

Case Study Definition and Implementation

Arun Babu¹, Francesco Brancati¹(✉), Sorin Iacob², David Mobach²,
Marco Mori³, and Thomas Quillinan²

¹ Resiltech SRL, Pisa, Italy

{arun.babu, francesco.brancati}@resiltech.com

² Thales Netherlands B.V., Hengelo, Netherlands

{sorin.iacob, david.mobach,

thomas.quillinan}@nl.thalesgroup.com

³ Department of Mathematics and Informatics, University of Florence,
Florence, Italy

marco.mori@unifi.it

1 Introduction

In this chapter we present three case studies in the smart grid domain: Electrical Vehicle charging, Household Management, and an integrated case study that combines the first two together with ancillary services. These case studies are first modelled using the AMADEOS Architectural Framework (AF) and associated tooling. We utilise the four levels of the AMADEOS AF: *mission*, *conceptual*, *logical* and *implementation*, as well as the seven viewpoints that have been defined: *Structure*, *Dynamicity*, *Evolution*, *Dependability and security*, *Time*, *Multi-criticality* and *Emergence*. We therefore examine the entire lifecycle of the framework considering some real-world case studies. These case studies are based on experts' feedback, including AMADEOS Advisory Board members, to ensure that realistic architectures are designed.

The architectures developed in this chapter will be further instantiated in a simulation environment, using the simulation tooling developed in the AMADEOS project. With these instances several experiments will be run in order to validate the framework as well as the architectures that were defined.

The three Smart Grid-based case studies described in this chapter are used to prove the effectiveness and consistency of the AMADEOS architectural framework. The method provided by the framework allows to design and implement a generic SoS in a procedural and systematic way. To accomplish this, the AMADEOS project defines a pyramidal top-down approach that must be undertaken passing through four different levels: a mission for the SoS, the conceptual level, where the ideas and concepts of the SoS are defined in order to support the capabilities of the SoS. Next, the logical level where the SoS is designed and these concepts are adapted towards supporting the requirements of the individual SoS domain. Finally, these are actualised in the implementation level, where the design is contextualized and realized in the enterprise.

This work has been partially supported by the FP7-610535-AMADEOS project.

© The Author(s) 2016

A. Bondavalli et al. (Eds.): Cyber-Physical Systems of Systems, LNCS 10099, pp. 207–238, 2016.

DOI: 10.1007/978-3-319-47590-5_8

2 Smart Grid SoS

2.1 Smart Grid SoS Model

Figure 1 shows the model of the Smart Grid SoS created using the supporting facility tool. The Smart Grid SoS is composed of: EV_Charging, Medium_Voltage_Control, and the Household CS.

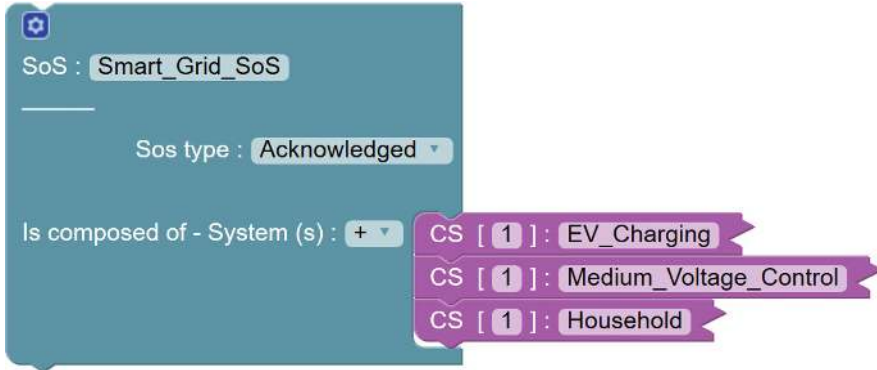


Fig. 1. The Blockly Smart Grid SoS model

2.2 EV-Charging SoS Case Study

Mission. The EV SoS must be designed to provide a friendly and convenient service to the users and at the same time, profitable to the provider. Planning and scheduling is of paramount importance for both energy providers and users: as an example, on one side, if the charging requests are spread during the day, there will be limited and/or controlled load peaks on the grid to be handled, thus the energy price may not vary abruptly over time and prioritized consumers (e.g., police and fire-fighter vehicles, ambulances, etc.) will be easily handled by the charging station operators. On the other hand, knowing the energy prices and available time slots, the users will be able to carefully plan the recharging operation while keeping the service affordable.

A typical scenario would be as follows: EVs travel through a wide area, where several charging station operators provide recharging services, by means of charging points. Drivers in need of power for their EV can provide the expected charging context (duration, power, etc.) to the e-mobility service in order to receive information regarding recharging time slots and associated energy prices of each charging station operator. A load management optimizer that cooperates with the charging station operators carries out planning and scheduling activities. The interested driver will then choose one of the slot-price pair possibility for recharging its vehicle, will be allowed to plug-in its EV at the charging point of the chosen charging station operator during the reserved time slot only, and the amount due will be based on the energy

consumption times the booked price. At the end of the recharging operation, the driver receives a billing invoice.

It is evident from the above-mentioned scenario that various dependability aspects need to be considered. The participant systems need to be time synchronized to provide consistent information for scheduling and planning purposes. Furthermore, critical EVs should be prioritized for recharging with respect to any other vehicle. Therefore, each EV should be assigned a priority level. Moreover, the e-mobility service should be accessible by registered users only, i.e. the owner of an EV, to reduce the possibility of denial of service attacks being performed by illicit malevolent users scheduling recharging reservations, without a real need.

Conceptual Level. In this section we report the result of the activities performed at the conceptual level for the EV charging case study. For each viewpoint we list the most representative identified SoS requirements defined taking as input the SoS meta-requirements [1]. Traceability of the full set of requirements on meta-requirements is also provided in [4].

Architecture Viewpoint. The EVC consists of a subset of the CSs of the SoS “*Electrical Vehicle Charging in Smart Grids*” described in Chapter 2.1 of [4]. In particular (Table 1):

High level representation of EV-Component interactions.

A pictorial view of the EV charging SoS is reported in Fig. 2.

The steps required to recharge an EV are described below. Each step number corresponds to a sequence of actions that are carried out.

Table 1. EV charging case study components description

Name	Description
EV	An electrical vehicle.
Driver	The driver of the EV.
Charging Point (CP)	A physical connection for recharging the EV.
Smart Meter (SM)	A smart meter providing production and consumption values. May also enable advanced sensor facilities providing active/reactive production, frequency monitoring, voltage monitoring, etc.
Charging Station Operator (CSO)	Operator of a charging station, which is an electrified parking lot with several CPs, represented by an independent enterprise or owned by energy provider.
Load Management Optimizer (LMO)	The main software component connected to the grid which is in charge of providing power constraints and energy set points to the CSO.
E-mobility service	<ul style="list-style-type: none"> - Lists the best charging station locations to EVs, - Handles reservations. - Receives availability updates from CSOs.

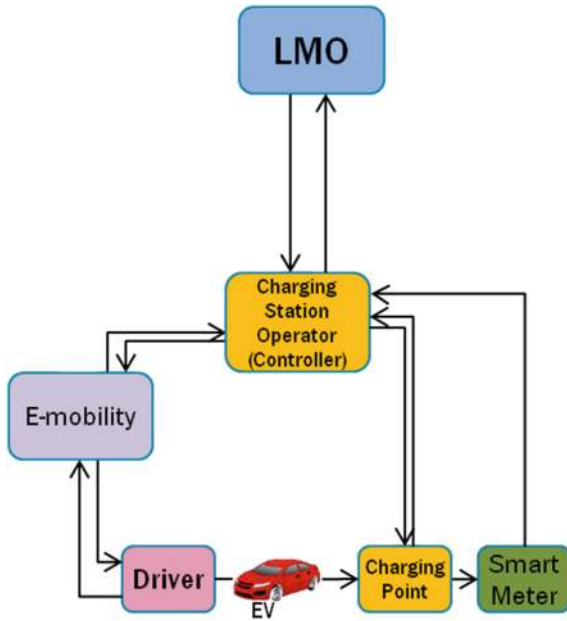


Fig. 2. Electrical vehicle charging SoS architecture

1. The information flow starts when a driver needs to recharge their EV. The driver requests a charging opportunity from the E-mobility service by providing the expected charging context (duration, power, etc.);
2. The E-mobility replies with the charging opportunity availabilities by accounting all CSOs present in the SoS;
3. Once the charging opportunities are received, the Driver asks for a reservation towards the most comfortable CSO according to his needs (e.g. availability of energy, distance, etc.) and the E-mobility forwards the message to the correspondent CSO;
4. The CSO updates its schedule, considering the received request and the power constraints defined by the set points provided by the LMO. The CSO reserves the desired time slot, allocates the resources and sends to the E-mobility service an acknowledgement;
5. The E-mobility service forwards the acknowledgment to the driver;
6. The driver reaches the CSO within the booked time slot. The EV is plugged in to the Charging Point (CP) and the CSO is notified about this event;
7. The CSO decides whether to allow the charging operation or to deny it proposing a re-scheduling of the time slot. This can happen if the situation has changed between the time the driver has booked their slot and they arrive at the CP. For example, if a higher priority EV has requested a slot;
8. While recharging, the Smart Meter measures the energy consumption and the resultant EV load for, respectively, billing and smart grid power flexibility purposes. Such information are sent to the CSO;

9. At the end of the charging cycle the EV is unplugged,
10. The CSO sends the billing invoice to the driver through the E-mobility service.

Dependability and Security Viewpoint. It is of paramount importance to guarantee the highest achievable availability and reliability of the SoS so that the charging requests are readily and continuously available. Achieving these properties allow not only to reach the highest quality of service, but also to maximise the profits. The SoS allows drivers to enquire for a charging opportunity through the E-mobility service and to have access to the CPs at any time. Thus, energy grid enterprises can maintain the grid stability, balance the load, and ensure correct voltage levels and frequency, etc., while energy providers can make the highest profits out of it. To avoid undermining such premises, Denial of Service (DoS) attacks caused by unauthorized users enquiring the E-mobility service or trying to access CPs need to be properly tackled.

Dynamicity Viewpoint. EVs join and leave the SoS, according to the need for charging of the EV. The change in the topology due to this turnover of EVs characterizes the dynamicity (DYN) of the SoS.

Emergence Viewpoint. According to [2], Emergence is: *A phenomenon of a whole at the macro-level is emergent if and only if it is new with respect to the non-relational phenomena of any of its proper parts at the micro level.* This indicates that these phenomena cannot be observed at CS level, but at SoS level (or other higher level). As consequence of identification of emergence scenarios, hazard analysis must be performed to identify and mitigate any hazards.

Evolution Viewpoint. Due to technological advances, marketing, or customer needs; different kind of EVs it may be necessary to improve, change or add services.

Multicriticality Viewpoint. The scenario described so far, only foresees normal EVs (e.g., private EVs) that need to recharge and ask for charging opportunity. However, during real-life emergencies, e.g. rescue operations, wildfires and public security, the requirement is that specific EVs must always be available. Therefore, the CSO scheduling strategies must prioritize such vehicles before any others. Emergency EV drivers access prioritized charging in the same fashion of any other driver, i.e. through the E-mobility service.

Time Viewpoint. As a general remark, it is worth noting that most of the information exchanges between CSs rely upon a notion of time. As an example, it would be impossible to plan and schedule a request of recharging an EV by a driver for the current day or even pay the billing invoice if not properly time-stamped. Thus, to provide the payment services, to allow users to enquire for charging opportunities and to effectively, and efficiently, plan, and schedule, recharging operations over time, there must be awareness of time over the SoS. Further, each CS must be time synchronized to a common reference time to successfully provide their services.

Logical Level. This section describes the SoS Logical Description of the SoS defined in [4] using the model made using the supporting facility tool.

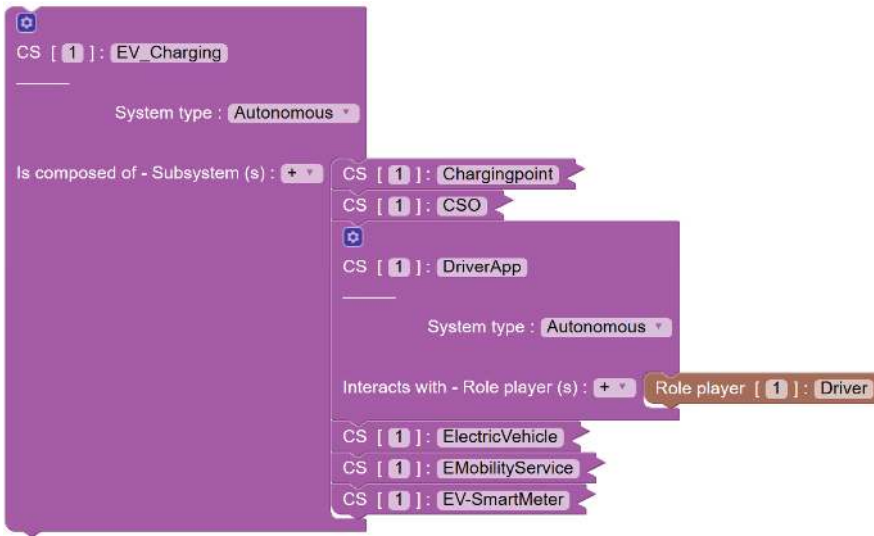


Fig. 3. EV Charging in Blockly

After loading the model in supporting facility and double clicking on EV_Charging CS we can see in Fig. 3 that the EV-Charging Blockly model consist of CSs that matches to the diagram depicted in Fig. 2.

Figures 3 and 4 show that EV-Charging CS consists of the following CSs: (i) Chargingpoint, (ii) CSO, (iv) DriverApp, (v) ElectricVehicle, (vi) EMobilityService, and (vii) EV-SmartMeter

In the following section will be described in details the CSs listed above, through the expansion of the blocks.

Charging Point CS. The Chargingpoint CS can be expanded by double clicking on it and the result is depicted in Fig. 4.

Figure 4 shows that the Chargingpoint CS communicates with the Electric Vehicle, CSO, EV-SmartMeter CSs by the CharginPoint-ElectricVehicle RUPI, Chargingpoint-CSO RUPI and Chargingpoint-EV-SmartMeter RUPI, respectively. Figure 4 also shows that the services provided by the Chargingpoint CS. Furthermore, it has a State Variable Chargingpoint: charging_done.

From the viewpoint of communication, double clicking on RUPIs and RUMI blocks of Charging Point, CS Fig. 4 shows that the Chargingpoint CS provides the services to CSO CS through the Chargingpoint-CSO RUMI. The Fig. 4 shows also that:

- The Chargingpoint-ElectricVehicle RUPI transports the Plug-OUT-Signal and it is connected to the ElectricVehicle-ChargingPoint RUPI of the ElectricVehicle CS,
- The Charging Point-EV-Smart Meter RUPI transports Electricity and it is monitored through the Charging Point Probe.

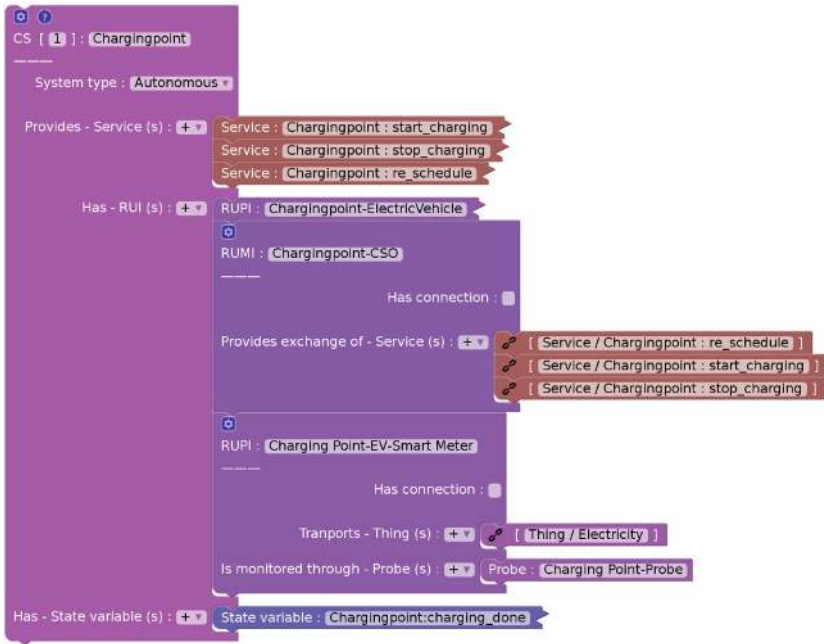


Fig. 4. The Blockly Charging point CS model

CSO CS. On expanding the CSO CS block (Fig. 5), its services, RUIs, MAPE, and state variables can be seen.

Figure 5 shows that the Chargingpoint CS communicates with the EmobilityService, Chargingpoint, LMO, Aggregator CSs by the CSO-EMobilityService, CSO-Chargingpoint, CSO-LMO, CSO-Aggregator RUMIs respectively. The Fig. 5 also shows the services provided by the CSO CS. It also has a State variable CSO: reservation.

From the viewpoint of communication, double clicking on RUMIs blocks of CSO CS, Fig. 5 shows that the CSO CS provides the following services:

- CSO:do_charging_reservation, CSO:do_priority_charging_reservation to the EmobilityService CS through the CSO-EMobilityService RUMI,
- CSO: EV Charging Schedule and CSO: Update energy consumption to the LMO CS through the CSO-LMO RUMI
- CSO: Set Energy price and CSO: Forward energy price to the Aggregator CS through the CSO-Aggregator RUMI

Figure 5 also shows that the CSO-Chargingpoint RUMI is connected to the Chargingpoint-CSO RUMI in order to call the service provided by ChargingPoint CS; the CSO-LMO RUMI is monitored through the Probe CSO Probe. The CSO has a local clock that we call CSO-clock and that will be described when addressing the Time viewpoint.

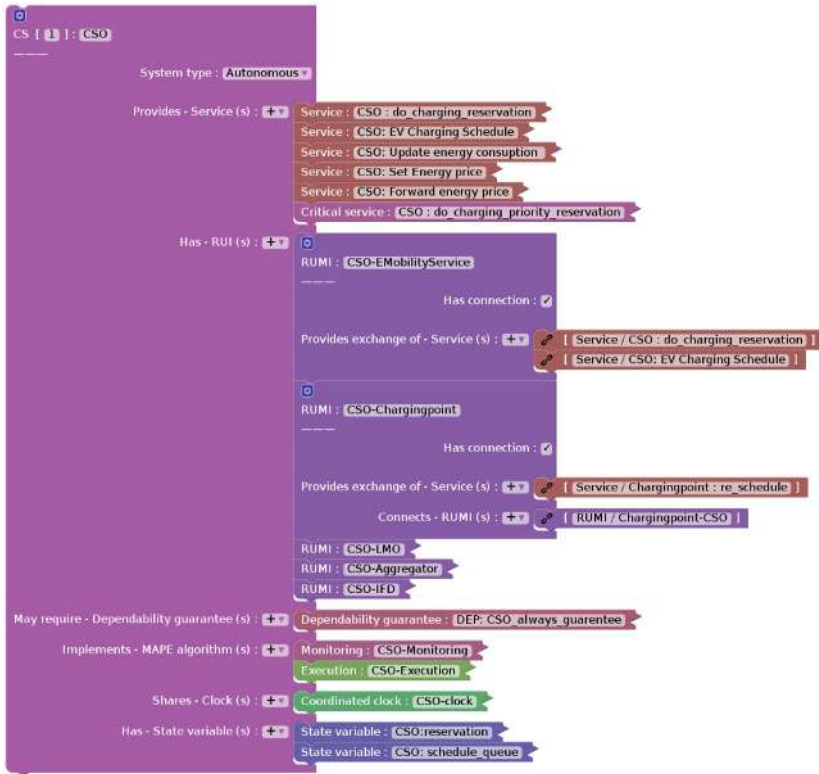


Fig. 5. The CSO CS Blockly model

DriverApp CS. On expanding the DriverAPP CS block in Fig. 6, it shows that the DriverApp CS communicates with EMobilityService CS by the DriverApp-EMobilityService RUMI, and shows that provides the service DriverApp: accept_reservation.

This figure also shows that DriverApp CS:

- Interacts with a Role player: Driver,
- Has the following State variables:
 - DriverApp:result_of_charging_opportunities_request;
 - DriverApp: selected_charging_opportunity;
 - DriverApp:duration;
 - DriverApp:power,
 - DriverApp:got_reservation.

From the viewpoint of communication, double clicking on DriverApp-EMobilityService RUMI, we see that the DriverApp CS provides the service DriverApp:accept_reservation by DriverApp-EMobilityService RUMI to EMobility CS, and

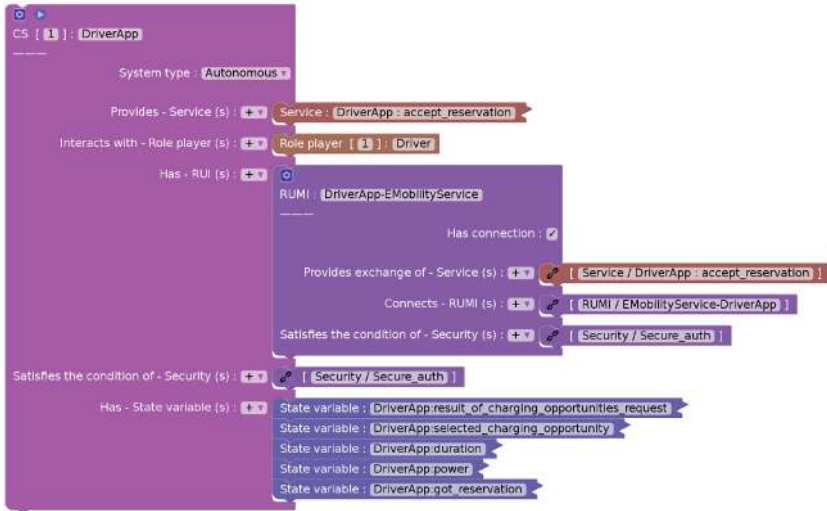


Fig. 6. The DriverAPP CS Blockly model

this RUI is connected to the EMobilityService-DriverAPP RUI in order to call the services provides by the EMobilityService.

ElectricVehicle CS. On expanding the ElectricVehicle block as depicted in Fig. 7, it is shown that the ElectricVehicle CS has ElectricVehicle-Chargingpoint RUI in order to connect with the Charging Point CS. Indeed, expanding the ElectricVehicle-Chargingpoint RUI you can see that the RUI transport Plug-In-Signal and it is connected with Chargingpoint-ElectricVehicle RUI.

EMobilityService CS. On expanding the EMobilityService CS block as depicted in Fig. 8, it is shown that the EMobilityService CS communicates with the DriverAPP, Aggregator, Market, CSO CSs by the EMobilityService-DriverAPP, EMobilityService-Aggregator, EMobilityService-Market, EMobilityService-CSO RUIs respectively.

Figure 8 shows also that EMobilityService CS provides the services:

- EMobilityService: Set Energy price
- EMobilityService: Forward Energy price
- EMobilityService: do_charging_reservation
- EMobilityService: get_available_charging_opportunities
- EMobilityService: re_schedule

and has the State variables:

- EMobilityService:available_charging_opportunities
- EMobilityService:charging_op_sent_by_driver
- EMobilityService:reservation_to_be_sent_to_driver

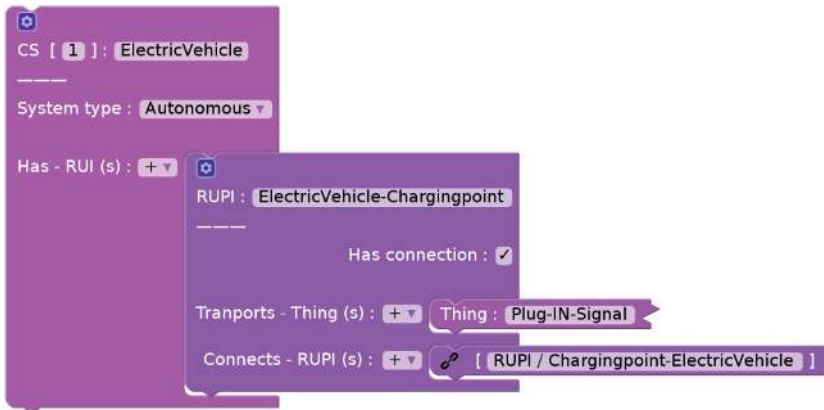


Fig. 7. The EV CS Blockly model

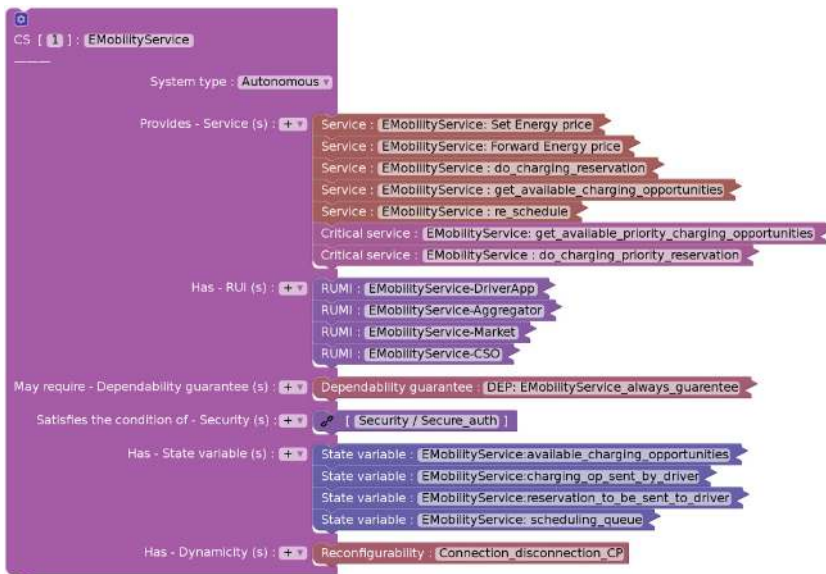


Fig. 8. The EMobility CS Blockly model

From the viewpoints of communication, double clicking on RUMIs blocks of EMobilityServices CS, the Fig. 9 shows that the EMobilityService CS provides the services:

- EMobilityService:do_charging_reservation, EMobilityService: get_available_charging_opportunities, EMobilityService: re_schedule through the EMobility Service-DriverApp RUMI



Fig. 9. The EMobility CS RUMI model

- EMobilityService: Set Energy price, EMobilityService: Forward Energy price through the EMobilityService-Aggregator, that is connected to the Aggregator-EMobilityService RUMI
- EMobilityService: Set Energy price through the RUMI EMobilityService-Market.

Furthermore, the EMobilityService-CSO is connected to the CSO-EMobilityService RUMI in order to call the services provided by CSO CS. The Fig. 9 shows also that EMobilityService has the State Variable:

- EMobilityService:available_charging_opportunities
- EMobilityService:charging_op_sent_by_driver
- EMobilityService:reservation_to_be_sent_to_driver.

EV-Smart Meter CS. On expanding the EV-SmartMeter CS block as depicted in Fig. 10, it shown that the EV-SmartMeter CS communicates with Chargingpoint and Meter Aggregator CSs by the EV-Smart Meter-Chargingpoint RUMI and EV-Smart Meter-Meter Aggregator RUMI respectively.

Figure 10 shows also that the EV-SmartMeter CS provides the service EV-Smart Meter: Get_energy_consumption and implements MAPE Algorithm.

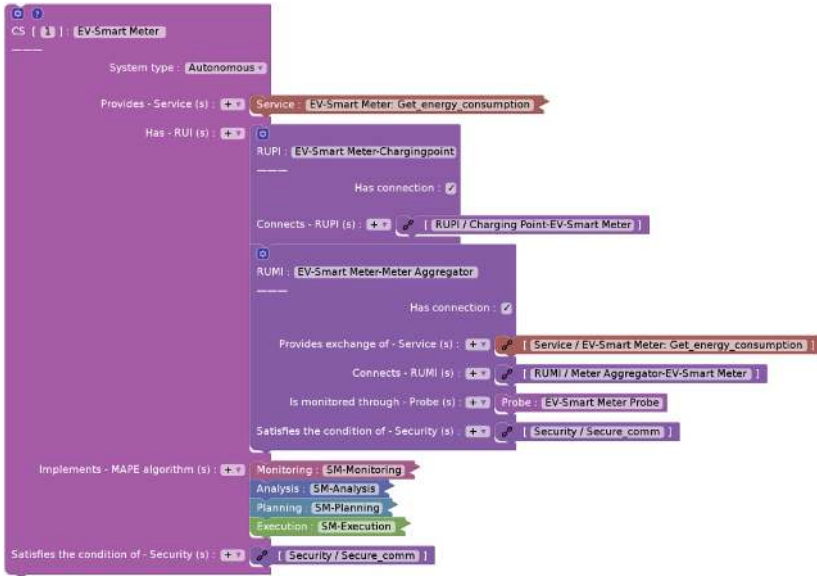


Fig. 10. The EV-smart meter CS Blockly model

From the viewpoint of communication, double clicking on RUMI and RUI blocks of EV-Smart Meter CS, Fig. 10 shows that the EV-SmartMeter CS provides the service EV-Smart Meter: Get_energy_consumption through the EV-Smart Meter-Meter AggregatorRUMI, and the EV-Smart Meter-Chargingpoint RUI is connected to the Charging Point-EV-SmartMeter RUI of the Chargingpoint CS.

Implementation. As stated in D4.2 [4], the EV SoS must be designed to provide a friendly and convenient service to the users and, at the same time, profitable to the provider. Planning and scheduling is of paramount importance for both energy providers and users: as an example, on one side, if the charging requests are spread during the day, there will be limited and/or controlled load peaks on the grid to be handled, thus the energy price may not vary abruptly over time and prioritized consumers (e.g., police and fire-fighter vehicles, ambulances, etc.) will be easily handled by the charging station operators. On the other hand, knowing the energy prices and available time slots, the users will be able to carefully plan the recharging operation while keeping the service affordable.

A typical scenario would be as follows: EVs travel through a wide area, where several charging station operators provide recharging services, by means of charging points. Drivers in need of power for their EV can provide the expected charging context (duration, power, etc.) to the e-mobility service in order to receive information regarding recharging time slots and associated energy prices of each charging station operator. A load management optimizer that cooperates with the charging station operators carries out planning and scheduling activities. The interested driver will then choose one of the slot-price pair possibility for recharging its vehicle, will be allowed

to plug-in its EV at the charging point of the chosen charging station operator during the reserved time slot only, and the amount due will be based on the energy consumption times the booked price. At the end of the recharging operation, the driver receives a billing invoice.

With this scenario in mind, and using the AMADEOS tooling, we designed a SoS (shown in Fig. 11, and described in D4.1 [3]) based on a typical EV rollout, in particular based on the desired situation in the Netherlands, based on interviews and workshops with experts, both in the EAB and Grid operators. This SoS was modelled using the Blockly tool (this is described in D4.2 [4]) and a simulator was generated from the tool. We combined this simulator with a simulation toolkit called SimPy and performed a number of experiments based on the scenarios defined in D4.2. This served to both validate the simulator as well as determine if such a simulation could be used to determine and validate possible future designs of the EV charging network, based on varying user and SoS behaviour.

The simulation is built using “SimPy” and uses fixed time slots for reservations. At the moment these slots are 15 min long and start on 0, 15, 30 and 45 min past each hour. Each Driver that wants to charge will attempt to make a Reservation for a Charging Point (via the E-Mobility Service) at a CSO. Because of these timeslots, a Reservation will only begin at the start of the next timeslot and will last for a number of time slots as calculated by the CSO. After the reserved time has passed, the Driver is expected to Plug Out its EV from the Charging Point, though he can do so at an earlier point in time (if the EV is already fully charged, or the driver otherwise decides to do so).

Below, we summarize some of the results received. These are preliminary results, and as the data retrieved from the simulator is extremely extensive, we identify only some of the more interesting aspects. In general, we found the tooling to be extremely useful, and, for example, several emergent behaviours were discovered in the data that, if proven true, would lead to significant challenges to the electrical grid. Furthermore, the aspects that the SoS was designed to test (primarily security and dependability) were successfully tested and validated. We plan to exploit these results in two manners: First to follow up on the results and perform more experiments after the end of the project, leading to publications, and secondly both Thales and ENCS are planning to make the simulator code available online, so other researchers can validate and use the same simulation code.

Results from the EV Charging Scenarios. The first scenario that we will discuss is a usage case where EV drivers do not immediately remove their vehicles from the charging points for a period of time. This means that those drivers who do not remove their EVs are acting badly – blocking charge points from other users. We chose a fixed period of 4 h for this behaviour in this scenario, and the simulation took place over a period of 24 h (simulated time via SimPy). There were 1000 EVs present and 500 Charging Points (CPs). In this simulation, market costs did not cause a significant change in behaviour. The EVs were set to desire a charge of 70%, with a “must charge” threshold at 20%. Four states are defined in the simulation: Charging, Driving, Idling and Waiting. Charging and Driving are self-evident. Idling was periods of time when the EV was not in use (due to lack of driver need). Waiting was an undesirable state where an EV was waiting for a CP to become available. In order to stress the Grid and

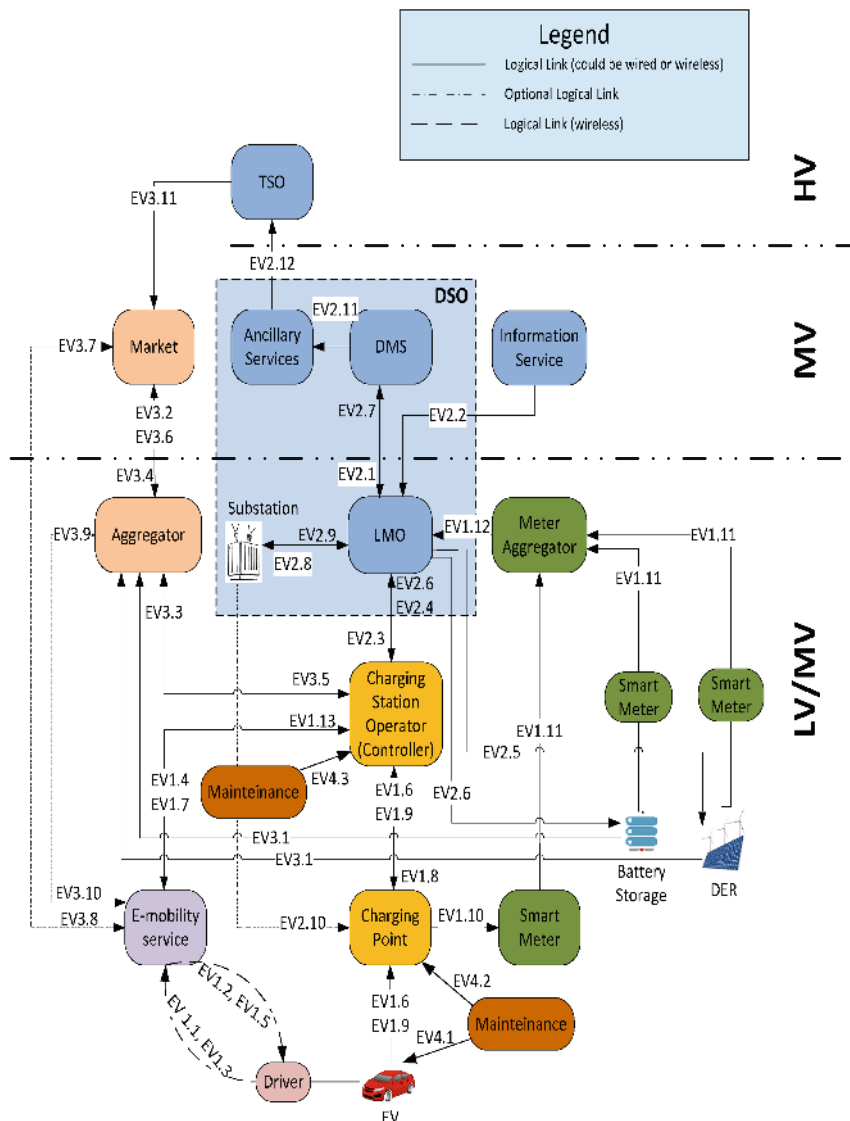


Fig. 11. EV Charging SoS

the SoS as a whole, Idling was the least desired state of the three normal states – the chance of an EV idling was set to 10%. Finally, note that the drivers, cars and initial state were randomly assigned and a period of 15 min was allowed to let the state settle. This can be seen in an initial jump in all of the graphs.

In Fig. 12, you see four basic types of behaviour: Driving, Idling, Waiting (for a free Charging Point) and Charging. The numbers on the X-axis reflect the number of EVs in that state. In this scenario, (and in all our tests, due to global variables that were

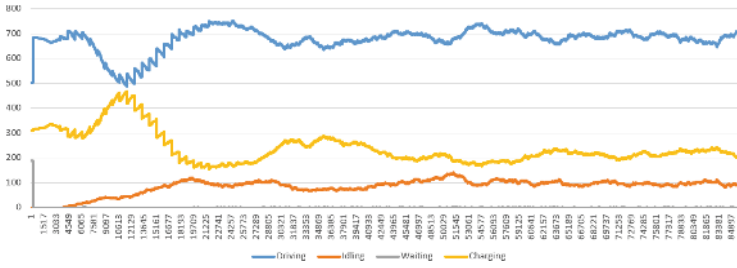


Fig. 12. Initial state where no drivers act badly

set) the initial state was close to optimum for each of the EVs, with around 70% driving and 30% charging. The initial spike in charging (around 10000 s into the simulation) is due to the initially driving EVs falling below the 20% lower threshold and requiring a charge. You can then see that after around 20000 s, the system reaches a steady state with around 70% driving, 20% charging and 10% idle. Note also that there is an insignificant number of EVs in the waiting state.

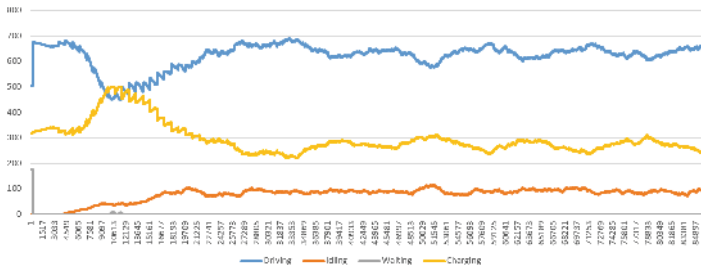


Fig. 13. 20% of EVs acting badly

Figure 13 shows the first set of EVs acting badly – in this case, 20% delay their disconnection from the CPs and block other drivers from using them. This scenario shows the resilience of the SoS to such behaviour – there is again the same spike in charging after around 10000 s, and a minor amount of EVs waiting, but again after around 20000 s, the SoS reaches a relatively steady state, although with more EVs charging (around 5% more) at any given time (and consequently 5% less driving) than the optimal case. One last thing to note is that all CPs are in use during the initial spike.

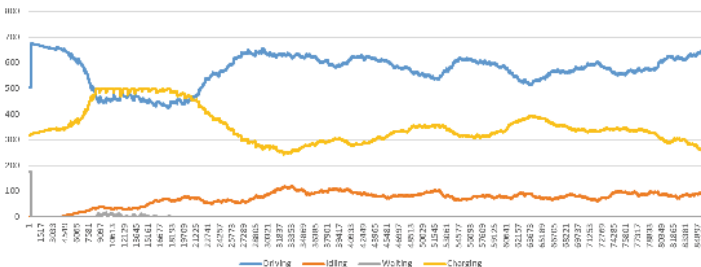


Fig. 14. 40% of EVs behaving badly

In Fig. 14, the consequence of bad actors is more apparent. Now, there is a noticeable issue for EV drivers during the initial spike in usage, with a period of time when all of the CPs are in continual use (for around 15000 s) and there are some drivers constantly waiting for service. However, the SoS is still very resilient to this issue – the number of waiting drivers is still very low (maximum was 16 drivers). Furthermore, the average wait time was around 600 s (10 min). Again, this shows the resilience of the SoS to malicious events. However, again there is a drop in number of active drivers when the SoS reaches a steady state of another 8%, with the number of charging drivers up by the same amount.

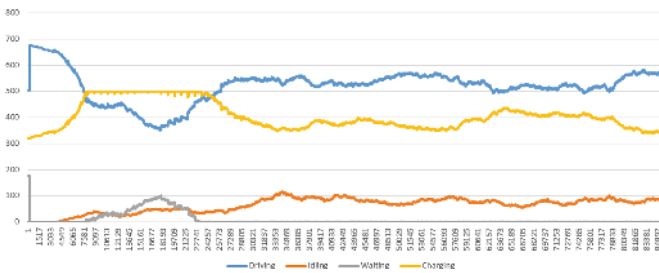


Fig. 15. 60% of EVs behaving badly

Finally, in Fig. 15, the first significant effects of bad behaviour can be recognized, while 60% of drivers are acting badly. First, the period where the initial spike causes full usage of all CPs, and consequently up to 100 EVs waiting for service at the worst point. Despite this, there is another drop in active drivers when the simulation reaches the steady state – down to around 58% from a high of over 70% in the control simulation.

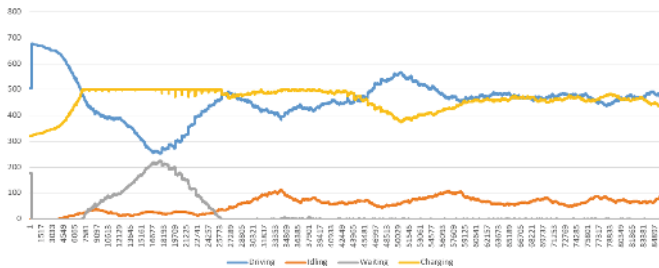


Fig. 16. 80% of EVs behaving badly

In Fig. 16, we can see some significant changes in the usage profile of the SoS due to the bad behaviour of the EV drivers. In this scenario, during the initial spike, there is now a period of around 35000 s (nearly 10 h in total) where all of the CPs are in constant use. This is also reflected in the more than 200 EVs waiting at one point. In this scenario, we now see less than 50% of the EVs driving when the SoS reaches a

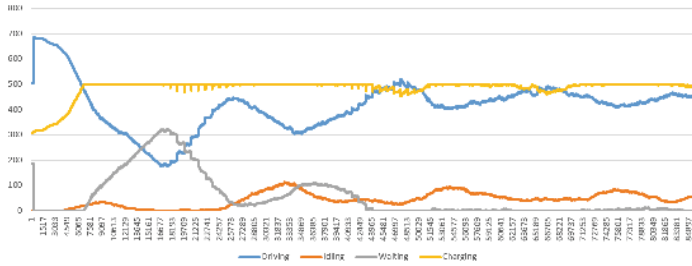


Fig. 17. 100% of EVs behaving badly

steady state, with roughly the same number charging as driving. This means that 20% of the EVs have changed from driving to charging since the control simulation. However, this is taking place where only 20% of drivers are behaving correctly and yet the SoS remains (for the most part) available for use.

In the final simulation, shown in Fig. 17, every driver is now acting badly. In this case, the effect on the SoS is dramatic and relatively catastrophic. In this simulation, for essentially the entire day, all of the CPs are in constant use, the number of drivers is down to less than 50% and for large periods of time, EVs are waiting to charge (more than 30% at the peak).

Market Simulation and Energy Usage. These simulations were intended to determine the reaction of the SoS to malicious behaviour on the part of the drivers. The results described above shows how the SoS behaves from the perspective of the EV drivers. However, there is another important aspect that can also be studied: the reaction of the SoS from the perspective of the Grid. This was calculated from the perspective of the TSO (see Fig. 11). The mission of the SoS is, basically, to ensure the stability of the Grid, and to ensure that large changes in energy generation are not required. In order to achieve stability, a number of measures were enacted. First, the CPOs (there are 5 independent CPOs in this simulation) received a wholesale price from the DSO, via the market, based on:

1. Forecasted demand (by the DSO) for the next 24 h, in 15 min intervals, and
2. How much energy they predicted that they required also in next 24 h, in 15-minute intervals.

The goal of a CSO was, using the market price, to ensure they did not stray too far from their forecasted need. Prices were set by the DSO based on five energy bands – the price per unit was set based on the band that the CSO requested, regardless of the actual energy used in reality. The ideal situation for a CSO was to get as close to the top of a band, without exceeding it, as the cost per unit would jump, and make the per unit cost more expensive. Therefore, if a CSO discovered that (based on EV reservations) that they were going to jump up to the next band, their behaviour would be to attempt to get many more customers, by reducing the customer price. We used this aspect to drive competition and the response from the domain experts was that this is indeed a desired future scenario. This aspect proves the evolutionary promise of an AMADEOS SoS design.

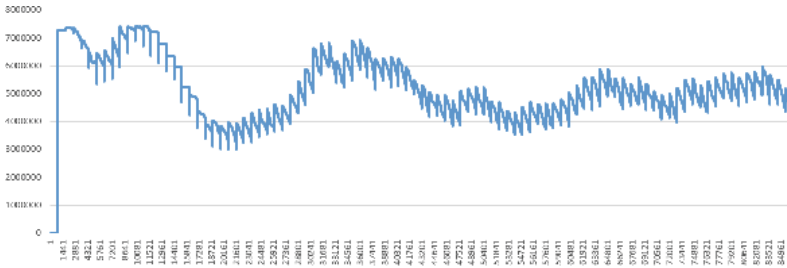


Fig. 18. Total network load with 0% bad behaviour.

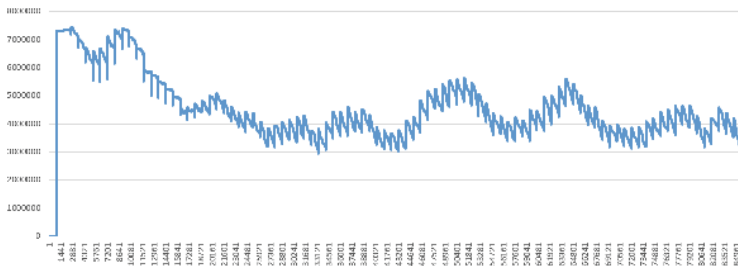


Fig. 19. Total network load with 40% bad behaviour

Based on the same simulations as shown in Section 0, Fig. 18 shows the reaction of the Grid to the EV charging scenario where no bad behaviour is present. The requests come in 15-minute intervals, and this can be seen in the jagged lines that are present in the graphs. In this instance, there is again an initial jump, after the 15 min settling down period and then a peak and trough after around 18000 s. This is again due to the jump in number of users charging at the start of the simulation based on their initial charge state and desire not to fall below 20%/attain 70% charge. The interesting outcome is the narrow band (between 40 MWh and 60 MWh) that the Grid eventually stabilizes towards. Future research will definitely consider running such experiments over several weeks of simulated time and integrating a typical Grid usage pattern into the demand¹.

The 20% (not shown) and 40% bad behaviour results (see Fig. 19) show how the bad behaviour is increasingly reflected in the variations of load over time. The intermediate results (60% and 80% bad behaviour) show increasingly wide variations, cumulating to the wide swings shown in Fig. 20, where all of the drivers are again behaving badly. These variations once again show how malicious actors will cause significant issues for Grid operators.

¹ As discussed in D4.2, the simulation was not run in concert with the MV case study.

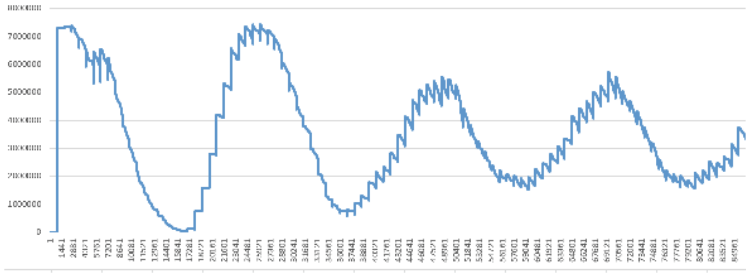


Fig. 20. Total network load where 100% of the drivers are acting badly

2.3 Household Management Case Study

Mission. The goals of the SoS are somewhat similar to the ones of the EVC: allow users, i.e. households, to use home appliances (e.g., flexible or general loads) at a convenient price while properly scheduling the energy consumption and distribution to improve revenues for energy providers with the constraint of keeping a balanced load on the grid.

During a normal scenario, a user wants to activate one or more home appliances, thus requiring for energy from the grid. Energy requirements are managed by an energy management gateway every time a person wants to activate home appliances. The energy management gateway handles the scheduling of energy consumption/provision within the household and sends requests to the coordinator to update consumption/production setpoints. The coordinator continuously updates the energy setpoints according to the grid energy availability. The actors need to be time synchronized to provide consistent information to the coordinator for scheduling and planning actions. Furthermore, some home appliances should be energy prioritized with respect to others (e.g., refrigerators, air conditioning systems, etc.).

Conceptual Level. In this section we report the result of the activities performed at the conceptual level for the Household management case study. For each viewpoint we list the most representative identified SoS requirements defined taking as input the SoS meta-requirements [1]. Though, the requirements come from meta-requirements, the traceability of the full set of requirements on meta-requirements is not provided. Some of the representative requirements are given below (the full set of requirements can be found [4]).

Viewpoint Examination. The objective for which the SoS is designed for is to allow end customers (i.e., households) to interact with the coordinator in order to request the activation of some particular appliance.

Architecture Viewpoint. The HHM consists of a subset of the CSs of the SoS “Household scenario” described in Chap. 2.2 of [4]. In particular (Table 2):

Table 2. Constituent systems for Household case study

Name	Description
Coordinator	An entity that receives energy prices and collects energy flexibilities. It applies optimization functions in order to shift power consumption and generation when energy prices are favorable.
Energy Management Gateway (EMG)	It is any device/software or group of them installed in the customer facilities that allows the visualization of metrological information, price and warning signals by the customer and has the capability to take action (e.g. rescheduling of power consumption/production) automatically or after approval by customer on any home appliances.
Smart Meter (SM)	A smart meter providing production and consumption values. May also enable advanced sensor facilities providing active/reactive production, frequency monitoring, voltage monitoring, etc.
Distributed Energy Resource (DER)	DER devices are generation and energy storage systems that are connected to a power distribution system.
Home Automation Device	Device providing additional functionalities enabling consumers to interact with their own environment (e.g., a smart thermostat).
Flexible Load	Load that can be controlled by the EMG (e.g., a smart washing machine).
Local Network Access Point (LNAP)	Provides the WAN connection for upload of the metering data.
Display	Main Human Machine Interface (HMI) between the householder and smart services.

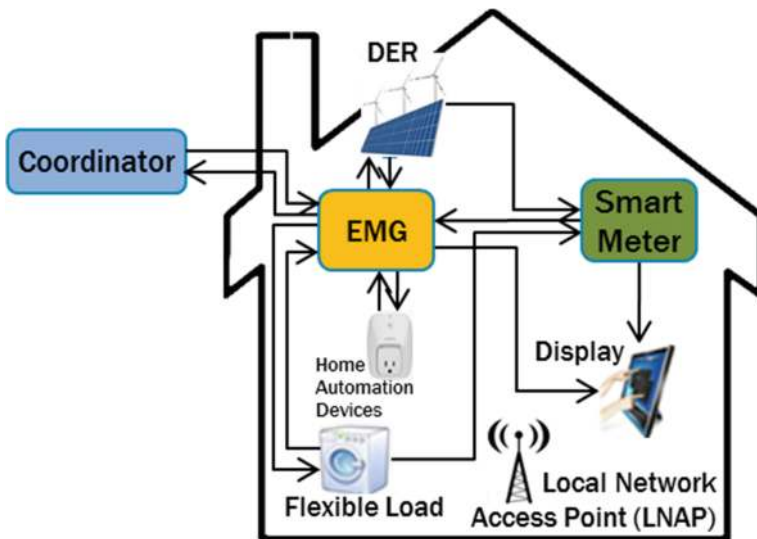


Fig. 21. Household management SoS architecture

High level representation of HH Management interactions.

A pictorial view of the HH management SoS is reported in Fig. 21.

Figure 21 shows a pictorial view of the temporal sequence 1–7 comprising, for each step, the involved systems and connections from Fig. 10.

In the following, the steps performed in a nominal household scenario are described. Each numbered step corresponds to a specific, adimensional, time instant at which the corresponding actions are carried out.

1. A person wants to activate the Flexible Load appliance. The Flexible Load sends an energy request to the EMG.
2. The EMG sends an aggregated energy request to coordinator.
3. The Coordinator decides whether to accept or reject the EMG request by updating energy setpoints according to the energy availability on the grid.
4. According to the Coordinator reply message, the EMG sends an OK/KO message to Flexible Loads Home Automation Devices and DERs.
5. In the case of an OK message, the Flexible Load is activated.
6. The SM measures the energy consumption and the resultant load for, respectively, billing operations and smart grid power flexibility purposes. Load information are forwarded to the EMG.
7. When the Flexible Load ends its tasks, the SM sends the total energy consumption and billing invoice to the user through the Display.

Dynamicity Viewpoint. The aforementioned case study is therefore very dynamic, in the sense that the service provided by the electrical grid is always changing, according to the customer requests for home appliances.

Evolution Viewpoint. Due to technological advances and new customer needs, the EMG should have the capability of easily integrate new HMIs, automation devices and loads. As an example, an householder may want the same interaction provided by the in-house Display on his smartphone and to use it not only while connected to the LNAP, but also when outside to schedule a washing machine while connected to the mobile carrier.

Multicriticality Viewpoint. Up to this point, we have described the case study assuming all the appliances with the same criticality. However, an HH consists of a large variety of possibly interconnected (net-centric) appliances providing different kind of services and, thus, characterized by different priority and criticality levels.

2.4 Medium Voltage Control SoS Case Study

Mission. The mission is to provide charging services to EV drivers and to provide energy-related services to households, but also to highlight interesting emergent phenomena that would not arise within the single SoS improving their interoperability with predictable, dependable behaviour avoiding negative cascading emergent effects.

Conceptual Level. The Medium Voltage energy distribution infrastructures are needed to interconnect the EV-Charging SoS and HH Management SoS and consist of the following set of CSs (Table 3):

Table 3. MV Energy distribution constituent systems

Name	Description
DER	Device able to produce energy in the grid.
Battery Storage	Battery used to store energy produced by DER
Smart Meter	A smart meter providing production and consumption values. May also enable advanced sensor facilities providing active/reactive production, frequency monitoring, voltage monitoring, etc.
Meter Aggregator	It is in charge of collecting metering data from the supervised smart meters.
LMO	The main software component connected to the grid which is in charge of providing power constraints and energy set points to the CSO.
DMS	A system which provides applications to monitor and control a distribution grid from a centralized location, typically the control centre. A DMS typically has interfaces to other systems, like a GIS (Geographical Information System) or an OMS (Outage Management System).
Information services	Commonly available services provided by a third party. E.g. weather information needed to predict PV production.
Substation	Substation system implementing the automation sequences and the control functions of interfacing process level control devices
Ancillary Services	Information services to TSO (e.g. extreme increment or decrement of the electricity frequency).
TSO	Operating centre for supervising critical regions of the transmission grid
Market	Set of services designed to evaluate the Energy price.
Aggregator	Is designed to provide information from the grid to the Market. The Aggregator is able to aggregate information incoming from sources reducing complexity and redundancy.

This section describes at logical level the infrastructures related to the Medium Voltage energy distribution (Fig. 22) and the examples of emergent phenomena that could be coming from the interoperability of Ev-Charging SoS and HH Management SoS.

In the following, the steps performed to manage the energy distribution are described.

1. Smart meters (SM) are devices connected to each prosumer;
2. SM collects and send information (also on demand) about power consumption/generation to the Meter Aggregator;
3. The Meter Aggregator sends the aggregated measures to the LMO (SM can provide information about energy consumption/generation to the EMG and the home Display);
4. The LMO receives information about Substation Monitoring Data, Household forecasted energy consumption generation and EV charging schedule;
5. The Distribution Management System (DMS) updates periodically LMO information for high level operation objectives, changes in data models (e.g. grid topology, newly connected charging station);

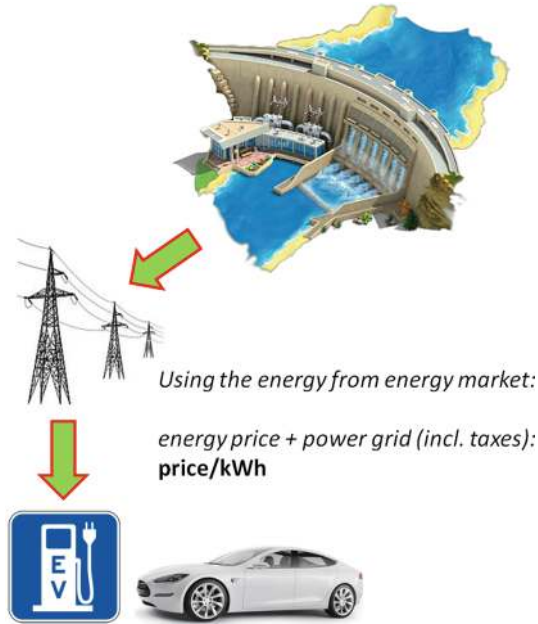


Fig. 23. The energy used to recharge an EV is bought from the market

their EV can be charged at a CSO by means of a CP, the energy used in this process is generally bought by the CSO from the energy market and sold at a specific price per each kWh (plus power grid fees including taxes) (Fig. 23).

The CSO may also buy energy directly from the HHs for recharging EVs, like, e.g., when the HH energy price is cheaper than the one provided by the market or for compensating effects of some energy-related service disruptions (e.g., disconnected energy generators). In this case the total energy price would be the energy cost from the HH plus the power grid costs with taxes (Fig. 24).

From the above scenario, the below beneficial emergence scenarios were identified:

Consider now the scenario in which the owner of the HH wants to recharge its EV to a CSO. It is evident that the energy needed to recharge the EV can be bought and provided, by the CSO, from either the energy market or HHs, including the one owned by the driver of the EV.

The HH owner can check the energy price, relative to the energy market, of the CSO and decide whether to recharge its EV with such energy or to use the energy stored at home using DER, i.e. the HH owner can compare the price of the energy stored at home with the one sold by the market and decide accordingly (Fig. 26).

Suppose that the energy on the market is more expensive than the one stored at the HH. The HH owner will then decide to use its own stored energy to recharge its EV, enabling the energy transmission from the house to the CSO. Using the energy stored at home will cost the owner to pay only the provision of power grid including taxes (the price may be related to the total power grid usage time or to total transmitted kWh) (see Figs. 25 and 27).

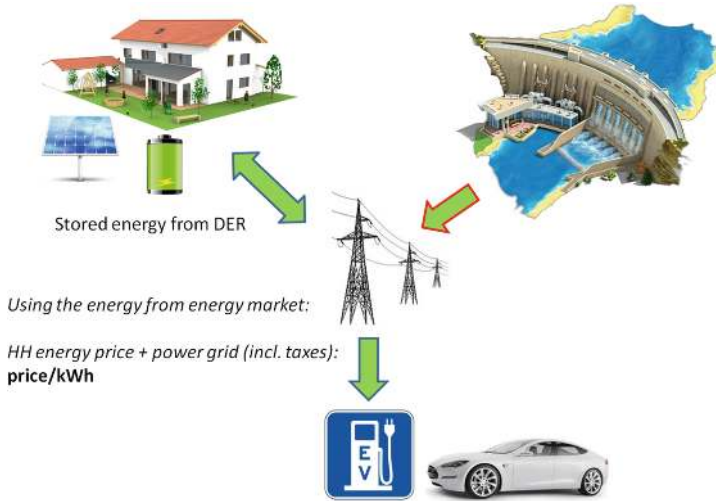


Fig. 24. The CSO buys energy from the HH.

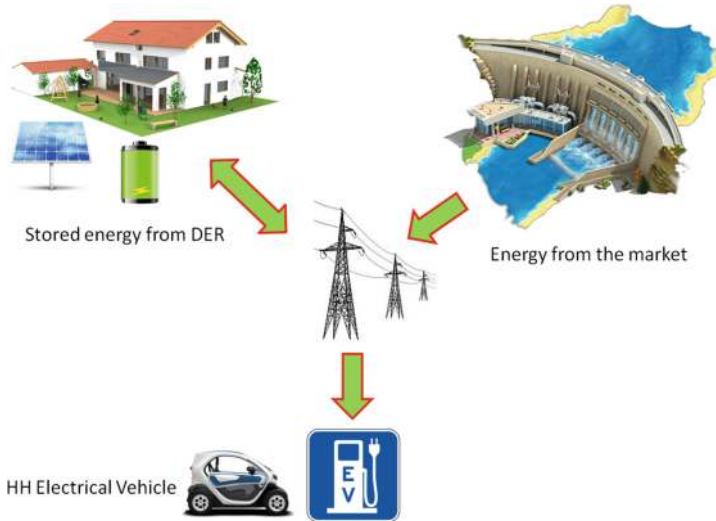


Fig. 25. The CSO can buy energy from either the market or HHs

These new emergent behaviors clearly arise with the interconnection and the available communications between smart constituent system of a SoS, which the AMADEOS framework is able to capture and describe. This way the possible large variations in the network load could be attenuated by a suitable adaptation of the prices. However, if this adaptation process is not correctly implemented (for instance, if the



Fig. 26. The HH owner can check and compare the price of the energy stored at home with the one of the market

adaptation is too quick and steep), then undesirable global phenomena, such as oscillations of prices and load, could take place.

Time Viewpoint. For identifying some of the time aspects, we describe two scenarios that extend the previous case studies with descriptions of some systems, processes, and interactions where the existence of an accurate global time is essential for measuring and controlling the state of the power grid.

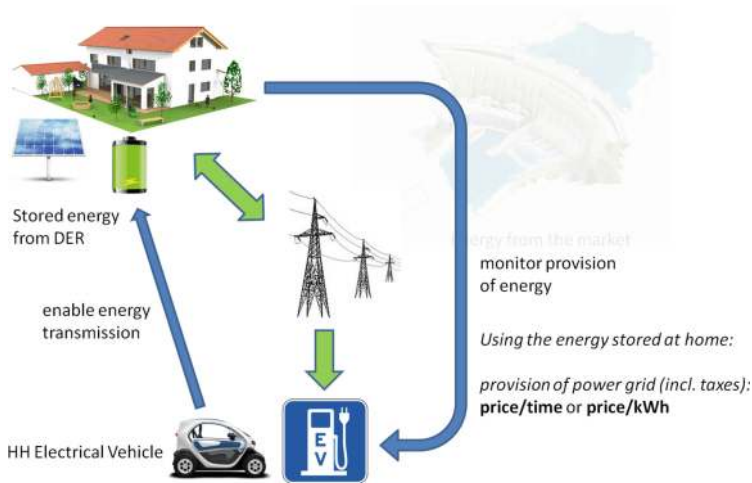


Fig. 27. The HH owner decide to recharge its EV with the energy stored at home

Scenario 1: Maintaining the Power Phasor Parameters. The power phasor is defined by the amplitude, frequency, and relative phase of the electric current or voltage in a power line. As the load in the power network varies, the braking momentum of the power generators increases, so the generators' rotors slow down. All variations in network load lead thus to variations in power frequency. Traditionally, these variations were compensated locally, by applying the load measurement signals to an Automated Generator Control (AGC), which basically is an electro-mechanical governor that compensates for the changes in braking momentum. The AGC signals are collected from different points in the network, and forwarded to large production facilities [5]. In order for the AGC to work efficiently, the following requirements must be satisfied:

- Timely measurements of network loads.
- Timely adjustment of voltage controllers.

The adjustment signals for AGC are generated once every 2 to 4 s, so the transmission time requirements are not very strict [5].

Another phenomenon that alters the power phasor is the reactive power produced by the reactance (i.e. the reactive part of impedance) of a load. A capacitive load will make the voltage lag behind the current, whereas an inductive load will make the current lag behind the voltage. Although the reactive power does not dissipate energy, it affects the efficiency of power transmission to the consumers, so this too must be controlled, such that the reactive angle φ_x is kept as close as possible to zero.

To compensate for reactive power introduced in the network primarily by motors and transformers, capacitive loads (capacitor banks) can be switched on or off by control units. Unlike the AGC, which acts centrally at the production facility, the reactive power control can be distributed across the grid [6]. In the AMADEOS scenarios this can be achieved by the LMO.

In a purely reactive approach the individual measurements by SMs are collected by LMOs (via the EMGs or CSOs), which try to compensate for load variations using the locally available power reserves and by re-scheduling, when possible, of some loads. When these local measures are not sufficient, appropriate AGC signals are transmitted to the power generation facilities. Of course, particular implementations could include additional aggregation layers, but those do not change significantly the problem.

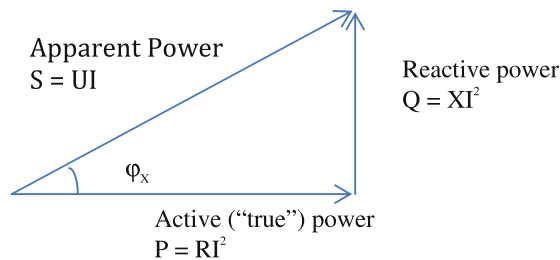


Fig. 28. Power triangle

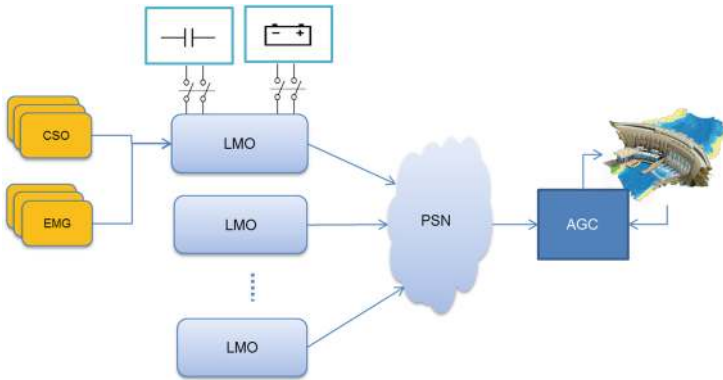


Fig. 29. Load measurement and control architecture

Concerning the reactive power in the grid, the LMOs try to locally compensate. In case when the local capacitor banks do not suffice, a signal must be broadcasted (Reactive Load Control - RLC) to request additional capacitive loads be switched on by other LMOs.

Overall, the load measurement and control architecture includes a number of geographically distributed LMOs and one local AGC for each of the power plants (see Fig. 28).

The effectiveness of this reactive approach is limited, due to the delays in the transmission of AGC signals and the time required for adjustments. A more advanced approach (see Fig. 29) tries to predict the variation of the load based on SCADA measurements from various places in the network, and a forecasting of load variations [7]. The prediction function is based on a sampling of the real load over some past time interval.

Although the AGC and reactive load balancing are still slow (in the order of seconds), a good estimation of the current state of the system, as well as the load prediction, require better time accuracy than in the previous case, and global time awareness. Indeed, if two consecutive measurements from a given location in the power grid arrive in reverse order, then the detected variation trend is reversed. If this reversal occurs randomly over a large number of measurements, then the whole predictive load balancing has, on average, no effect. In a worst-case scenario the order of a sequence of measurements can be altered such that the variations are amplified, leading potentially to the activation of circuit breakers on some network segments.

Specific timing requirement can be identified for the two cases:

A. Predictive balancing of active loads – in this case we assume that LMOs compete with each other, so they are not willing to use their local energy reserves to balance the load at other LMOs. In this case AGC messages are sent to the power plant to increase its power output. Similarly, when the load decreases and LMO's storage capacity is reached, another message is sent to decrease the generators' power output. Obviously, these are sent asynchronously by all LMOs. In order to ensure a proper ordering of these events at the local AGC Message Queue, all AGC messages must be

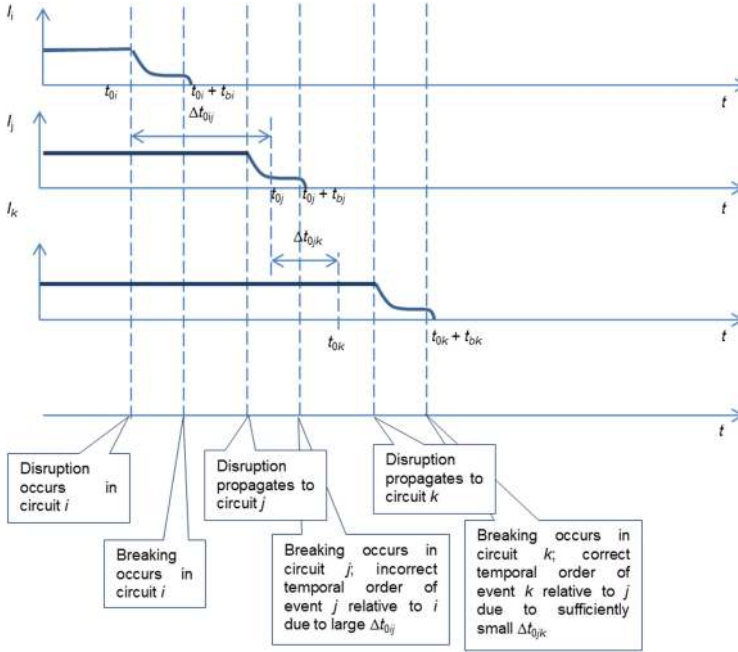


Fig. 31. Cascaded propagation of disruption and time stamping of breaking events; in nominal terms, $t_{0i} = t_{0j} = t_{0k}$

The time requirements resulting from these two cases are:

1. The maximum synchronisation error should be smaller than the sampling interval of the load prediction function; otherwise, two messages arriving in reversed order could indicate a wrong trend (cases A and B)
2. The polling of the queue should be done at specific time slots allocated to each LMO (time-division multiplex); the maximum synchronisation error (i.e. difference between local clocks) should be smaller than the time slot minus the time required for processing the queue (case B)

Scenario 2: Forensic Analysis of Disruption Events. When a disruption occurs in a network, it may lead to cascading effects (Fig. 31). In case of complex networks of producers and consumers it is not always easy to identify the original event that initiated the cascaded failures. However, this information is crucial when network design needs to be improved to increase the resilience of the network, or when the network operators are liable for the quality of the services they provide.

To cope with this requirement each circuit breaking event could be logged by the breakers' control computer. This way a chain of events can be reconstructed and the root cause identified. However, given the high speed in which these events occur³, the temporal order of the logged events may be distorted by the differences in the local

³ A typical reaction time for modern equipment is 1/8th of a 20 ms cycle [8].

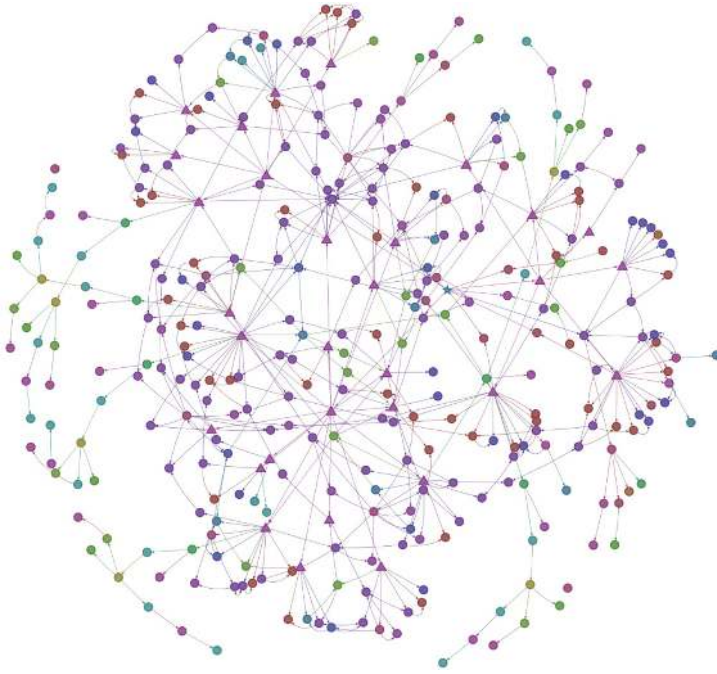


Fig. 32. The full SmartGrid model generated using the model query: “return true;” (i.e.: select all blocks). Here, the triangles represent systems, star represents the SoS and other blocks are represented by circles. The color of the blocks is the same as the one in the Blockly model.

clocks used for time stamping. For this reason, it is necessary that the synchronization error between local clocks is kept below the breaking time of the fastest breaker plus the propagation time of the disruption. This is illustrated in Fig. 30, where a time value t_0 corresponds to three different moments in three circuits whose local clocks are not synchronized. If the delay between two local clocks is too large, as it is the case for circuits i and j , then the events cannot be ordered correctly.

Finally, for both scenarios the maximum transmission times for all the messages related to events occurring in the smart grid must be guaranteed. If this is not the case, then most of the functions described here cannot be implemented.

3 Conclusion

This chapter has shown how a realistic case study can be modeled in the Blockly tool and directly simulated. A high-level view of the entire model can be seen using the model query tool of supporting facility to select all blocks. Below is the graph of the model consisting of elements and relationship between each blocks. The diagram shows the complexity of the full model (Fig. 32).

Modeling complex and pervasive infrastructures as the one used as case study clearly highlights how the support of a precise conceptual model and of specific tools

for its instantiation is fundamental for a sound and comprehensive codification of the various properties of the whole. At design time the identification of causal loop in the lower levels of the hierarchy, enabled by the support for simulation through model execution, is a mandatory step to identify possible emergent behaviors at the higher levels, that may lead, also in future evolution of the system of systems, to a violation of system requirements. A correct representation of the environment is necessary. Global time Awareness and monitoring are fundamental to early detect and to contain the effect of detrimental emergence phenomena at run time. The main benefits of the AMADEOS approach can be easily seen in the results from the simulations: The AMADEOS architectural framework and associated tools allow an SoS architecture to be comprehensively designed and a simulation extracted that can be tested. This allows system architects to quickly test hypothesis regarding future systems and determine what attributes will lead to advantageous or poor results.

References

1. AMADEOS Consortium, Deliverable D1.1 - SoSs, commonalities and requirements (2014)
2. AMADEOS Consortium, Deliverable D2.2 - AMADEOS Conceptual Model (2015)
3. AMADEOS Consortium, Deliverable D4.1 - Case study and use cases (2015)
4. AMADEOS Consortium, Deliverable D4.2 - Case study realization (2016)
5. Bhowmik, S., Tomsovic, K., Bose, A.: Communication models for third party load frequency control. *IEEE Trans. Power Syst.* **19**(1), 543–548 (2004)
6. Saypasrth, P., Premrudeepreechacharn, S.: Allocating reactive power using artificial neural network. In: 2011 Asia-Pacific Power and Energy Engineering Conference (APPEEC), Wuhan, pp. 1–4 (2011). doi:[10.1109/APPEEC.2011.5749101](https://doi.org/10.1109/APPEEC.2011.5749101)
7. Tomsovic, K., Bakken, D.E., Venkatasubramanian, V., Bose, A.: Designing the next generation of real-time control, communication, and computations for large power systems. *Proc. IEEE* **93**(5), 965–979 (2005)
8. ABB Miniature Circuit Breakers – Application Guide. <https://library.e.abb.com/public/ef5f3a6e43c99c288525758500534fe6/1SXU400142M0201.pdf>

Open Access This chapter is distributed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, duplication, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the work's Creative Commons license, unless indicated otherwise in the credit line; if such material is not included in the work's Creative Commons license and the respective action is not permitted by statutory regulation, users will need to obtain permission from the license holder to duplicate, adapt or reproduce the material.

