Adoption of Model-Driven Design for Future Power Grid

ABSTRACT

Economical and environmental concerns push toward novel solutions for sustainable and renewable energy power grid. The Smart Grid emerges as promising solutions to cope with challenges of future power grid.

Such complex systems of systems, involve number of various stakeholders coming from different areas of expertise. Development of such systems, that eventually includes number of innovations as well, requires serious planing, modeling, optimizing, verifying and validating before system deployment.

In order to efficiently cope with system complexity, we adopt and apply model-driven methodology, to the problems concerned. Discovered system requirements and models developed are generically derived from several on-going Smart Grid projects. The main goal is to provide mutual understanding platform for stakeholders and to provide a basis of future standardization and design patterns discovery.

1. INTRODUCTION

Due to concerns on increasing pollution coming from coil based electrical energy generation; limited production capabilities of existing power grid; insufficient and inefficient ICT structure, appearance of new type of loads, and constant growing consumption needs, existing power grid has to evolve in a near future [17], [8], [20]. That evolution mostly referees to a possibility of accepting a large penetration of distributed energy resources (DERs), new type of intelligent loads (e.g. electrical and hybrid vehicles), and enmeshment of the ICT structure that will provide a more efficient electrical power grid and new customer services. According to [16], [14], smartness of a grid lies in it's ability to sense, monitor, and intelligently control the system in order to provide enable new energy services and energy efficiency improvements, that emphasizes the role of novel ICT structure of future power grid and embedded device in particular.

Introduction of large number of DERs that are by nature highly unpredictable and uncontrollable will cause power grid instability. In order to cope with this problem local encapsulation of DERs in technical and commercial terms,has been proposed in scope of Virtual Power Plant (VPP) concept, [7], [24], [5]. Similarly to microgrid, Virtual Power System (VPS) proposes a more general solution aggregating not only DERs and controllable distribution level generators but also loads and storages into one entity. Like that, VPS interacts with the rest of the system as one prosumer, being producer or consumer depending on current energy balance. With dynamic energy prices, that are seen in a near future, the main goal of VPS is to obtain local energy balance, and maximize usage of renewable energy from DERs while still keeping the power quality.

In order to achieve all previously mentioned, just like Smart Grid [25], VPP/VPS has to be equipped with highly advanced ICT structure that mainly consists of secure communication infrastructure, distributed devices for sensing, monitoring and controlling as well as centralized or distributed computing/control centers. Besides increased number of renewable generators and intelligent loads, this novel ICT structure represents a major difference between existing and future power grid. VPP/VPS is seen to be a cluster of Smart Grid that has many similarities with Smart Grid itself. For that reason, it represents very good case for adoption of concepts and methodologies development and proving since the same principles can be latter improved and extended for the case of Smart Grid.

Such a complex system as VPP/VPS is, requires already proven system engineering methodology and methods. Model-driven design is already widely used in software industry [23] and design of complex system (e.g. avionics), [26] and as a proven design methodology can be adopted and used first for formal modeling and optimizing and latter validation and verification of VPS/VPP as well as Smart Grid itself. Developed formal models can serve as main means to improve existing projects communication, correlation and components reusability. This paper aims at presenting simplified VPS design methodology that can be easily understood by stakeholders with a main goal of gathering information from various sources in order to efficiently define HW/SW functionalities of future ICT infrastructure of power grid.

The paper is organized as follows: Section 2, provides a review on VPP/VPS concepts and methodology used in ongoing projects. In Section 3, goals of the paper and work behind are described, while Section 4 gives an overview on methodology to be used. Developed models and tools are presented in Sections 5 and 6. Finally, Section 7 presents conclusions and future work.

2. STATE OF THE ART

Global energy demand growth urges for evolution of modern power systems in both commercial and technical terms. The Smart Grid [15, 22] concept has emerged as a promising response to these needs. Nevertheless, such complex system will require tremendous amount of data to be collected, transmitted and processed (in real time) which will pose another challenge to underlying ICT structures. Such scenario calls for partitioning and clustering of SmartGrid in certain autonomous modules.

On the other hand, efficient integration of heterogeneous distributed energy resources represents another challenge to be tackled by researchers and industrial experts. Several international research projects are dealing with this issue [1, 2, 3] from different aspects using custom approaches. Considering these facts, Virtual Power Plants (VPP) [11, 24] have been introduced as a means to aggregate DERs and represent them as a single, autonomous entity/module inside SmartGrids in technical and commercial terms. The VPP concept aims at leveraging security of energy supply coming from renewable resources with their optimal commercial exploitation.

The Virtual Power System (VPS), defined in scope of AlpEnergy project ([4]) can be seen as an extension of VPP. It actually additionally considers integration of consumers and prosumers (producer-consumer) entities in the system with renewable generation. VPS is conceived as modular and scalable system that adapts to local commercial and legal framework yet providing reliable energy supply and efficient exploitation of resources with respect to economical utilization. It is able to autonomously trade energy providing higher benefit for the system.

Nevertheless, such complex and inter-disciplinary concepts require appropriate instruments to manage user requirements collection and design complexity. In order to efficiently manage design of complex systems many system engineering and modeling concepts and languages have been introduced. In other environments, system-level specification languages have been introduced and are by now fairly widely used. AADL [9] has been introduced mostly for the purposes of modeling complex systems in avionics and space domain with an aim of expanding to other field. SysML [13] has been introduced as an extension of UML [18] for specification and modeling of complex systems including hardware components, personnel, facilities etc. Recently MARTE [12] has been proposed as a UML profile for modeling and analysis of real-time and embedded systems.

In this paper we use UML/SysML language as it is currently the most widely accepted specification instrument and modeling standard in both industry and academia. UML has also been already introduced as a modeling tool for finance issues and through the CIM [6] standard in power systems engineering. Moreover, model-driven design has been already proposed for similar problems in energy-efficiency in [21].

3. PROBLEM STATEMENT

Besides numerous projects in Smart Grid area as presented in a previous section, that include various analysis and implementations of VPP, VPS and microgrids, a unique formal language for describing system requirements, and provide an instrument for formal system modeling has not been accepted. This largely effects projects coordination and makes conclusions and best practice extraction very hard. Moreover, lack of such mutual language makes identification and reusability of mutual components and standardization of such a components, protocols and interfaces much harder. We aim at proposing a model-driven methodology to be applied to the case of VPP/VPS and deriving a generic model by combining results from several on-going VPP/VPS projects. Such a model should serve not only as a mutual understanding platforms amount stakeholders and project leaders, but should evolve depending on project results with a final goal of providing a basis for standardization and VPP/VPS formal design patterns. In order to make such a model more complete, we provide a beta version of tool that can serve for local consumption and storage scheduling in order to provide efficient usage of local DERs and lower total price of electrical energy used from the rest of the grid in dynamic pricing system.

4. SIMPLIFIED MODEL-DRIVEN METHODOLOGY

As previously mentioned, model-driven system design has been already widely accepted in software industry and complex systems design [23], [26]. As industrial standard for modeling software-intensive systems Unified Modeling Language (UML) and it's customization for a more general purposes, System Modeling Language (SysML), are seen to be the mail drivers of model-driven design, [10].

Since traditional power systems design does not include model-driven methodology not SysML, methodology presented on figure 1 is highly simplified in order to make it easier to understand and accept by different stakeholders that future power grid will include.

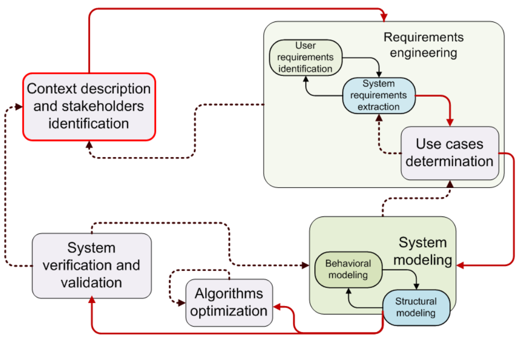

Figure 1: Simplified model-driven methodology for VPS

As any complex system development, it starts context description and stakeholders identification. Context description refers to unambiguous definition of system environment in terms of surrounding systems and it's relation to the system under development. This the basis for stakeholders identification that are de facto sources of users requirements.

Requirements engineering that includes collecting, tracing, analyzing, qualifying and managing user requirements [19]. The main purpose of this step is efficient extraction of system requirements and it's refinement trough use case that latter traced to concrete components that are identified in next step.

Despite the fact that various authors propose structure modeling before behavioral, or vice versa, developing system model is an iterative process. It starts with behavioral modeling of each system functionality presented trough use cases and assumed set of components. In order to keep model as simple as possible interaction based behavioral modeling presented troughs SysML sequence diagrams are suggested but not necessary. While progressing with behavioral models of system functionalities, system structure changes in terms of growing of components number or changing of the components. In some cases, structural change will affect additional behavioral change. The final result of this step is consistent structural and behavioral model.

Before actual deployment, achieved models have to formally validated and verified. Moreover, very same model or it's parts serv as a basis for functional optimization of the system that, in particular case, can be done trough under-development tool whose main purpose is energy balance algorithm optimization.

5. VIRTUAL POWER SYSTEM MODEL

Methodology presented assumes system whose development is just about to start, clear identification of stakeholders and possibility of requirements collection directly from stakeholders. On the other hand, Smart Grid and so VPP/VPS are still concepts under research that makes it hard to clearly determine stakeholders and get in touch with all of them. For that reason, some of the requirements that follow are extracted from the ongoing projects. The goal of the models latter derived is to serve as a generic mutual understanding platform and basis for once clear identification of VPP/VPS components and functionalities.

5.1 VPS context and stakeholders

As a first step of system modeling, context description and stakeholders identification heavily effects all the steps of system modeling that follows. Combining information from various VPP/VPS projects, we managed to identify existing as well as future stakeholders and VPS subsystems. Figure 2 represents VPS context in terms of surrounding systems. Note that particular systems and components (e.g. controllable generator) are presented outside of VPS in order to emphasis new functionalities and components that VPS should introduce.

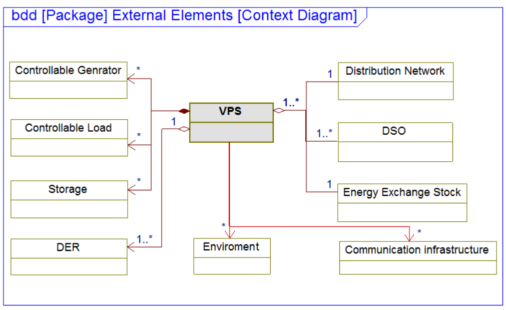

Figure 2: VPS context

Starting with initial, abstract system description and derived context diagram, stakeholders are identified. They include a subset of stakeholders from the VPS context but also additional stakeholders who are interested "non-functional" VPS requirements (e.g. legal ones). A full set of identified stakeholders, including individuals and organizations, includes: Distribution System Operator (DSO), Dispatcher, Energy Exchange Stock, Energy market legislation, Local authorities, Equipment producers, VPS shareholders, Home users, Industrial users.

Since both, Smart Gird and VPP/VPS are concepts under development, it should be no surprise if additional stakeholders would be identified latter. That, in-fact, will require additional system changes and re-engineering. Moreover, some of the stakeholders and contest systems identified above are seen to just appear or to drastically change in the future (e.g. Energy market). This will trigger additional changes in system requirements and model proposed.

5.2 VPS requirements engineering and use cases definition

Starting with stakeholders identified, requirements engineering starts as collecting user requirements from stakeholders. As already stressed, due to research nature of VPP/VPS, some stakeholders will justa appear in near future and for that reason it is not possible to collect real requirements from them. For that kind of stakeholders some of the requirements were assumed, based on predictions on energy market and practice from relevant projects. While focusing on functional requirements, user requirements are carefully collected and organized in tree packages representing communication, commercial and technical requirements. For each user requirement separate sequence diagram has been developed representing the way that requirement can be satisfied trough interaction of initially assumed VPS system components and surrounding systems. As the system modeling progress, set of system components could change, and those sequence diagrams could be re-engineered after validation and verification phase. Figure 3 gives a sequence diagram for one the requesters identified (DSO requests defined amount of energy during defined period). For each interaction that appear amount VPS and surrounding systems as well as so far identified internal components, system requirements is defined.

The final set of system requirements is organized in a similar way like the user's ones and presented on figure 4.

With defining use cases, behavioral modeling of the system practically starts. A few very basic use cases, describing the main functionalities of the VPS are defined first and related with stakeholders and internal components involved in operations they describe (figure 5). While defining those main use cases, major user requirements were taken into account. This is the starting point for understanding the system functionalities. Each stakeholder can identify use cases he is interested in, and latter follow decomposition only on those use cases without bothering with the functionalities of the rest of the system. One should constantly keep in mind that, according to the methodology presented here (figure 1), the whole modeling process is rather iterative than straight-forward and that latter phases could effect additional changes. For instance, during validation of behavioral modeling phase, it could come out that additional components are needed or that some the identified components has to be involved in some other operations as well. Still, diagrams presented here are the final ones and do not require any additional changes with defined set of require-

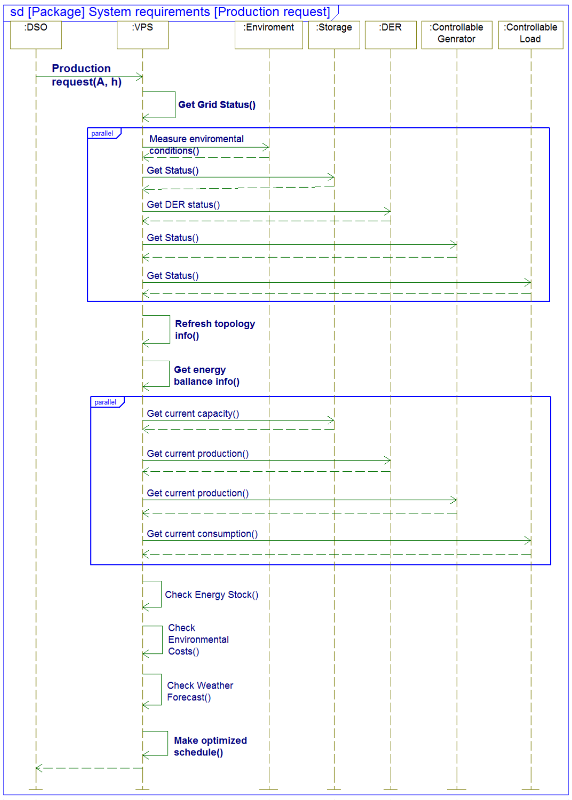

Figure 3: Production request sequence diagram

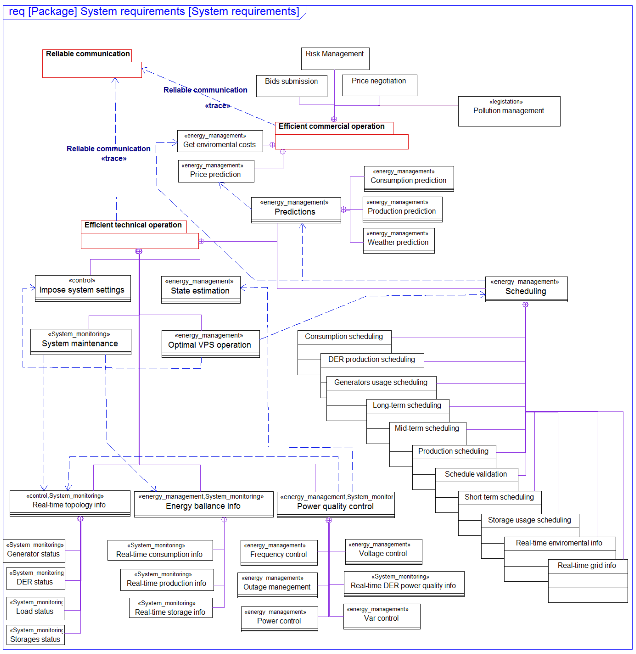

Figure 4: VPS system requirements

ments and use cases.

Each of the major use cases is fatherly decomposed into more fain-grained diagrams with atomic uses cases directly traced to system requirements. As a demonstration, figure

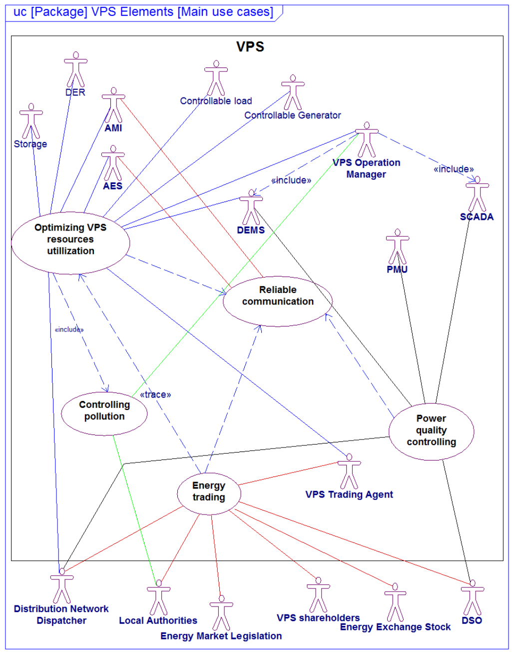

Figure 5: VPS main functionalities

6 represents decomposition of use case "Optimizing VPS resource operation".

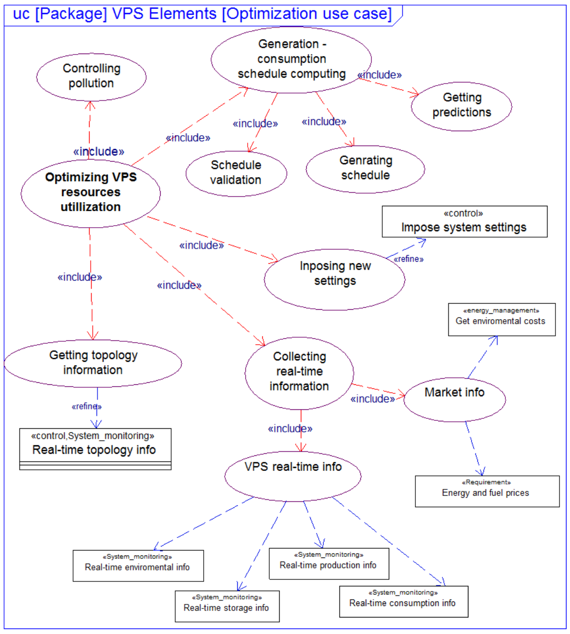

Figure 6: Decomposition of major VPS functionalities

5.3 Behavioral and structural model

Even thought that structural and behavioral modeling practically started from the very begging of the methodology (use cases defining system functionalities and assume a basic set of components that requirements/use cases are traced to) the final system model defining starts in this step. For each VPS functionalities/use cases a separate sequence diagram has been derived describing how particular use case can be satisfied troughs interaction of various components. Initially each component has some predefined set of functionalities that is being extended and additionally defined trough each sequence diagram. Figure 7 gives a system behavior in case of use case "Collecting real-time information".

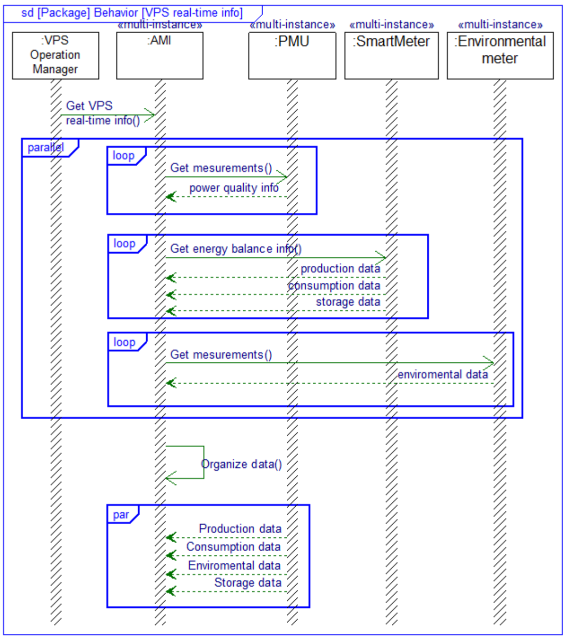

Figure 7: VPS behavioral model example - Collecting real-time information

A full set of VPS components together with identified functionalities, excluding the ones that could exist separately and being already presented on figure 2, are given on figure 8. A short description of those components follow:

- SCADA and DEMS are components that already exist in the power system with a main purpose of assuring the power quality and providing the basic system controllability and monitoring.
- Advanced Measuring Infrastructure goes way beyond simple consumption measuring and includes remote measuring of additional parameters need for proper production planing and consumption scheduling (e.g. temperature, humidity, VaR, etc.).
- Actuating Embedded Systems are extending the SCADA functionalities caring with particularities of each smart load, DER or storage being included into VPS.
- Trading Agent handles all the comitial issues of the VPS including energy trading, stock state reading, etc.
- VPS operation manager is the brain of the system that schedules production and consumption applying advanced algorithm to be developed, checking the feasibility of production, etc. It is seen as a centralized one at the moment, but could evolve to distributed solution based on agents technology.

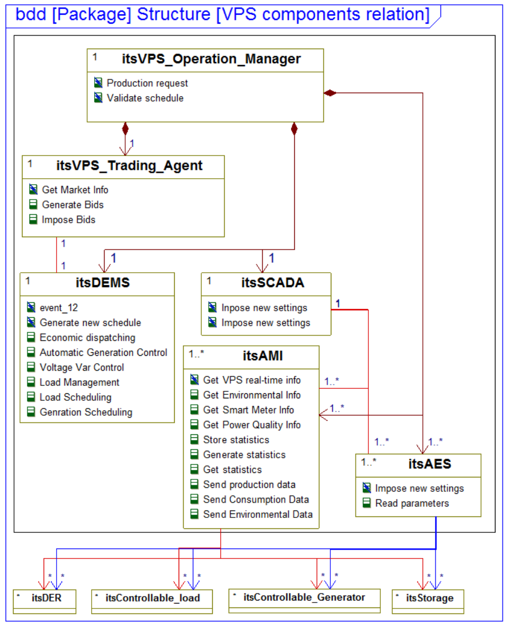

Figure 8: VPS Block Definition Diagram

Finally, without going into any details we present VPS internal diagram giving details about information flow between components and hopefully helps in defining the communication standard for VPS (figure 9).

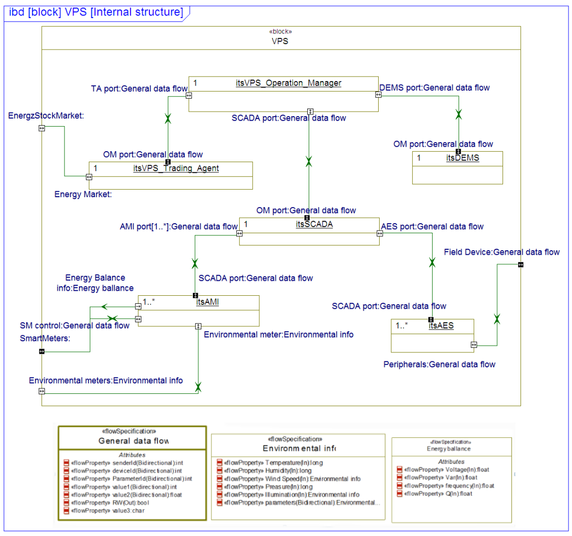

Figure 9: VPS Internal Block Diagram

With a clear functional identification of components, one that model has been verified, same methodology described here can be used for deriving a fine-grain model of the components.

6. ALGORITHMS OPTIMIZATION

Each of the components identified should provide some set of functionalities. As already stressed, the main purpose of this models is to give a straight definition for further components deployment. Moreover, our final goal is not only to provide a description of interfaces but also to propose the most optimal way that those functionalities could be achieved. For that purpose, we are, presently working on frameworks that will serve for exploring the best solution algorithms for all the functionalities of components identified. At the moment we are focusing on scheduling algorithm optimization by providing a framework able to compare solution obtained by particular scheduling algorithm with the optimal one being found with exhaustive exploration. At the moment it is very simple solution that assumes predictions of dynamic energy price, local production troughs DERs and several consuming tasks that can be preemptive or non preemptive. The main goal is to finish all the tasks till their deadline by rescheduling consumption according to the local production and current energy price in order to minimize the total price to be payed for using the external energy source (from DSO). Assuming tree consumers, optimal solution presented on figure 10 can save up 13% of energy bill when compared with one with no scheduling.

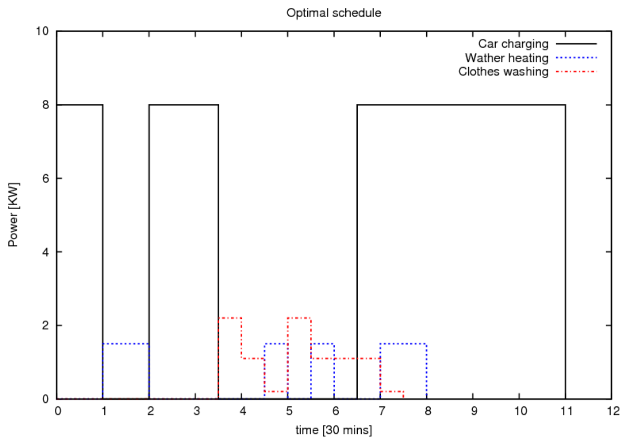

Figure 10: Consumption scheduling

7. CONCLUSION AND FUTURE WORK

VPP and VPS emerge as solutions to aggregating heterogeneous renewable resources and leveraging their cumulative capacities; on the other hand, this requires a strong presence of ICT supports beyond the traditional energy management ones, namely distributed and ad-hoc hardware/software solutions consisting of suitable embedded systems. These systems could represents building elements of Smart Grids being at the same time autonomous technical and commercial nodes. In sense of system structure and functionalities VPP/VPS should reflect Smart Grid solutions and in that sense simplified methodology given here can be considered as a case study for a more general case of Smart Grid design.

Our goal is to introduce model-driven methodology into this are of research in order to collarette on-going projects and provide a generic model that will serve as a mutual understanding platform among project leaders and stakeholders. Moreover we aim at fully providing a tool for exploring and comparing algorithms for optimal VPS resources usage.

Our future work includes identification of design patterns by comparing various concrete solution after being transformed into SysML models.

8. REFERENCES

[1] http://www.fenix-project.org/.
[2] http://www.addressfp7.org/.
[3] http://www.integral-eu.com.
[4] Alpenergy project. http://www.alpenergy.net/.
[5] E-energy project. www.e-energy.de/en/.
[6] J. P. Britton and A. N. deVos. CIM-based standards and CIM evolution. *Power Systems, IEEE Transactions on*, 20(5):10–16, May 2005.
[7] K. Dielmann and A. van der Velden. Virtual power plants (vpp) - a new perspective for energy generation. *Modern Techniques and Technologies, MTT*, pages 18–20, April 2003.
[8] H. Farhangi. The path of the smart grid. *IEEE power & energy magazine*, pages 18–28, January/February 2010.
[9] P. H. Feiler, D. Gluch, J. Hudak, and L. B. *Embedded Systems Architecture Analysis Using SAE AADL.* Software Engineering Institute Carnegie Mellon, June 2004.
[10] S. Friedenthal, A. Moore, and R. Steiner. *A Practical Guide to SysML: The Systems Modeling Language.* Morgan Kaufmann OMG Press, 2008.
[11] http://www.fenix-project.org/. *Fenix Project Bulletin*, 2008.
[12] http://www.omgmarte.org/.
[13] http://www.omgsysml.org/.
[14] http://www.smartgrid.gov/. *US Smart Grid projects.*
[15] http://www.smartgrids.eu/.
[16] http://www.smartgrids.eu/. *SmartGrids: European Technology Platform*, 2006.
[17] http://www.tradingeconomics.com. *Trading Economic Analytics*, April 2010.
[18] http://www.uml.org/.
[19] E. Hull, K. Jackson, and J. Dick. *Requirements Engineering.* Springer, 2011.
[20] K. Kobayashi. *Forecasting Supply and Demand up to 2030.* http://www.iea.org/, October 2009.
[21] A. Mady, M. Boubekeur, and G. Provan. Compositional model-driven design of embedded code for energy-efficient buildings. In *Industrial Informatics, 2009. INDIN 2009. 7th IEEE International Conference on*, pages 250 – 255, 2009.
[22] S. Massoud Amin and B. F. Wollenberg. Toward a smart grid: power delivery for the 21st century. *IEEE Power and Energy Magazine*, 3(5), October 2005.
[23] D. Milicev. *Model-Driven Development with Executable UML.* Wiley Publishing, 2009.
[24] D. Pudjianto, C. Ramsay, and G. Strbac. Virtual power plant and system integration of distributed energy resources. *Renewable Power Generation, IET*, 1(1):10–16, March 2007.
[25] Y. Serizawa, E. Ohba, and M. Kurono. Present and future ict infrastructures for a smarter grid in japan. In *Innovative Smart Grid Technologies (ISGT)*, Gaithersburg, MD, 2010.
[26] T. Weilkiens. *Systems Engineering with SysML/UML: Modeling, Analysis, Design.* Morgan Kaufmann OMG Press, 2007.