Autonomous Power Management
Architecture Notebook
Elaboration – Iteration 1

1. Purpose

This document describes the philosophy, decisions, constraints, justifications, significant elements, and any other overarching aspects of the system that shape the design and implementation.

2. Architectural goals and philosophy

The system will be designed as a multi-agent system, since MAS Adequacy Confirmation showed that such an approach is viable. For this reason, a modular structure, driven by separation of concerns between the different agent types will be a main architectural goal and philosophy. Furthermore, to coincide with PosoMAS's architectural areas, the individual agent, the agents' organisations, agent interactions, and the system level will be regarded separately.

3. Assumptions and dependencies

We assume that there exists a sufficiently secure communication infrastructure and that interfaces to measuring systems and sensors in the power grid are available. We furthermore assume that the legal and contractual foundations for an external control of decentralized energy generators exist.

4. Architecturally significant requirements

The following goals and requirements are taken from the System Goals Document and are suspected to have the biggest impact on the system's architecture:

- Current schedule communicated to power plants
- Interface provided to receive schedules
- Maintain[Suitable AVPP Structure]
- Interface provided for communication of predictions
- Interface provided to access weather forecasts
- Use power plant or consumer state in predictions
- Predictions communicated in timely fashion

5. Decisions, constraints, and justifications

Since scheduling plays an integral part in the system, it has been modelled as a component of its own. That allows quickly exchanging the actual scheduling algorithm or using several at once. This is useful if it is necessary to calculate a solution within a certain amount of time and a heuristic and an optimal algorithm can run in parallel. If the time runs out, the heuristic can provide the solution and if the optimal algorithm found a solution before the deadline, this can be used. Modelling the scheduling algorithm as a separate component forces us to clearly define an interface for such purposes.

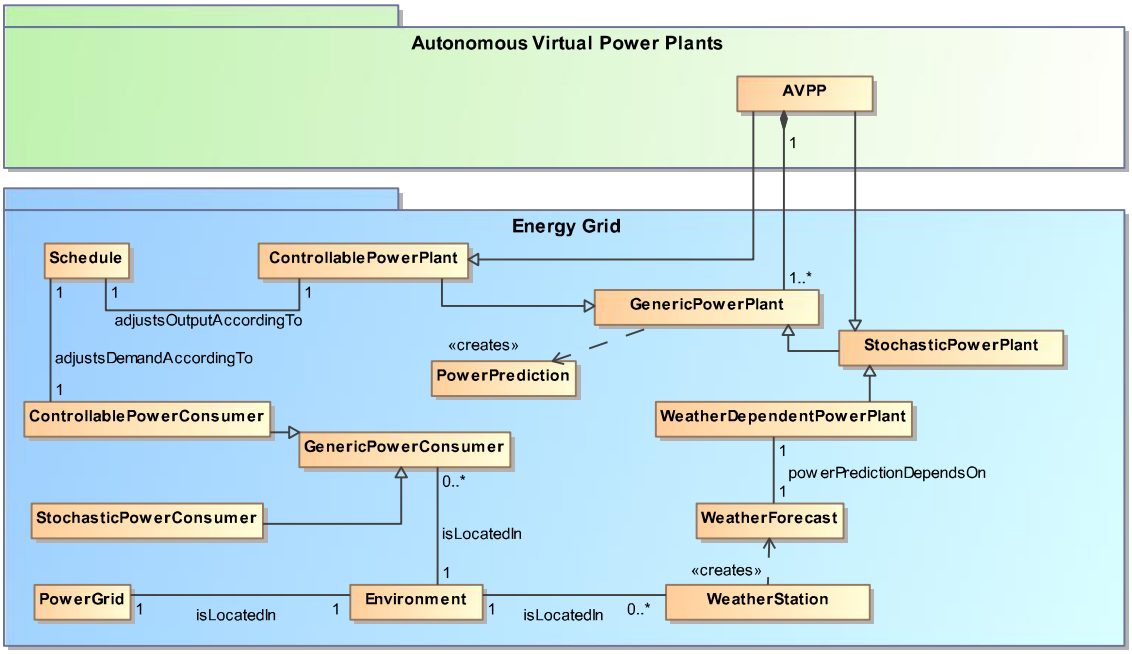
6. Architectural Mechanisms

A number of patterns and architectural styles have been used in the creation of the design in the elaboration phase according to the PosoMAS practice "Pattern-driven MAS Design". The following list of architectural styles introduces them and their origin briefly.

1. Trusted Energy Management Reference Architecture [1]: A reference architecture containing the most important concepts and relations relevant in the energy management domain. Has an impact on the conceptual domain model and the system architecture.
2. Direct Trust Pattern [2]: Introduces the concepts necessary for agents to define contracts on interactions, create experiences from interactions, and create trust values for use in decision procedures from these interactions.

7. Key abstractions

The key abstractions of the system correspond to the domain model, shown below. The elements are described in the Glossary. The reference architecture for trusted energy management [1] was used as an inspiration for the model.



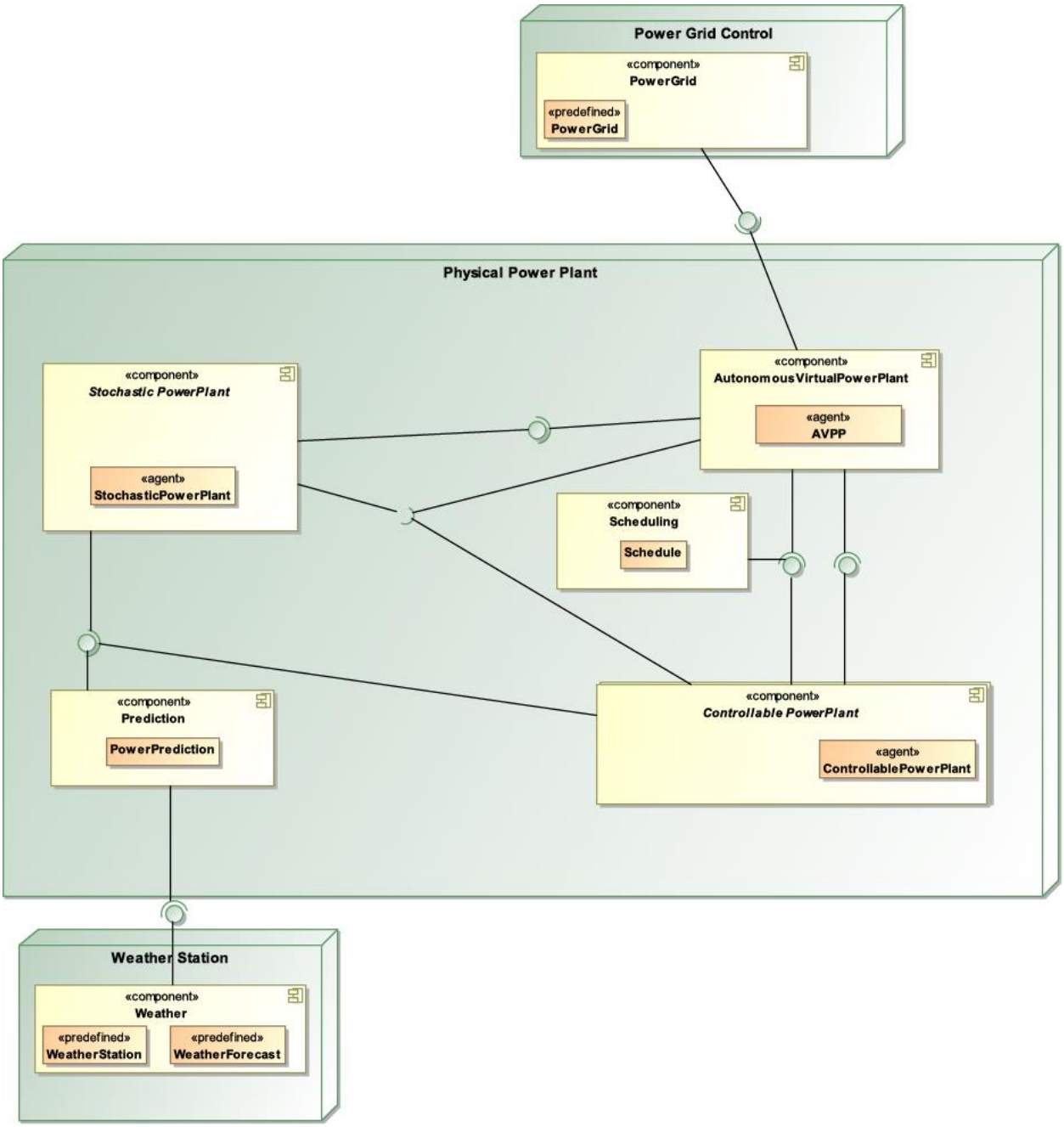
8. Layers or architectural framework

The different layers on the architecture will follow the four areas defined by PosoMAS:

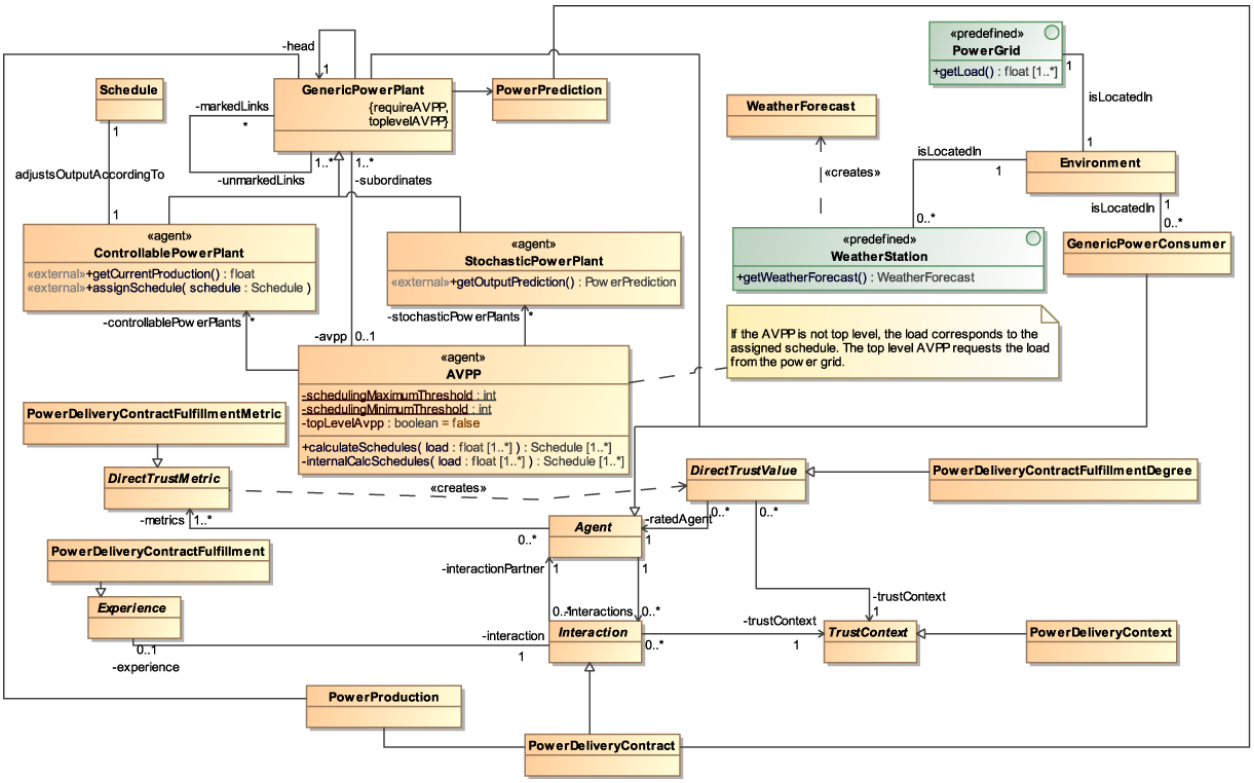
- **Agent Architecture**: the design of the individual agents, separate for each type of agent.
- **Agent Organisation**: the specification of an organisational paradigm that will structure the agent population at runtime.
- **Agent Interaction**: the definition of interfaces and protocols used by agents to exchange information, delegate control, and change their organisational structure.
- **System Architecture**: the relationship between the different types of agents, the supporting infrastructure, and external components as well as the environment.

9. Architectural views

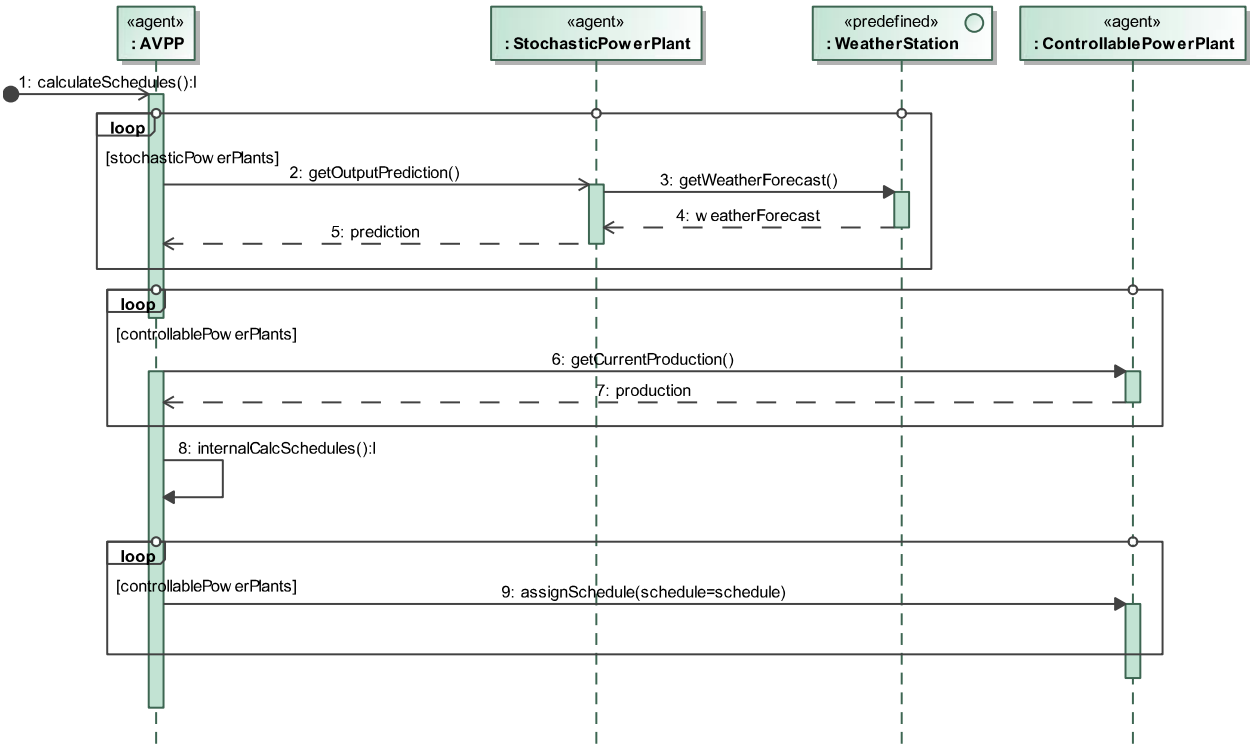
A high-level, abstract view can be seen below, identifying the different elements as components. In comparison to the inception, the diagram has changed since Weather Stations are now predefined components that will be reused by the system.



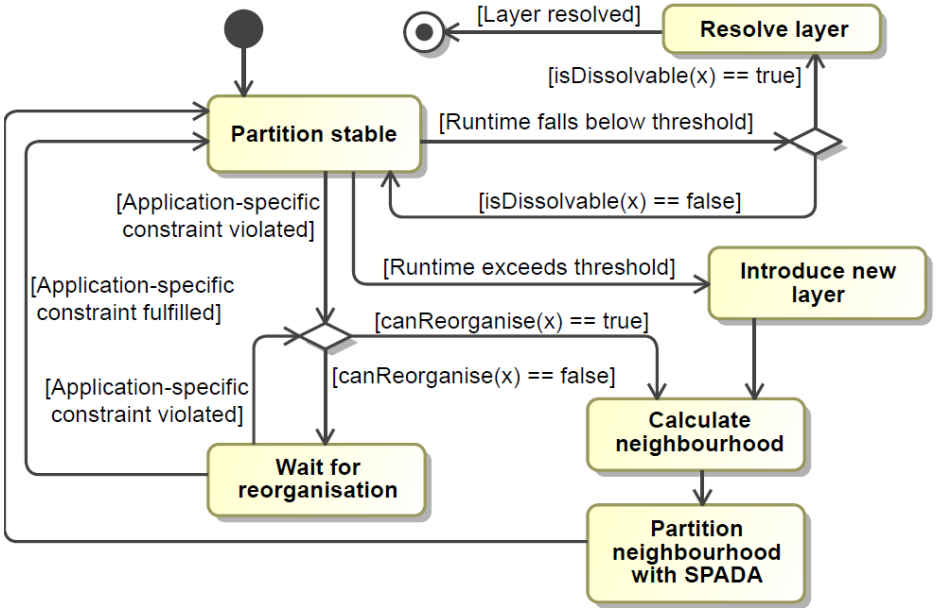
The design class diagram below details the concepts captured in the Conceptual Domain Model shown above and includes the architectural styles introduced in Section 6.



The sequence diagram below shows the basic interaction necessary to create a schedule.



The formation of hierarchies is motivated by the system requirements to maintain a suitable AVPP structure and especially to keep the scheduling times within certain bounds. The chosen self-organisation mechanism HiSPADA [3] uses the following control loop:



10. Bibliography

[1] Gerrit Anders, Jan-Philipp Steghöfer, Lukas Klejnowski, Michael Wissner, Stephan Hammer, Florian Siefert, Hella Seebach, Yvonne Bernard, Wolfgang Reif, Elisabeth André, Christian Müller-Schloer: Reference Architectures for Trustworthy Energy Management, Desktop Grid Computing Applications, and Ubiquitous Display Environments. Technical Report 2013-05. Faculty of Applied Computer Science, University of Augsburg, April 2013.

[2] Gerrit Anders, Jan-Philipp Steghöfer, Florian Siefert, and Wolfgang Reif. 'Patterns to Measure and Utilize Trust in Multi-agent Systems'. In Self-Adaptive and Self-Organizing Systems Workshops (SASOW), 2011 Fifth IEEE Conference on, pp. 35 –40. IEEE Computer Society, Washington, D.C., Ann Arbor, MI (Oct. 2011)

[3] Jan-Philipp Steghöfer, Pascal Behrmann, Gerrit Anders, Florian Siefert, and Wolfgang Reif. 'HiSPADA: Self-Organising Hierarchies for Large-Scale Multi-Agent Systems'. In ICAS 2013, The Ninth International Conference on Autonomic and Autonomous Systems, pp. 71–76. IARIA, Lisbon, Portugal (March 2013).