Developing Self-Organizing Smart Grid Systems by Means of S-BPM Concepts

Stephan Borgert,[1] Albert Fleischmann,[1] Markus Albert,[2] Ludwig Karg,[3] and Henri Oliveras[4]

[1] [ui!] - the urban institute, Julius-Reiber-Straße 15a, 64293 Darmstadt, Germany
stephan.borgert@ui.city
[2] B.A.U.M. Consult GmbH, Gotzinger Str. 48, München, Germany
l.karg@baumgroup.de
[3] VSE Verteilnetz GmbH, Heinrich-Böcking-Straße 10-14, 66121 Saarbrücken, Germany
markus.albert@vse-verteilnetz.de
[4] Stadtwerke Saarlouis GmbH, Holtzendorfferstraße 12, 66740 Saarlouis, Germany
henri.oliveras@swsls.de

Abstract. In this paper we describe how a subject oriented method is used for the communication between different professions (IT-people, electrical engineers, business people) and how the software for the management of a self organizing smart grid system is developed step by step in a subject oriented way.

1 Introduction

In this article we describe how subject oriented software design is used for the development of a holonic energy management system. The holonic paradigm advocates the use of autonomous and cooperative units, called Holons. Holons are organized in a flexible hierarchy in order to increase the agility and reconfigurability of the processes. We investigated whether this is valid for energy management systems.
The software for managing a holonic energy system is a realtime application and in the development process people with different professional backgrounds were involved. Therefore we need a language which supports the communication between the involved parties.
A traditional electrical grid was designed for one way interaction and this makes it difficult for the grid to respond to changing and rising energy demands of the 21th century. Smart grids introduces a two way dialog where electricity and information is exchanged between energy providers and its customers. A smart grid is a network of communication, controls, computers new technologies and tools working together to make the grid more efficient, reliable and greener. Smart grids enable new technologies to be integrated such as wind, solar, fuel cell heating stations and batteries for energy production and energy storage. The management of smart grids is often centralised and is based on predefined decentralised parts of the grid. Holons are a next development step in delivering

electricity or more general energy from producers to consumers. Holar energy systems are managed decentrally and the decentralised structure of the system is created on demand. The software required for the management of such a totally distributed energy system must reflect the structure of this approach.
The application of Holonic systems has been evaluated in the manufacturing industry [1], [2] and for control systems for intelligent buildings [3]. Theses investigations have shown that holonic systems are more robust. In these use cases robsutness was measured by the productivity over a long time. In the manufacturing industry the holonic paradigm promises to meet the upcoming challenges by introducing autonomous and cooperative units that drive the manufacturing process [1]. In the project Poly Energy Network (PEN) sponsored by the German Federal Ministry for Economic Affairs and Energy it has been investigated whether the holonic approach can be applied in managing energy systems.

In this article we will describe the properties of holonic energy systems and the resulting requirements for a energy management software. First we introduce the properties of a holonic system in more detail. From these properties we derive the corresponding system architecture. Based on this architecture the interactions of the various components are described. All this work was been done together with electrical engineers and business people. This means we have needed a language for the specification of realtime systems which supports the communication between energy and software engineers as well as business people. We used a subject oriented specification method which is based on the active voice of natural languages. The components of the architecture (Sensors, actuators, software processes etc.) are considered as active entities which do their work independent from each other and coordinate their activities by exchanging messages. We call these components subjects. In order to describe each subject we use natural language (German) but only active voice was allowed: Who makes what, when, by what means and communicates when with other subjects. The final specification of the subjects has been done by graph based diagrams. Based on these specifications the executable code was derived. Subjects are behaviour descriptions which are independent from their implementation. They are placeholder for active, autonomous communicating entities. In a later step it will be defined whether a subject is implemented with IT, humans, physics or a combination of them. This means a subject oriented specification is independent from the technology in which the various subjects will be implemented.

2 Fundamentals

In smart grids production, storage and consumption of energy is combined. This mix is managed by a system which balances energy fluctuation by switching on or off renewable energy sources. The management of this energy fluctuation requires the exchange of information about the status of the various entities in the energy network. In an energy network not only energy flows also information is sent and received. Smart grids are often managed in a centralised way or use predefined decentralised parts. The main threats of future energy networks are

the fluctuation of decentralised energy production and consumption, technical or human failures and cyber attacks. In order to overcome these problems the concept of holons has been applied. In the PEN project holons are self -organising smart grid systems.
The term holon has been derived from the greek and is used in the work of the Austrian-Hangarian philosopher Arthus Koestler. He develops his philosophical idea of Holons in the book with the title "Janus, A Summing Up" [4]. In that book the term holon means a whole that is part of another whole. Koestler [4] presented the definition of holon and holarchy. A holarchy is a hierarchy of self-organized holons,which behave as autonomous wholes in supra-ordination to their parts, dependent parts in subordination to wholes/sub-wholes at higher levels, and in coordination/synchronization with their local environment.

A holar energy network is a holarchy. In a holon the energy required by the consumers being part of a holon can be produced inside of the holon. If more energy can be produced than consumed energy storages can be filled up. In a period of time in which more energy is required than can be produced these batteries can be used as additional energy sources. A so called holon coordinator coordinates the production and consumption of energy inside a holon. In a situation in which not enough energy can be produced for the consumers inside a Holon the related coordinator tries to identify some flexibilities in energy consumption. The coordinator switches off some consumers which can interrupt or can reduce their energy consumption for a certain period of time e.g. cold storages. If there is still a energy gap the holon coordinator tries to find a holon which can produce more energy than currently consumed. These two holons can decide to merge in order to balance their energy consumption and production. This new holon produces the energy which is consumed. If the situation changes and both former holons can produce the consumed energy again by their own they split. Instead of one holon there are again the original two holons. This kind of decentralised energy management is more resilient to volatility and technical or human failures.

The software system for the management of such a holar energy system must act in realtime and is highly distributed. Realtime systems have timing constraints and interfaces to the external environment. They are event driven and must respond to external stimuli. It is usually the case in such a reactive system that the response made by the system to an input stimulus is state dependent. This means the response does not only depend on the stimulus itself, the reaction also depends on status of the control system. In general these responses must be executed during a certain period of time. Concurrent tasking is an effective solution to the design of realtime systems because it reflects the natural parallelism that exist in the realtime problem domain. Details about properties of realtime software systems can be found in [5] or [6].

The design of a holar energy system must be done by people from several professions e.g. electrical engineers, business people, lawyers, software engineers etc.. Therefore a language is required which can be understood by all members of such a interdisciplinary team but also allows to express all the requirements of a realtime system to be developed.

3 Related Work

In [7] and [1] it is stated that in the industrial automation domain holonic systems have undoupted advantages but the exploitation of the holonic approach in industrial practice is still very low. The autohrs claim that the decision makers are reluctant to take the risk of being the first adopters of this technology at large scale due to many negative factors, such as higher investments, anxiety over effects of emergent behavior,and lack of skilled maintenance personnel. In the PEN project this was not a problem. The goal of this project was to investigate the advantages of holonic systems in a highly decentralized energy supply system.

For the design and development of realtime software systems several methods have been developed. An overview can be found in Wikipedia [8] and [5] on page 57. Today design and development methods for real-time systems are object-oriented combined with component based concepts [5] and concurrent tasks. In COMET/RTE developed by Gomaa [5] UML (Unified Modeling Language), SysML (Systems Modeling Language) and MARTE (Modeling and Analysis of Real-Time Embedded Systems are combined. The result is a very expresive but very complex language. For the development of a management system for a holar energy network it was to complex for non IT people. Because several people of the team are familiar with the subject-oriented approach and it shows up that non IT people have no difficulties to use it we decided to describe the software in a subject-oriented way. Details about subject orientation can be found in [9], [10].
In order to make it easy for non IT-people to describe their requirement we introduced subject-oriented user stories. This type of user stories are influenced by the book "Writing Effective Use Cases" [11].

4 Modelling of Holons

4.1 Architecture of Holons and Use Case

Figure 1 shows the architecture of the Holon System developed in the PEN project. It shows the the building blocks of a holar energy system. The major functional components are the system management (Systemmanagement), the holon creation (Holonbildung) and the flexibility management (Flexibilitätsmanagement). These functional components are supported by the forecast function (Prognostik), sensor system (Sensorik) and the actuator system (Aktorik). Each

holon consists of these management components. All the other components exist only once for the whole energy network. The identifcation of these building blocks was based on the experiences of the energy engineers.
The sensor management system observes the status of the energy network e.g. it delivers information about voltage range deviations. The holon coordinator uses this information to initiate actions in order to keep the energy balance in its holon e.g. it can try to find a holon which can produce more energy as used by its consumers. These two holons can merge to a new holon which is in balance. The corresponding switching is initiated by one or both involved holon coordinators. They send the corresponding messages to the actuator system (Aktorik). Based on that functional blocks the corresponding subjects are identified. Subjects are entities which are active (thread of controls) which execute actions and they coordinate their work by exchanging messages.

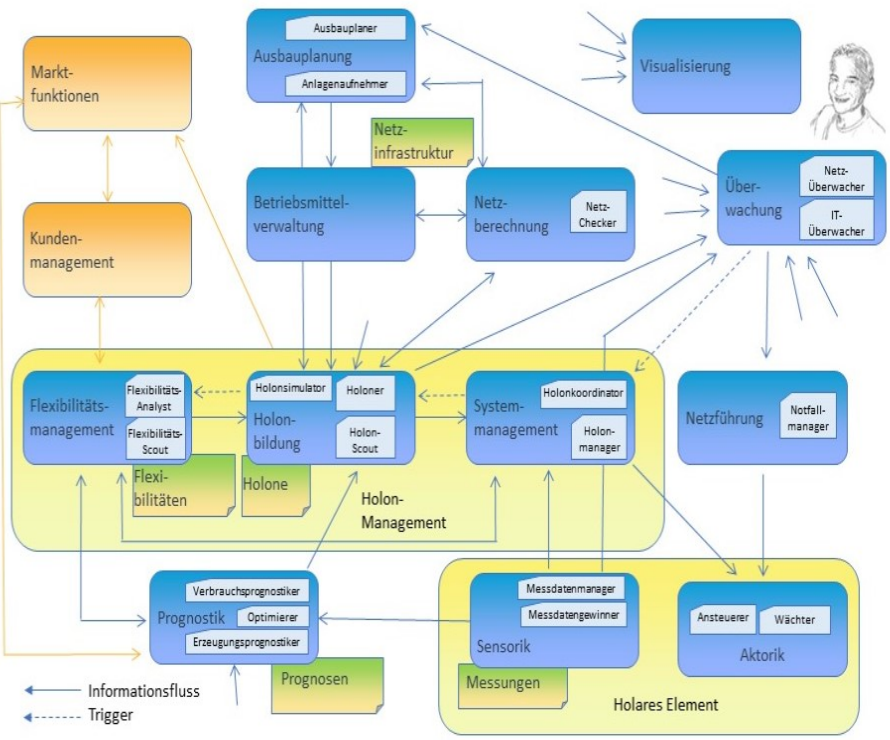

Fig. 1. Architecture of a Holar Energy System

Figure 1 outlines the identified subjects. The most left column "Name" contains the name of the iedenified subjects. Column "Typ" contains the implementation type of the subject. A subject can be either implemented in software or as a cyber physical system (Physical entities combined with IT). The subject "Holon Manager" is implemented as software service whereas the subject "Holares Element" is a "Komponente" (Cyber physical system). "Holare Elemente" are either consumer or producer of energy. The subject "Holon Koordinator" is implemented as a software service. The subject "LeitStand (Control center) is implemented as combination of human and IT.

Name	Typ (Person, Organisation, Komponente, System, Subsystem, Anwendung)	Beschreibung	Spezifika in diesem Use-Case
Holon Manager (HM)	Softwareservice	Die Rolle eines HM beschreibt eine Menge an Funktionalitäten, die durch eine oder mehrere Softwarekomponenten bereitgestellt wird	Beispiele für Use-Case-relevante Funktionalitäten sind: • Einen Holonkoordinator ermitteln • In der Rolle eines Holonkoordinators agieren
Holares Element (HE)	Komponente	Holare Elemente sind Erzeugungs- oder Verbrauchselemente, Umwandlungselemente, Speicherelement oder Leitungselemente. Ein HE besitzt immer eine Verbindung mit dem Energiesystem, eine Verbindung mit dem IKT System ist optional aber die Regel. Ein HE besitzt minimal eine (i. d. R. IT-technische) Ansteuerbarkeit, meist auch eine Auskunftsfähigkeit.	Beispiele für Use-Case-relevante Funktionalitäten: • HE können Statusinformationen senden • HE können Steuerbefehle empfangen
Holon Koordinator (HK)	Softwareservice	Holon-Kordinatoren erbringen die Basisfunktionen zum Betrieb eines Holons. Sie steuern Holare Elemente im eigenen Holon unter Nutzung der Funktionen eines Holon-Managers an.	Beispiele für Use-Case-relevante Funktionalitäten sind: • Ermittlung eines zulässigen Holons (ggf. unter Berücksichtigung weiterer Ziele). Kriterien sind noch zu definieren • Schaltbefehle für Etablierung eines Holons ermitteln • Schaltbefehle in einem Holon initiieren und deren Umsetzung überwachen.
Steuerbox	System	Eine Steuerbox kann Schaltbefehle für Holare Elemente empfangen und umsetzen.	
Leitstand (Leitstelle?)	Person/Organisation	Überwacht Netze und veranlasst Maßnahmen zur Sicherstellung einer optimalen Versorgung	Beispiele für Use-Case-relevante Funktionalitäten sind: • Detektion von Netzzuständen • Kommunikation von Netzzuständen
Netzüber-wacher (Unterschied zur Leitstelle?)	**Softwareservice/System**	**Überwacht Netze**	Beispiele für Use-Case-relevante Funktionalitäten sind: • Detektion von Netzzuständen Kommunikation von Netzzuständen

Fig. 2. Overview Subjects and their Tasks

After the identification of the acting entities in our energy management system we described a scenario which has to be relized in a first version of the management software. This use case was described in a sequence of active voice sentences. The subjects identified in figure 2 are used as subjects in these sentences. Figure 3 contains in the first column the sequence number of the action. The column "Handlung" contains the active voice sentence who does what on which object.

Based on that functional blocks the corresponding subjects are identified. Subjects are entities which are active (thread of controls) which execute actions and they coordinate their work by exchanging messages.

Ablauf im Holaren System			
Nr.	**Handlung**	**Hinweise**	**Nachfolge-aktion**
1.	Netzüberwacher stellt Spannungsabfall im Niederspannungsnetz fest	nutzt dafür Funktionen aus der Funktionsgruppe „Messen“	2
2.	Netzüberwacher meldet die Situation an Leitstelle		3
3.	Leitstelle identifiziert und lokalisiert den Fehler	nutzt dafür Funktionen aus den Funktionsgruppen „Überwachen“ und „Messen“ (Bemerkung: Was passiert wenn Leitstelle den Fehler nicht identifizieren kann ?)	4
4.	Leitstelle informiert Holon-Manager (HM) bzw. Holon-Koordinatoren im betroffenen Netzgebiet	Achtung: hier kann ein single-point-of-failure bzw. Engpass vorliegen; ist ggf. durch einen dezentralen Mechanismus zu ersetzen	5
5.	(sofern angesprochen) Holon-Koordinatoren treten mit Holon-Managern in den betroffenen Holonen in Verbindung	nutzt dafür Funktionen aus der Funktionsgruppe „Holonmanagement“	6
6.	(sofern kein Holon-Koordinator tätig wird) Holon-Manager identifizieren einen HM, der die Koordinationsaufgabe übernimmt		7
7.	Holon-Koordinator und betroffene Holon-Manager ermitteln eine plausible neue Holon-Konstellation	nutzt dafür Funktion „Holonbildung“ aus der Funktionsgruppe „Holonmanagement“	8
8.	Holon-Koordinator meldet den errechneten Vorschlag an die Leitstelle		8
9.	Holon-Koordinator geht in Zustand „alert“ und wartet auf Rückmeldung von der Leitstelle	hier ist zu prüfen, inwiefern Funktionenaus der Funktionsgruppe „Überwachen“ eingesetzt werden können; falls keine Rückmeldung seitens der Leitstelle im eingestellten Zeitfenster erfolgt, wird Use-Case „Betrieb bei ausgefallener IKT“ angestoßen	10

Fig. 3. Use case scenario in active voice sentences

4.2 Definition of Subjects/Actors

From figure 3 the behavior of the subjects are extracted. This means the sentences with the sam subjects are grouped to one story. The activities are brought into the right seqeunce and for the synchronisarion with other subjects send and receive operations are inserted at the corespondingplaces. The functionality of each subject is described separately in a table. In this table it is defined which actions are executed by the subject in which sequence. This actions can be local (changing local data) or interactions with other subjects (sending and receiving messages). One of the most important subject is the holon coordinator (Holonkoordinator) because it monitors the energy balance of the holon it is responsible for. Figure 4 shows part of the behaviour description of the subject holon coordinator (Holonkoordinator). The table has four columns.

- Name of an action (Sprungziel): Used for identifying the successor action
- Action (Aktion): Description of the action (local, send or receive)
- Comment (Kommentar): Additional explanation for an action
- Succeeding action (Nachfolgeaktion): Name of the action which will be executed next.

Each line in Figure 4 is one action and the table describes the behaavior of subject . The first line in the table defines the start state of the holon coordinator. The succeeding action is called Kontrollaktorik (see column on the most right in Figure 4. This action is defined in the second line (see first column in Figure 4). In this action the state of the holon is checked. If the result is ok the next action is endehauptfunktion otherwise the succeeding action badanHolonManager. With this type of tables the dynamic of the holon management is described in a first draft.

4.3 Subject-Oriented Process Models

Based on the function blocks shown in Figure 1 and on the description of these blocks in behaviour tables (see figure 4) we derived a formal subject oriented specification. Figure 6 shows the essential subjects and the messages they exchange (Subject Interaction Diagram, SID). These 5 subjects are mainly responsible for managing the energy balance of a Holon.

For each subject a behaviour diagram is created (Subject Behaviour Diagram, SBD). In this diagram the behaviour is precisely defined. Figure 5 shows the first steps in the behaviour diagram of the Holon Coordinator.

For the specification of Holar energy system management we used the complete description possibilities of S-BPM. We used multi-processes for specifying the holons. A holar energy system consists of many holons. The management process of each holon is one of the multi-processes. In each subject behaviour there are action sequences which are executed at different places in the behaviour these sequences are collected in macros. In order to create the SIDs and SBDs a S-BPM editor is created based on the y-ED graphic platform. For validation purposes the models created with that editor could be directly executed with a ASM (Abstract State Model) based tool called CoreASM [12]

Sprungziel	*Aktion*	*Kommentar*	*nachfolgende Aktion*
(Haupt-) Funktion: Objektmanagement			
beginn_hauptfunktion	sende Steuerbefehle an zugeordnete Aktorik	Ggf. ist hier eine „Übersetzungsaufgabe“ notwendig, wenn eine Zielgröße (z.B. Reduktion Leistung) vom Objekt gefordert wird und dieses unterschiedliche Wege zur Zielerreichung hat.	Kontroll_aktorik
Kontroll_aktorik	Überprüfe ob erfasster Zustand bei Netzanschluss (Summe) plausibel für Objekt	Erfragt IST Werte bei seinen HE; bei Aktorik kann ein einzelner Wert auch als Bestätigung zurückgesendet werden; Es wird eine Analysekompetenz vermutet, die es dem HM ermöglicht, Fehler auch zu lokalisieren	Ja, plausibel gemäß Zielvorgaben: weiter bei: ende-hauptfunktion nein: weiter bei bad_an_holonkoordinator
bad_an_holonkoordinator	Meldung Abweichung Zielvorgabe an (akutellen) Holon-Koordinator		Weiter bei: Prüfung HE einzeln
Prüfung HE einzeln	HM hat bei HE (Anlagen) das Recht, ein Fehlerprotokoll abzufragen	Da eine Auswertung eines Fehlerprotokols zu komplex ist, wird der Bericht (oder nur die Abweichung?) 1:1 weitergeleitet. Datenformate ??	Fehlerprotokoll_senden
Fehlerprotokoll_senden	An vordefinierte Empfänger in Abhängigkeit vom Standort und Besitz (z.B.		Ende_hauptfunktion

Fig. 4. Behaviour specification of subject "Holon Koordinator" in natural language

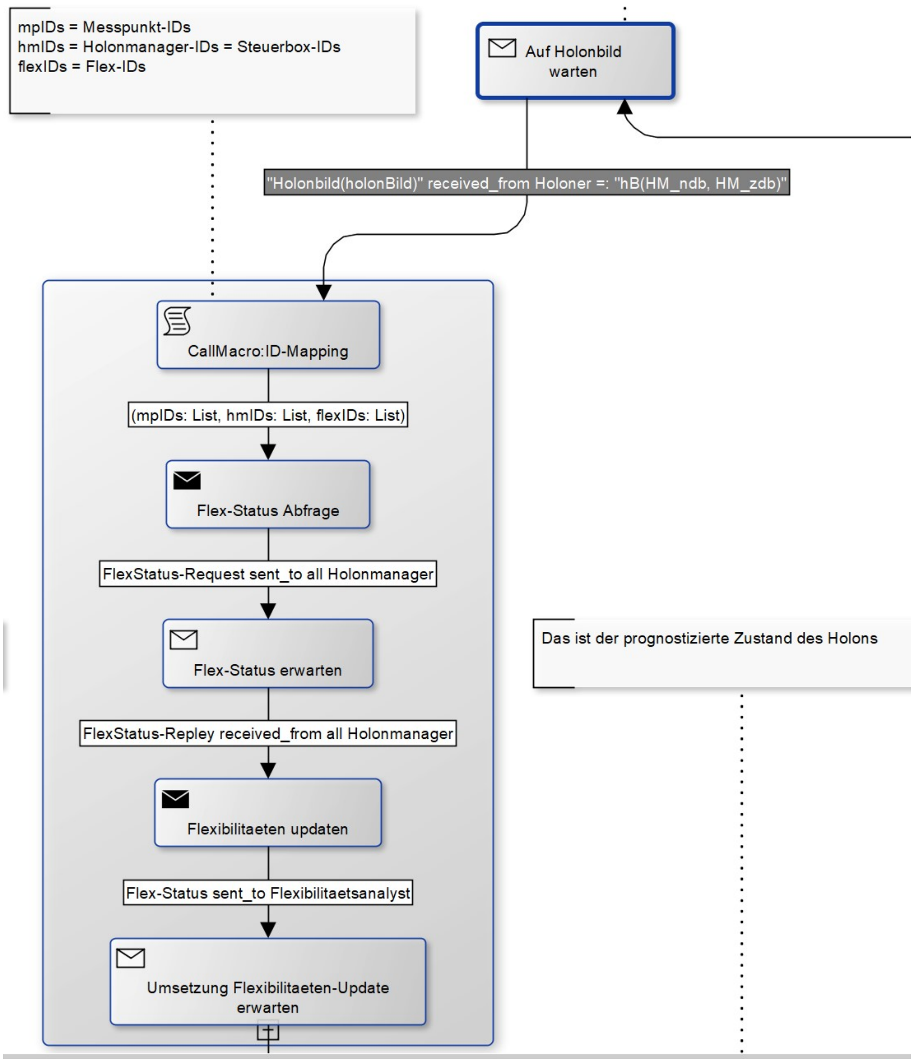

Fig. 5. Subject Behaviour of the Holon Coordinator

4.4 Voltage Range Deviation S-BPM Process

Based on the complete specification we decided to implement the management system in a iterative way. We started with the use case Voltage Range Deviation. The voltage of the network is not any more in the defined range. It is either higher or lower as required. The voltage is measured regularly by the sensor system. The result of the measurement is sent to the holon coordinator via the subject Netzueberwacher. The Holon Koordinator tries to find a solution in cooperation with the subject Holoner and MP-Simulator. The MP-Simulator makes forecasts about the voltage trends and the Holoner evaluates whether the proposed solution can be realised with the available network infrastructure. If the proposed solution can be realised the Holon-Koordinator sends the corresponding switching commands to the subject Ansteuerer via the the subject Holoner. Figure 6 shows the SID for the use case Voltage Range Deviation.

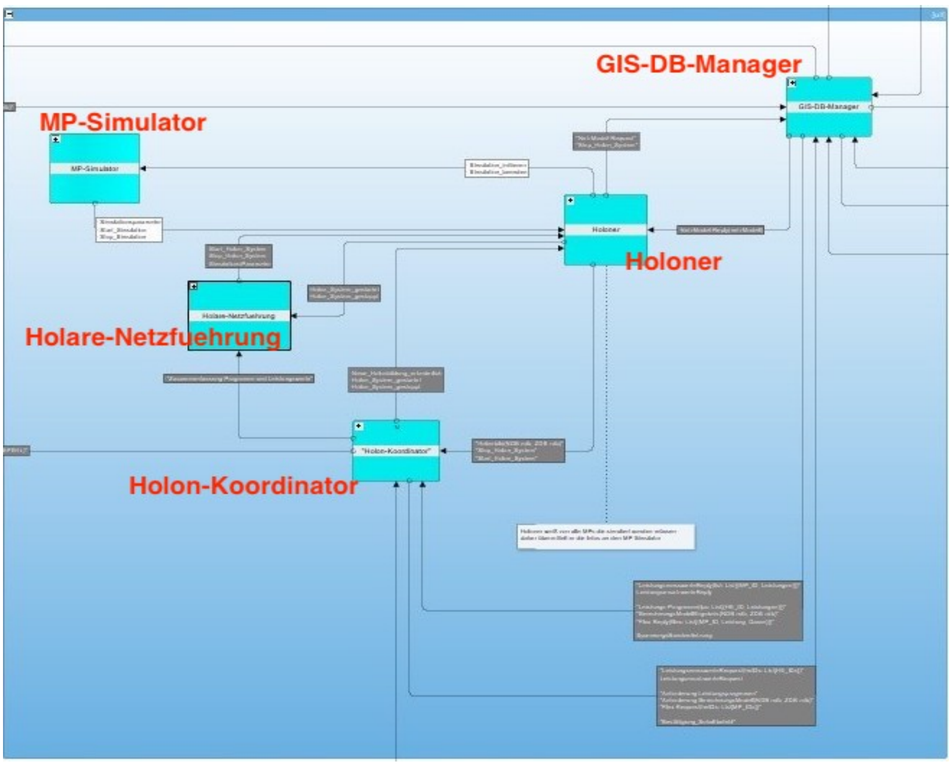

Fig. 6. Subject Interaction Diagramm for Voltage Range Deviation

5 Model-City Application

In order to evaluate the software solution in detail a physical model has been built. Figure 7 shows the physical model which is used for testing a first version of the management system for handling voltage range deviation. The logical network structure is shown in Figure 8. This is the initial state for the tests.

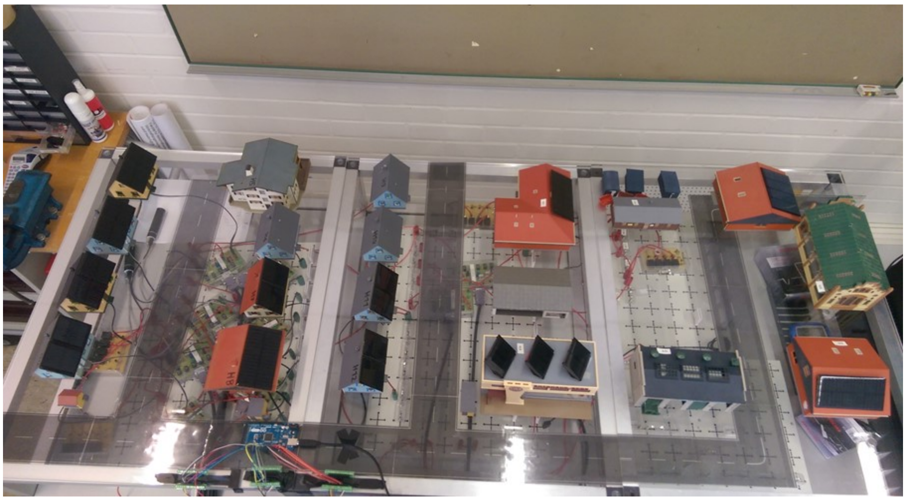

Fig. 7. Physical model: Modellstadt

In the initial state in our Modellstadt there are two Holons Holon 1 and Holon 2.

Beginning from this initial state the scenario shown in Figure 9 is executed. Essentially Figure 9 is only a precise definition of the use case shown in figure 3. The execution of this scenario is based on the five subjects doing the Holon Management (see 6).

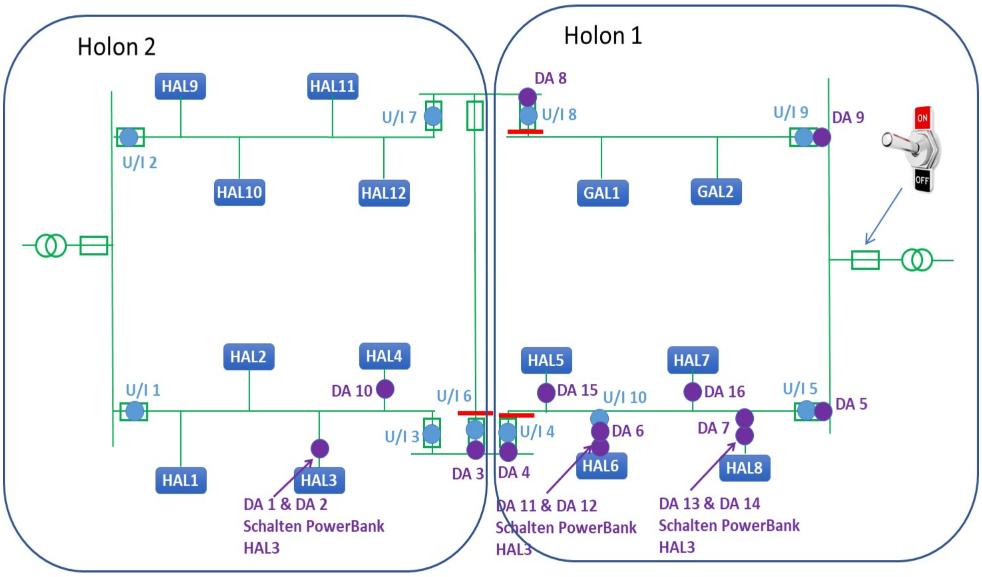

Fig. 8. Initial power supply situation

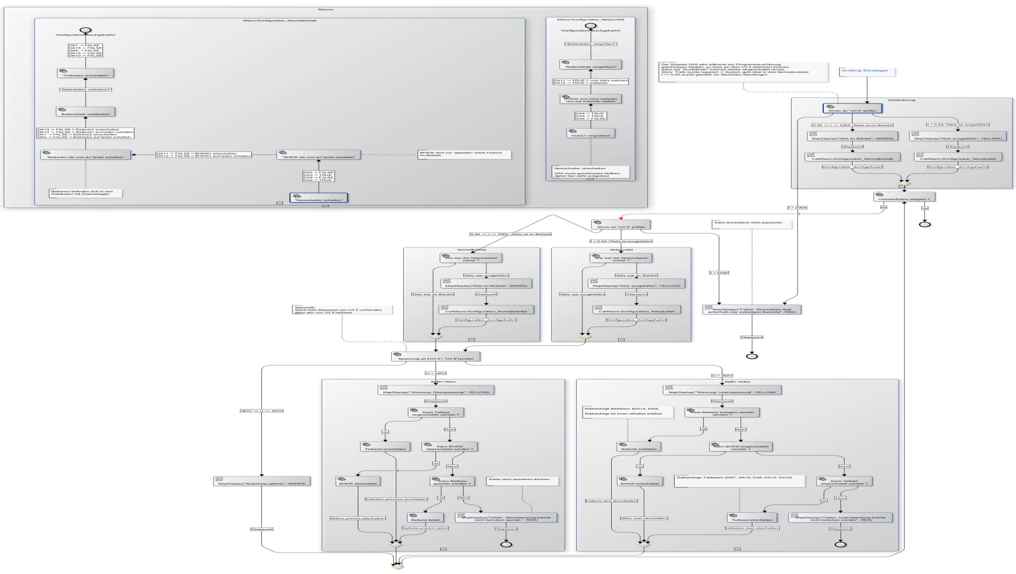

Fig. 9. Demo scenario Modellstadt

In order to simulate a problem with the voltage range a switch is built into the Modellstadt. If this switch is turned off a voltage range deviation happens.

Turning off the switches simulates the break down of a transformer (see Figure 10). The break down of the transformer implies that right part of the network is without energy (see 11). This means Holon 1 has not enough energy.
Now the Holon Coordinator starts its activities to solve this problem. It starts a cooperation with Holon 2. Both Holons decide to merge with each other to built a new Holon. The switches are brought into positions for connecting the two Holons. The result is a new Holon (see figure 12). When the transformer switch is turned on again, the system considers the new state as "transformer is repaired" and evaluates whether a split of the present Holon would be possible. If both initial holons can produce the energy which is consumed the switches are brought into the initial positions. Now we have again two independent holons.

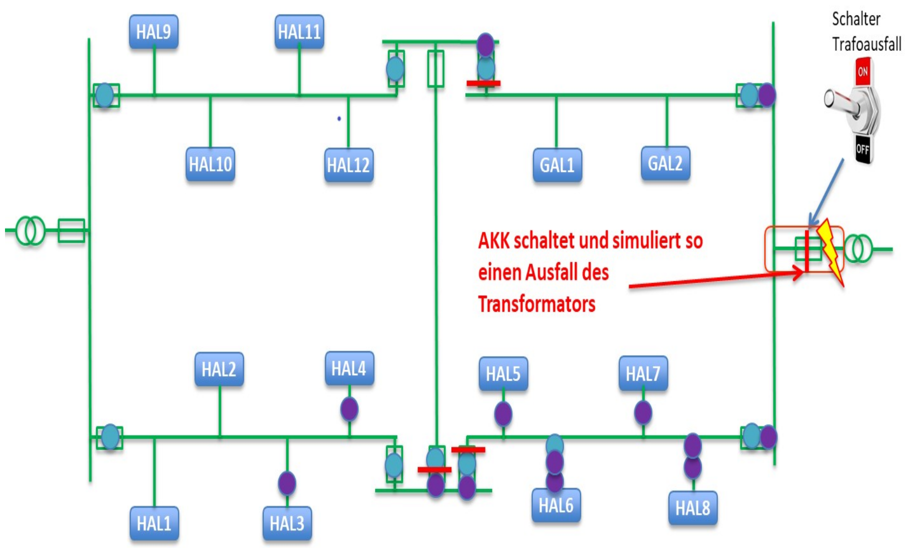

Fig. 10. Break down of a transformer

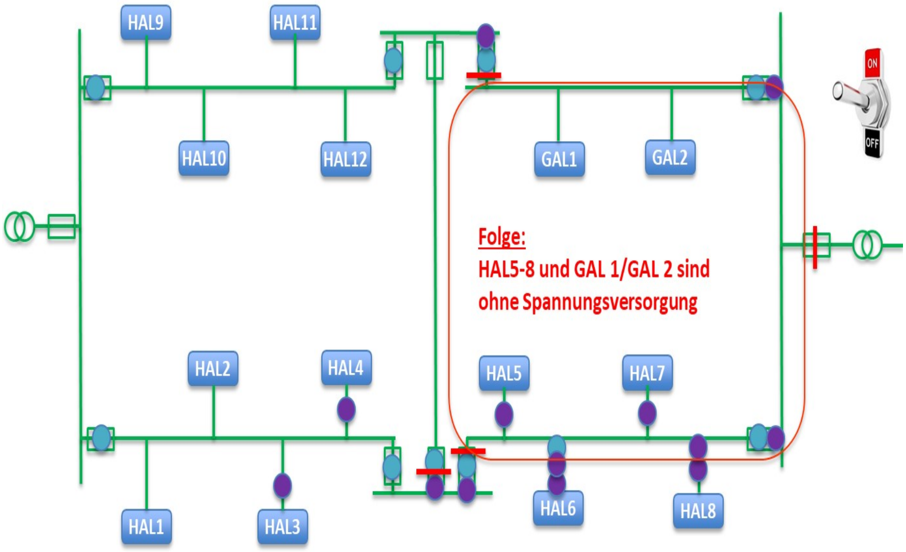

Fig. 11. Supply of energy after break down of the transformer

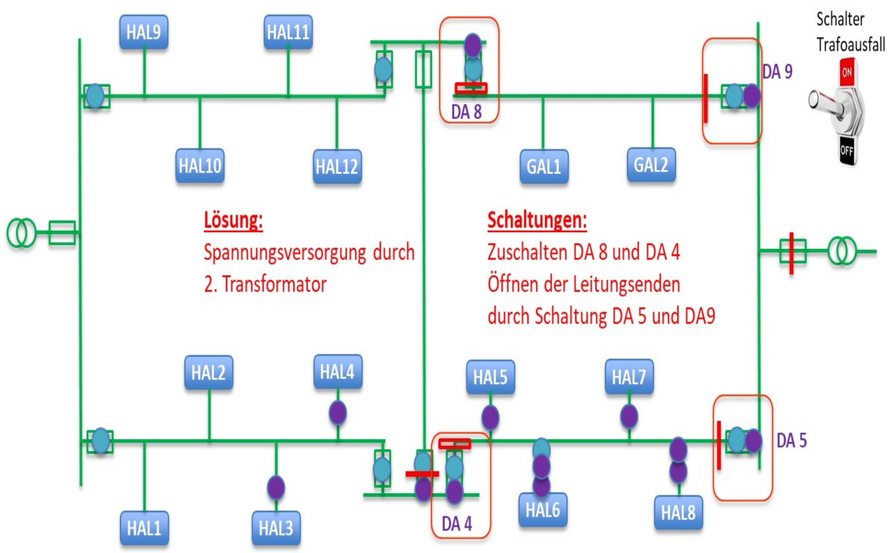

Fig. 12. Reconfiguration of the energy network

6 Summary and Conclusion

In this paper we have shown how the subject oriented concept is applied for specifying and implementing a complex real-time problem. Subject oriented methods have been used for collecting the requirements for the management software system of self organising smart energy grids. It shows that the subject oriented approach supports the communication between It-people, electrical engineers and business people. Subject oriented modeling is close to holonic systems and supports the development of such system.
It also shows up that the tool support is not sufficient. It would be helpful if the validation of subject oriented models is more user friendly. Therefore this tool could not be used in a helpful way.

References

1. Bussmann, S., Sieverding, J.: Holonic control of an engine assembly plant an industrial evaluation
2. Leitão, P., FranciscoRestivo: Adacor: A holonic architecture for agile and adaptive manufacturing control. Computers in Industry **57**(2) (February 2006)
3. Silva, R., Arakaki, J., Junqueira, F., Filho, D.S., Miyagi, P.: Modeling of active holonic control systems for intelligent buildings. Automation in Construction (25) (August 2012)
4. Koestler, A.: Janus, A Summing Up. Random House, London (1978)
5. Gomaa, H.: Real-Time Software Design for Embedded Systems. Cambridge University Press, New York, NY (2016)
6. Shin, K.G.; Ramanathan, P. Proceedings of the IEEE. In: Real-time computing: a new discipline of computer science and engineering. IEEE (1994)
7. Vrba, P., Tichy, P., Marik, V., Hall, K., Staron, R., Maturana, F., Kadera, P.: Rockwell automation's holonic and multiagent control systems compendium. IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS—PART C: APPLICATIONS AND REVIEWS **41**(1) (January 2011)
8. Wikipedia. Wikipedia. In: Real-time computing. (last access Nov. 2017)
9. Fleischmann, A.: Distributed Systems - Software Design and Implementation. Springer Verlag, Berlin (1994)
10. Fleischmann, A., Stary, W.S.C., Obermeier, S., Börger, E.: Subject-Oriented Business Process Management. Springer Verlag, Berlin (2012)
11. Cockburn, A.: Writing Effective Use Cases. Addisson Wesely, Indianapolis (2001)
12. Farahbod, R., Gläser, U.: The coreasm modeling framework. Software: Practice and Experience **41**(2) (February 2011)