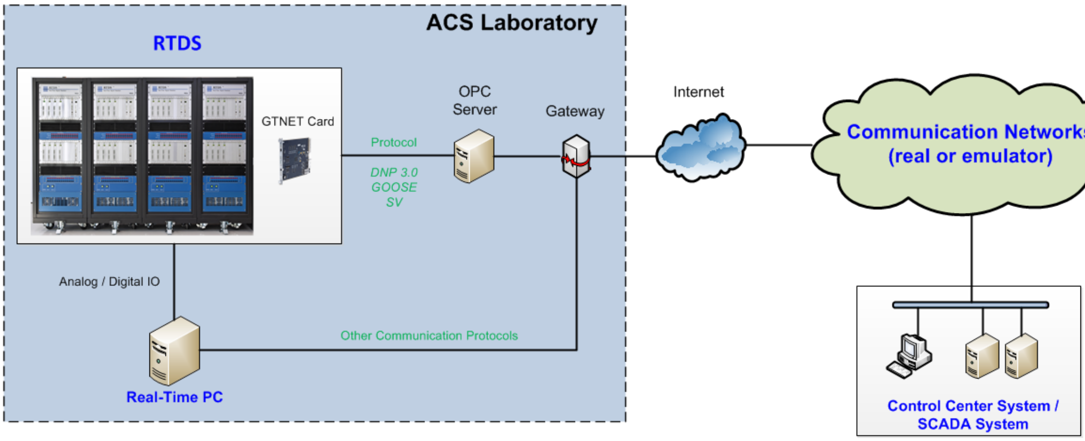


Figure 42: Possible RWTH simulator environment setup

The Figure 43 shows a more detailed representation of the modelling capabilities. The blue arrows visualize the information exchange between the described entities.

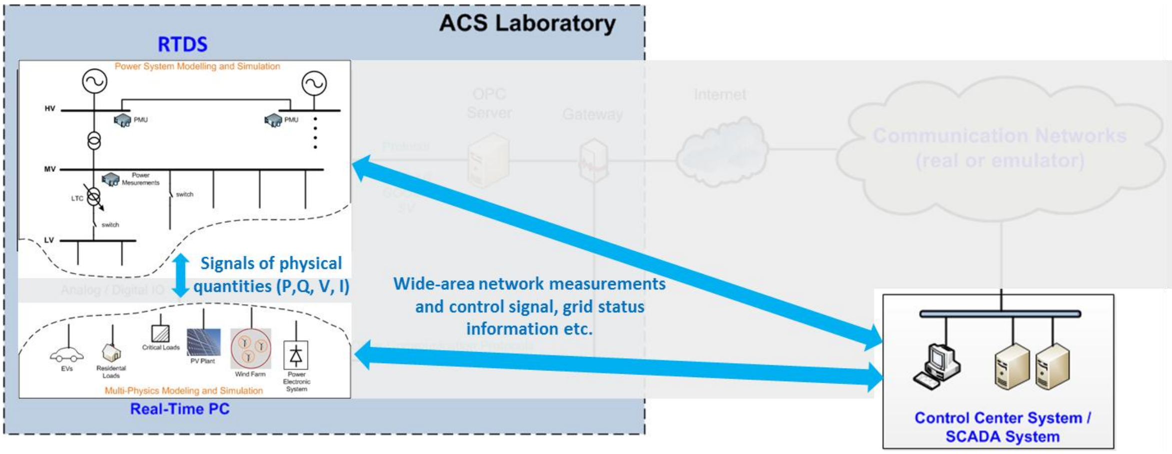


Figure 43: RWTH simulator detailed modelling capabilities

The Control Center System with the tasks of monitoring and load shedding control will evaluate measurements received from RTDS and derive control commands enabling the required load shedding actions. The control commands will be sent to the respective control units of the loads like the smart charging unit for e-vehicles or the home energy management unit. Due to the real-time nature of the grid simulation, the control actions taken for load shedding will directly be seen as impacts to the grid conditions, so that the effect of the concept can be verified. The communication can be set up via Internet but will be realized locally in the lab for simplicity reasons. However, communication characteristics can be emulated as described below.

Considering that communication infrastructures are expected to have a growing impact on performance of future power systems, a communication infrastructure emulating network behaviours inside the lab can be used to complement the use case. By the use of WANem (Wide Area Network Emulator) the setup of a transparent application gateway can be realized, which can be used to emulate WAN characteristics like time delay, Packet loss, Packet corruption, Disconnections, Packet re-ordering, Jitter, etc.

6.4 Intended experiments and expected results

In order to provide a clear picture and scope of the Experimentation Lab set-up, a clarification of the intended experiments and expected results is going to be presented by using the following table. Some of the experiments, even if not planned for the Phase 1, are mentioned and suggested for the Phase 2.

EXP LAB – CAPABILITY (reference to facility and lab description)	**INTENDED EXPERIMENTS**	**TYPE OF EXPERIMENTS**	**WAY OF CONDUCTING THE EXPERIMENTS (detailed)**	**EXPECTED OUTCOMES/RESULTS**
ACS Laboratory - RTDS (par. 5.1)	Provide proof-of-concept for frequency recovery use case with the support of mass storage in distribution network. Related to WP5 use case 'ICT enabled Demand Side Management' WP2 DSE - Verify the impacts of different communication link characteristics in the control loop between the Control Center System on one hand, and the Grid and Load systems on the other hand.	Simulation in real time digital grid simulator Emulated communication channel between the Control Center System and the simulated grid and load systems	- This experiment/simulation supports the WP5 e-mobility experiment by providing the simulation environment for power systems and connected storage on distribution level. - Different communication environments (packet loss, latency, etc.) will be emulated and their implications on the management and control of the grid and load systems will be analysed	The effects of the charging behaviour of mass charging of storage entities connected to the distribution level on the system frequency shall be investigated Understanding on the impacts of different communication systems for the SG communication. The experiments are set up as simplified examples and shall lead to indications of impacts as input for further studies and trials.
Atlantic Labs - TSO, Home Energy Manager, Network Control Server (par. 5.2)	- Demonstrate the ability to poll the usage of power on the monitored vehicles (what they are drawing) - Demonstrate the operation and control of charging units (Electric Vehicle Supply	A trial site will be developed in Ireland with real world communications and EVSE equipment to demonstrate the feasibility of	- The EVSE will be connected to a Home Energy Manager (HEM) via Wifi. The TSO will be able to be able to periodically poll the HEM to request the latest SoC of the EV - The control of the EVSE will be determined by the fluctuations in Grid frequency. These fluctuation signals will be simulated, though the turning on/off of the EVSE will be carried out on a	- A demonstration will show the feasibility and practicality of allowing a Grid Operator to poll the HEM to determine the charging status of the EV - A demonstration will show the feasibility to allow the

	Equipment, EVSE	demand side management of EVSEs.	real testbed.	Grid operator to instruct the HEM to turn the EVSE on/off in the event of a significant fluctuation in grid frequency.
BeyWatch (par. 5.3)	User is informed on the DSM signals, incentives/counter incentives and re-scheduling. The user can then reject or accept them.	Emulation using real measurements from the BeyWatch EDF trial in Paris	The DSE receives a DSM signal, the incentive/counter incentive signals, the new scheduling and are shown to the user GUI according to the contract. By default the signals are accepted by users have the option to reject them. 1. The user doesn't do anything, the new schedule is applied. The user accesses the GUI and rejects the re-scheduling.	The information is available from the user GUI. Depending on the action done by the user: 1. New schedule is running. 2. There is no change in the schedule.
BeyWatch (par. 5.3)	Flatten demand curve	Emulation using real measurements from the BeyWatch EDF trial in Paris	1. One DSM signal is sent to several households in a defined area. 2. By default, these signals are accepted by the user	The overall demand curve is changed accordingly.
BeAware - Service Layer and (par. 5.4)	Verify the completeness of information model for supporting the use case scenarios (e.g. detailed consumption information) and of APIs specification for supporting the features of the DSEs (e.g. Customer Energy Information Service, Home Energy Monitoring).	Data Log Analysis	Existing data from run trials will be analysed for identifying lack/completeness in the information model and suggesting refinement on use case scenarios (in particular the "detailed consumption information" – e.g. most consuming appliance, peak of consumptions, historical consumption, household energy breakdown, changes on Energy Consumption Profile of the End users, etc).	Detailed consumption (i.e. appliance consumption, historical consumption, energy breakdown, comparisons) and the user profile are key information for supporting the market mechanisms in the DSM scenarios. Results from the experimentation can be used for refining the design and specification phase (iterative development approach) of the functional architecture and the information model.
Energy@Home	Verify the capability to:	Test on trial site	1. Smart metering and control:	- to keep under control and

– Customer Interface (par. 5.5)	• collected metering data and implement on/off control on simple plugged energy loads. In this context it will also be respected the autonomy of Smart Appliances in order to assure the correct execution of its working procedure, its results and performance. • interface, in wireless mode, with Smart Appliances and with other user's devices and then to provide a broadband connection to internet for supervision and remote control. • exchange information between utilities and appliances in the houses to enable each customer to "self-manage" his/her energy behaviour depending on both power supply availability and price.	and data log analysis	a. Involving smart plugs and Smart meters will be collected metering data and implement on/off control on simple plugged energy loads. b. For the Smart Appliances the above remote load control will be subject to its control in order to assure the correct execution of its working procedure, its results and performance. For example, a smart washing machine, when requested to modify its consumption behaviour, shall assure the result of the washing cycle. 2. Communication and Data Protocol: a. it interface Smart Appliances and other user's devices and provide a broadband connection to internet. b. it collects energy data from the Smart Info and additional information from Smart Appliances, publish them in the HAN (Home Area Network) and use all collected data to control Smart Appliances and optimize their behaviour. c. it offers a web user interface and provides an execution environment (e.g. Java OSGi framework) to host third-party application (e.g. a SW component implementing the algorithm to calculate the energy price at a given time, provided by the energy retailer). 3. Detailed Consumption Information: a. display information on energy usage like instant power, historical data, contractual information and similar, from the whole house (coming from the Smart Info) and from every single smart appliance. The level of details and graphical layout of their user interface is freely defined by every device. b. transmit control message to Smart Appliances to request a modification of their behaviour.	manage (at home and outside), the household electrical devices avoiding power-off for excess of load. - to help the end user to choose the best dynamic rate when the retailers offer him/her multiple time-slot rate. - to offer at the end user the ability to sign a contract to more low power thanks to the possibility to keep under control the peak load, saving so money every month. - to have a consistent data-history, collected at a set time, in order to obtain important information about the contractual choice of energy retailers and to optimize the use of energy available. - to check the functionality of the management of priorities among all the smart appliances and the traditional ones.

			c. Configure Smart Appliances to modify their power consumption profile (e.g. a personal computer used to configure a thermostat to activate the controlled load only in certain time slots). d. The software application, which implements the user interface, could be local in the device or remotely hosted in another device (e.g. the Home Gateway) and accessed through web-services.	
Grenoble INP - DSO SCADA (par. 5.6)	A modelling of the automation system fully derived from French "Substation Numerical Control Command" with advanced features compliant with IEC-61850 is implemented in a substation. Few computer based automation systems are interconnected with their internal ICT network and emulate real industrial automation systems linked with internal communicative protective relays.	Implementation of test software application.	1. A dispatching centre which consists of an open SCADA system upgraded with observability and dispatches functions where a self healing function was implemented. 2. A connection to the main grid enables the supply security and the interaction with the market (selling/buying power depending on the price of the energy) 3. A real industrial SCADA application was developed in order to assess the interaction between the distributed control and the centralized control that exists.	Test control schemes using communication from a SCADA in 60870-5-104 to an emulation of a substation with have its internal bus in 61850
Grenoble INP - ICT Emulator (for the end of FINSENY project) (par. 5.6)	The real substation is connected to an emulated reduce scale distribution network. The public communication infrastructure is emulated with Virconel (virtualization of real routers/switches software application on a server). ICT public infrastructures for the Power System are designed by combining both ICT and energy models	Emulation	This experiment is suggested for the Phase 2. 1. The RTUs which are microprocessor based measurement devices, including interface unit to communicate, will play a role as a fault passage indicator (FPI) or fault recorder (FR). 2. On the one hand the current and voltage signals measured in magnitude and in phase at given point are sent by RTUs as inputs for some functionalities. 3. On the other hand, the control output signals will be sent to the control devices to complete the regarded actions. 4. For the demonstration, the RTU may be constituted of an acquisition board (3 current and 3 voltage) or program logic controller (PLC) with an Ethernet controller (TCP/IP 1000/100/10) card in order to send the measured signal during the fault (300ms @5kHz) under a text file or other standard semantic via a simulated communication system from BTH.	To be able to play with failures (short circuits) from both infrastructures especially on distribution network, a reduced scale distribution network with its given surrounding ICT emulated infrastructure is used
Grenoble INP - Smart Grid,	This demonstrator is able to implement energy management system at the grid	Emulation	1. The network implemented is a reduced scale one (20kV → 400V; 30 MVA → 30 kW) from a real French Distribution	Multi technology proof – demonstrating the ability of

Generation facilities (par. 5.6)	level and thus allowing the test of Advanced Distribution Automation functions such as “self-healing/defence” mechanisms using communication infrastructure and Optimal Power Flow control at the Smart Grid level.		Network with three substations (63/20kV) 2. Seven emulations of distributed generators (synchronous generators with active and reactive control) and nine controllable loads (induction machines and dimmers). The network components such as lines and cables are represented with their equivalent values. 3. Substations are emulated with internal electrotechnical components (transformers, breakers) and substation automation	multi DG technology operation with various control capabilities
Building Abstraction Layer (integrated in BOS - Building Operating System) (par. 5.7)	Demonstrate - capabilities for identifying non-networked non-digital entities from environment sensor data within a shared abstraction layer integrated in a broader “building operating system” - capabilities for dynamic reconfiguration of these entities, accounting for changes in the environment - capabilities for monitoring & controlling the identified entities and interfacing them to building applications through this abstraction layer	Test with real hardware appliances, possibly on other Finseny home/building Trial Sites	Non-networked appliances will be identified to and monitored through the Building Abstraction Layer, integrated in the Building Operating System 1. Identification: discovery of entities and their modelling with more or less specific and accurate models based on amount & of available data from sensors 2. Auto/re-configuration of entities: invert the sensors of two or several entities and the system should be able to recognize the new association between the entities and the sensors 3. Monitoring the states of equipments: environmental events detected by sensors can trigger state change of the identified entity through its model in the abstraction layer	The system can recognize these non-networked entities thanks to the Building Abstraction Layer and knowledge of corresponding domain ontology. The reconfiguration and monitoring functionalities should show the correct result expected in the scenario.
MiLeSenS (MultI LEvel Smart ENvironment Simulator) (par. 5.8)	Demonstrate the capability of supervising the behaviour of a system and the capability for offering a basic and minimum control to the system with some common and generic rules	Test with Building context Simulator,	The entities to be controlled exist in a simulated environment upon which operates the supervisory controller. According to pre-defined rules in the controller, it will effect the required transitions to maintain the system out of undesirable combination of states. Different scenarios will be implemented and executed in the simulator to cover all the desired features of supervisory controller. The simulator can host every scenario with adequate configurations described in a configuration file coming with the scenario executable file.	The system to be tested never goes to undesired combination of states in the simulated environment and the feedback result of the controller is visible (e.g. change of people’s behaviour according to environmental change)

Table legenda:
EXP LAB – CAPABILITY, a capability of a selected experimentation lab to be chosen among those layered in par. 6.1 "Component layer"
INTENDED EXPERIMENTS, brief description of the experiment(s) to be conducted on a DSE by using the experimentation lab capability
TYPE OF EXPERIMENTS, to choose among "Test on Trial Site", "Simulation", "Emulation", "Data Log Analysis"
WAY OF CONDUCTING THE EXPERIMENTS (detailed), series of steps to be followed to conduct the experimentation previously described
EXPECTED OUTCOMES/RESULTS, what to expect at the end of the experimentation in terms of measurable results (e.g. obtained data are correct as expected, a planned action has been overridden as expected, …..)

7. Summary

The aim of the task 8.1 is to set up an experimentation laboratory. This lab is going to provide a common framework, in terms of methodology & tools, for the evaluation and validation of enablers and the scenarios.

In particular, in cooperation with other work packages (i.e. work package scenarios WP[2..6] and work package consolidation WP7) this task has considered the requirements that the common framework has to satisfy for properly supporting the evaluation/validation.

For aiming this objective, the task has carried out a detailed analysis of industry best practices for supporting the evaluation/validation phase with widely accepted methodology and identified the key parameters for measuring the relevance of enablers and the practicability of scenarios.

From that analysis, it has been identified the need for supporting the management of requirements which will allow to derive the domain specific enablers. For that reason, it has been considered the methodology of Traceability. Indeed that is a best practice on Software Engineering coming from Capability Maturity Model Integration – CMMI. That traceability is already adopted in the work package scenarios WP[2..6] for the requirements and the use case scenarios. So it has been naturally considered its extension in WP8 for the evaluation of the enablers. Furthermore, the FINSENY project already adopts Modelling tools such as Enterprise Architect which supports that best practice with Traceability Matrix tool.

Whereas the real-time simulation and the experimentation with real capabilities have been adopted to evaluate and validate the practicability of the scenarios. Indeed, those are usual approaches for evaluating and validating behaviour and performance of algorithms and modelled systems, and for providing further details on potential system-failure scenarios and/or physical damage. That best practice avoids effecting the operation and security of the real power grid environment and in this perspective represents a fundamental step before to move forward on implementation and deployment in phase 2.
For supporting the expected functionalities and services of real-time simulation and experimentation, the experimentation lab has been defined as a configurable set of preliminary solutions and, for this reason, existing capabilities have been selected in order to match part of the required functionalities and requirements of the most prominent scenarios. Those existing capabilities are experiences and results of various other national and international research projects. In particular, it is very important to remark the importance of those existing capabilities: after the end of the phase 1, it could be reasonable that part of expected FI-WARE functionalities will not be available because under development and/or delayed. Those existing capabilities will allow to guarantee preliminary functionalities for enacting the selected FINSENY scenarios.

This document is providing also useful guidelines for showing how to apply in the next steps of the project (i.e. task 8.2, task 8.3 and task 8.4) the methodology, tools and capabilities provided by the common framework of the Experimentation Lab.
Indeed, for the evaluation of the relevance of the enablers, it has been considered a sample from WP6 Electronic Market Place for Energy (eMarket4E). The sample shows how to use in practice the methodology, and the tools of Enterprise Architect.
For the evaluation and the validation of the practicability of the scenarios, few samples from WP2, WP4 and WP5 have been considered for providing details on the different configurations of capabilities of the Experimentation Lab.

The practicability evaluation will be performed in task 8.3. Finally, traceability and Real-time Simulation and Experimentation results from Task8.2 and Task 8.3 will be inputs and supports the selection of broadly accepted and widely applicable enablers that are specific to the energy domain (mainly the activity carried out in Task 8.4).

8. References

This section lists the documentation used for editing this deliverable. The reader can use those references for further details on topics addressed in the document such as: Glossary, FI-WARE, Existing Solutions, Software Engineering Best Practices and Standards, Feasibility, CMMI, Traceability, Real-time Simulation.

[Gloss] General FINSENY Glossary of Terms v2.4

[FI1] Summary of Key Concepts in FI-WARE Vision - available at http://forge.fi-ware.eu/plugins/mediawiki/wiki/fiware/index.php/Overall_FI-WARE_Vision#Summary_of_key_concepts_introduced

[BeyWatch] FP7 project, ICT-2007.6.3, ICT for environmental management and energy efficiency, http://www.beywatch.eu

[BeAware] FP7 project, ICT-2007.6.3, ICT for environmental management and energy efficiency, http://www.energyawareness.eu

[PMHut] "Initiating Phase - Feasibility Study Request and Report" available at http://www.pmhut.com/initiating-phase-feasibility-study-request-and-report

[D6.1] FINSENY D6.1 "Electronic Market Place for Energy Building Blocks" - available at http://www.fi-ppp-finseny.eu/wp-content/uploads/2011/09/D6.1_Electronic-Market-Place-for-Energy-Building-Blocks_v1.0.pdf

[EA] Enterprise Architect – available at http://www.sparxsystems.com/products/ea/index.html

[IEEE-610] IEEE Standards Software Engineering, IEEE Standard Glossary of Software Engineering Terminology, IEEE Std. 610-1990, The Institute of Electrical and Electronics Engineers, 1999, ISBN 0-7381-1559-2.

[SEI-00] CMMISM for Systems Engineering/Software Engineering, Version 1.02 (CMMISW/SW, V 1.02); CMMI Staged Representation, CMU/SEI-2000-TR-018, ESC-TR-2000-018; Continuous Representation, CMU/SEI-2000-TR-019, ESC-TR-2000-019; Product Development Team; Software Engineering Institute; November 2000. – available at http://www.sei.cmu.edu/reports/00tr018.pdf

[Westfall06] Linda Westfall, "Bidirectional Requirements Traceability" – available at http://www.compaid.com/caiinternet/ezine/westfall-bidirectional.pdf

[Dufour10] C. Dufour, C. Andrade, J. Bélanger, "Real-Time Simulation Technologies in Education: a Link to Modern Engineering Methods and Practices"; March 2010 – available at http://www.opal-rt.com/sites/default/files/technical_papers/2010_Intertech_Dufour_RTLABApplications.pdf