European Commission

Information Society and Media Directorate - General

Model of local energy resource cluster

SmartCoDe

Project No.:	ICT-2009-247473
Deliverable No.:	D-1.1.1
Deliverable Title:	Model of local energy resource cluster
Due Date:	December 31th 2010
Nature:	Report
Dissemination Level:	public
Authors:	Christian Wysoudil, Markus Damm, Tamás Bertényi, Tom Young
Lead Beneficiary No.:	3
Lead Beneficiary:	TUV

SmartCoDe has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement n° 247473

Contents

1 Introduction

In order to optimize energy consumption using an Energy Management Unit (EMU) and a wireless network, we need to develop an easily understandable picture of the components involved. This initial work concentrates on qualitative characteristics of Energy using Products (EuPs) and Local Energy Providers (LEPs) and combines the first results of Tasks 1.1 and 1.2 in Work Package 1.

The knowledge and information gained through the work described here is a basis for several ongoing and upcoming endeavours in the SmartCoDe project:

- Task 1.3 Energy Generation Forecasting
- Task 1.4 Demand side management in local grids combining regenerative energies and household/office appliances
- Task 1.5 Automatic power management
- Task 2.3 System design of a SmartCoDe node
- Task 4.2 SmartCoDe Demonstrator

This report is organised as follows: In Section 2, we present the current state of EuP classification. Section 3 discusses classification of LEPs, and models for wind and solar power generation are presented in Section 4. Section 5 discusses some consequences regarding energy management in the SmartCoDe network before concluding.

2 EuP classification

The original approach proposed in the SmartCoDe Description of Work (DoW) was to base the abstract EuP models on discrete Markov state models. However, this does not take into account the specific service offered by the EuP as well as the user interaction. The different states of such a Markov model would basically correspond to the different power consumption levels, which are often simply on and off. Therefore, very different appliances would fall into the same category if we just look at the states, e.g. a lamp would look very similar to an electric kettle because both have an on- and an off-state. Also, similar devices may fall into different classes, e.g. fans might have different numbers of power consumption levels.

Therefore, if we want to use Markov models as the basis of the classification, we would also need to take into account the transition characteristics. These, however, are often user-specific. The Markov model of a washing machine in a one-person household will have different transition probabilities than one in a family household, since it is used less frequently.

Consequently, the proposal presented here is not based on Markov models, but on the following considerations:

1. What service does the EuP offer?
2. What interface does it provide?
3. How can it be controlled by the EMU to reach the goals of SmartCoDe?

However, the discussion above shows that the user behaviour is a variable which has not yet been sufficiently taken into account. In particular, simulation of a SmartCoDe network requires models for the influence of user behaviour on the power consumption as well as methods to generate user-induced loads. We still need to decide how to tackle this problem.

2.1 Current proposal

Figure 1 shows an overview of the current proposal for the classification of Energy Using Products (EuP), and Table 1 on the next page goes more into detail. The first three columns of Table 1 contain the class name, the source of the class name, and a brief description of the class. The "Parameters" column contains three sub-columns:

- **Configuration:** These parameters are updated rarely by the user through the EMU, or by the EMU itself according to a schedule or a policy defined by the user.
- **Sensor input:** These parameters are provided either by the EuP itself or by the SmartCoDe node's sensor interface.
- **Online user input:** These parameters are updated frequently by the user directly at the device (possibly through the SmartCoDe node).

Note that only those parameters are listed which are specific to the particular class. There are additional generic parameters which are relevant for every class, like the power consumption of a EuP. These parameters are referred to as SmartCoDe parameters from now on. Exact data type definitions will be decided at a later stage in cooperation with Work Package 2.

The column labeled "Action" gives an outline on how this EuP class could be handled by SmartCoDe depending on the parameters. The actions are described in a way that leaves open which part of the action is handled by the EMU and which part is handled by the SmartCoDe node. For example, the action described for the class VARSVC (which effectively constitutes a control loop) could be handled by the SmartCoDe node itself. It also leaves open the question of how far SmartCoDe parameters are used for the handling of the specific device.

The last column gives some examples of EuPs falling into the respective class.

Figure 1: Overview of the classification of Energy Using Products (EuP) in SmartCoDe

Table 1: Classification of Energy Using Products (EuP) in SmartCoDe

			Parameters				
Class	Abbrev. from	Description	Configuration	Sensor input	Online user input	Action	Examples
SKDSVC	schedulable service	The EuP provides a service which runs a program for a certain time and can be scheduled within a certain time span.	runtimes & power profiles of the different programmes	none	earliest start time, latest stop time,	On input of earliest start time and latest stop time, SmartCoDe has to find a start time within the given bounds which minimizes costs.	washing machine, dryer, dishwasher, baking machine
VSTSVC	virtually storable service	The EuP provides a inert service which can serve as a virtual storage	interval defining upper & lower tolerance bounds	value describing the current state of the service, mostly temperature	current user demand	SmartCoDe has to keep the state of the service within the threshold values such that costs are minimized, exploiting the virtual storage property.	Fridge, Freezer, HVAC, Water-boiler
VARSVC	variable service	The EuP provides a service which might vary due to user interaction and/or daytime	interval defining upper & lower tolerance bounds	value describing the current state of the service, e.g. illuminance	current user demand	SmartCoDe has to keep the state of the service within the threshold values (determined by the current user demand and the tolerance bounds) such that costs are minimized.	lighting controlled by illuminance level (e.g. in garden, at entrance), dimmable lighting, blinds
ETOSVC	event-timeout controlled service	device is switched on and kept on by sensor events, and switched off in absence of sensor event	absence time span for switching off	event, e.g. presence detection	none (indirectly through sensor input)	SmartCoDe switches device on if event is detected, and switches it off after the time span set if the event did not occur again.	lighting controlled by presence detector (e.g. on corridor)
COMCON	complete control	charging and using up power decoupled; latter only restricted w.r.t. time slots & minimal service	minimal runtime per time span, time slots	current charge status	none	SmartCoDe charges the device and runs the device within the given time slots such that costs are minimized.	robot vacuum, robot lawn-mower
CHACON	charge control	charging and using up power decoupled; latter is mostly (or solely) user-dependent	charging policy	current charge status, device presence	device removal	SmartCoDe charges the device according to the charging policy such that costs are minimized.	battery & cellphone chargers, hand-held vacuum, emergency backup storages
CUSCON	custom control	device does not fit into other classes, therefore custom control by user and/or EMU	none	none	user demand / EMU demand	SmartCoDe does not control the device except through direct user-input or EMU control	HiFi, PC, Oven

2.2 Remarks and open questions regarding EuP classification

The focus of our considerations in SmartCoDe are SKDSVC (e.g. washing machines, dishwashers), VSTSVC (e.g. fridges, heating), VARSVC (e.g. lighting), and CHACON (e.g. charging an electric car). This section collects some remarks and open questions regarding these classes.

2.2.1 SKDSVC (schedulable service)

If we take the washing machine as a typical representative of this class, the usual user operation is

1. Load the machine
2. Choose a washing program
3. Start the machine

In the context of SmartCoDe, step 2 is replaced by

2. Choose a washing program **and a time interval** for SmartCoDe to schedule the operation in

Therefore, SKDSVC devices need additional user input (time interval), but for energy management we also need to know

- the runtime of the program the user chose and
- the *load profile* of the program

Depending on where this information is processed, this may require a custom message type to transport this information within the SmartCoDe network. However, in the currently favoured decentralized approach (see Section 5), this information would only be processed within the SmartCoDe node itself. The details of how this information is accessed by the SmartCoDe node have yet to be finalised. The most straightforward solution is to provide an interface to select the chosen program, while the load profiles of the different programs are already stored within the SmartCoDe node.

The **energy management task** here is to find a point in time to start the program such that it completes within the given time frame and where the load profile of the program incurs minimal costs. For example, the program could be scheduled such that peaks in the load profile like heating cycles fall into times where energy is cheap.

2.2.2 VSTSVC (virtually storable service)

Apart from SKDSVC, this class is the most interesting one in terms of energy management. For the cases we consider in SmartCoDe, each VSTSVC node has a thermal capacity which is charged either by cooling (fridge, air-conditioning) or warming (heater). It also has an associated temperature (e.g. fridge or room temperature) which has to be kept within certain bounds by cooling or heating. The temperature is measured either directly at the node via a sensor attached to the sensor interface, or by a remote sensor elsewhere in the network.

The **energy management task** for VSTSVC devices is to charge its thermal capacity at favourable times, while maintaining its temperature at an acceptable level. For example a room could be cooled down more than usual by the air-conditioning during favourable wind conditions, allowing the air conditioning to be switched off later when the winds are lighter.

The thermal behaviour of a typical VSTSVC EuP can be modelled using low-pass filters [5]. Such models can be used for predictions of the temperature that can be incorporated into the energy management process.

2.2.3 VARSVC (variable service)

In the SmartCoDe context, this class covers mostly lighting applications. Regarding energy management, nothing much can be done here since light is a service which is demanded by the user at unforeseeable times. Switching of lights automatically using presence detection is covered in the class ETOSVC. A possible application for energy *saving* is to couple the dimming of lights in a room to the illuminance level in this room, so that only the minimal required amount of artificial light is used. However, this is not dependant on the current energy cost; dimming based on the cost, i.e. having less light when the cost is high is unlikely to be tolerated by the user.

For emergency-like situations like power outages, however, it is possible to define special behaviours for each node. For example, it would be possible to disable lights except in rooms with no natural light sources.

2.2.4 CHACON (charge control)

This class is meant purely for consumers like electrical car battery chargers; the charged energy won't be fed back into the system. For energy storage devices which can feed back into the local network the LEP class ENSTOR should be used instead (see Section 3).

Regarding energy management, this class is similar to VSTSVC but usually remains charged for much longer than VSTSVC devices retain their heat. However, unlike VSTSVC, we can't always charge CHACON devices whenever we like because the charged device will be removed by the user at unforeseeable times. Also, certain charging policies have to be considered, for example charging with a certain profile to extend the operation time [4].

Therefore, the **energy management task** for CHACON devices is to charge them at favourable times for the network while also adhering to certain charging policies. Knowing when the user will remove the device would be helpful and enable much better planning. For larger CHACON devices like electrical cars, a user interface for this could be foreseen. This is unrealistic for smaller devices like hand-held vacuums; determining the typical user behaviour automatically might be an option here.

2.2.5 Technical and legal limitations of EuPs

For some EuPs, there might be limitations regarding how often they can be switched on and off:

- Old fridges can break (compressor failure) due to too high switching rate.
- Certain lamp types (e.g. high pressure lamps) have time limits for switching them on again after the last switch off.

For other types of EuPs, certain requirements regarding the duty cycle might need to be met:

- Boilers might be obliged to heat up water to certain temperatures in certain limits for hygienic reasons (e.g. 70° C once a week to avoid Legionellae contamination [2]).
- Water pumps might need certain minimal operation (e.g. once a week for one minute) to avoid jamming even if the water they pump is not needed.

These limitations will be considered when necessary, but will not be explored exhaustively. They will result in extra boundary conditions to be incorporated into the energy management process.

3 LEP classification

Local Energy Providers (LEP) are nodes on the SmartCoDe network that can provide electrical energy to the local grid. There are many potential sources of energy, but these have been grouped into four classes. Each class offers a fixed interface of data and services which allows the rest of the SmartCoDe system to interact with the node in a generic way.

The LEP classification follows a similar philosophy to the EuP classification, namely concentrating on what can be done with the LEP in the SmartCoDe context (services, interfaces). This results in certain LEPs falling into the same class, although they are very different regarding their supply characteristics (e.g. solar vs. wind). See Figure 2 (overview) and Table 2 (detail) for the current classification. In the following, we elaborate more on each class.

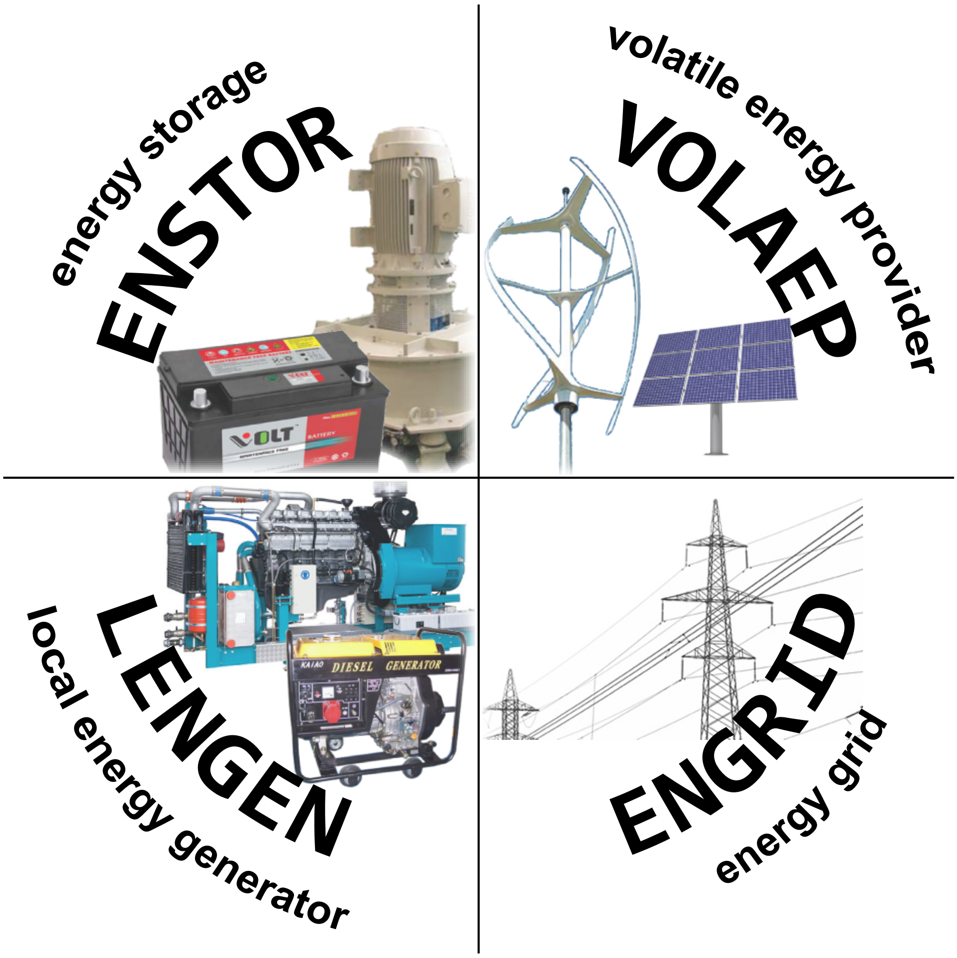

Figure 2: Overview of the classification of Local Energy Providers (LEP) in SmartCoDe

3.1 ENGRID (energy grid)

Fixed Parameters max power in, max power out

Data Provided energy supply tariff, feed-in-tariff

Commands Accepted none, but can receive a *load profile*: a forecast of the SmartCoDe network's power consumption, e.g. for the next 48 hours)

The ENGRID class represents a connection to a larger energy grid, usually a national electricity grid. The amount of power that this connection can supply is limited, although in real-world SmartCoDe applications this limit is very unlikely to be reached.

Table 2: Classification of Local Energy Providers (LEP) in SmartCoDe

Class	Abbrev. from	Description	Parameters			Examples
			installation	from LEP	to LEP	
ENGRID	energy grid	conventional energy provider	max power in, max power out	tariff inc. feed-in tariff	load profile	local electrical power provider
VOLAEP	volatile energy provider	energy source which depends on weather, daytime etc.	switchable (true/false)	supply forecast	on / off if switchable	wind turbine, water turbine, solar
ENSTOR	energy storage	energy source which has to be charged	storage capacity, max power in, max power out, round trip	charge level	power output (can be negative to indicate charging)	batteries
LENGEN	local energy generator	energy source which transforms some kind of fuel to energy	fuel price, max power out	fill level	power ouput	block power generator, diesel generator

The energy supplied from this node has a financial cost. The tariff for energy drawn from the grid can either be fixed, varying on a fixed schedule or varying "randomly" depending on the state of the national electricity market and the fixed contract. In all cases the ENGRID node must supply price forecasts to the SmartCoDe system, though these are trivial to compute in the fixed and scheduled pricing regimes.

A common business model in the energy market for large customers is to provide a load profile to the power supplier / grid provider. The customer is then charged depending on how well he complies to the load profile. Since the goals of the project include enabling smaller-scale customers (neighbourhoods, offices) to participate in the energy market, SmartCoDe will support the issuing of such load profiles. Most likely, these load profiles will already be available as an inherent byproduct of the Energy Management Process.

Some ENGRID nodes allow surplus energy to be fed back into the grid. The amount of power that can be fed back into the grid will have an upper limit. The price paid for this energy is usually different to the price of energy consumed. The feed-in energy price is usually fixed, but in future it could be varied dynamically in response to the state of the wider electricity network. SmartCoDe will allow ENGRID nodes to vary the feed-in price because this is the more general case. The nodes must therefore provide forecasts of the feed-in price as well as the consumption price.

It is likely that all SmartCoDe electricity networks will have exactly one ENGRID node. When the SmartCoDe system is operating, either in simulation or in the real world, it is important that the power flows within the system are balanced on a second-by-second basis. In networks with one ENGRID node the simplest way to do this will be to manage the power demand/generation of all nodes except for the ENGRID node. The ENGRID node will "automatically" supply or sink the correct amount of power to balance the loads on the system.

Networks with no ENGRID nodes are known as "islanded" networks. In these networks, balancing the power flows is much more difficult. Islanded networks will not be considered much in the SmartCoDe project, but the SmartCoDe infrastructure and energy management system could be of real benefit in such a network with only minor changes to the SmartCoDe Energy Management Unit. This will be tested using the SmartCoDe simulator.

This highlights an important design approach within SmartCoDe: making the majority of the nodes and infrastructure as generic as possible means that the system is adaptable to new scenarios by altering only the Energy Management Unit. The flexibility that this offers is important when the future will probably see many different types of local grid structure evolving.

3.2 VOLAEP (volatile energy provider)

Fixed Parameters switchable (true/false)

Data Provided forecast of power available over next 48 hours

Commands Accepted on/off if switchable

The VOLAEP class represents local generation sources which are not always able to provide the same amount of power. These include many renewable sources such as on-site wind turbines, hydro-electric plants and solar PV arrays. Under the SmartCode model, these sources of energy have no or very little incremental financial cost for generating power. They must provide the SmartCoDe network with a forecast of the amount of power that they will be able to produce at various times over the next 48 hours. This forecast must include the standard deviation of each prediction as a measure of how accurate the forecast is likely to be.

Unlike other generators, these nodes cannot be commanded to produce a certain level of power: instead the power output is determined by the environmental conditions and other factors. However, some of these nodes may accept a command telling them to switch off completely if their energy is not required.

3.3 ENSTOR (energy storage)

Fixed Parameters storage capacity, max power in, max power out, round trip efficiency

Data Provided charge level

Commands Accepted power output (will be negative to indicate charging)

Energy storage systems have a limited energy capacity and can be charged or discharged on demand at a given rate. There are maximum limits on the rate of charge and the rate of discharge. Losses mean that not all the energy used to charge the system is recovered when the system is discharged. The parameter that captures these losses is the "round-trip efficiency," defined as the fraction of the input energy that is returned to the grid during a charge/discharge cycle.

In reality these parameters are all variable and interact: for example, in some energy storage technologies charging at a fast rate may improve the round-trip efficiency. However, within SmartCoDe all energy storage nodes will have fixed parameters over the lifetime of the storage system.

It is assumed for now that using an energy storage system to supply or sink power does not have any financial cost other than the energy lost through inefficiency.

3.4 LENGEN (local energy generator)

Fixed Parameters fuel price, max power out

Data Provided fill level

Commands Accepted power output

The LENGEN class represents local energy generators such as diesel generators. These typically have a limited amount of fuel available, and the fuel has a relatively constant price per unit of energy generated. The amount of power that a LENGEN node can produce is limited by the size of the generator, but the power output can be varied in response to commands from the SmartCoDe EMU. LENGEN nodes cannot sink power.

Real-world generators take time to start generating after being switched on, typically a few minutes for diesel generators. It will be assumed that a SmartCoDe type system would be able to forecast demand adequately to switch these assets on such that their full capacity is available when needed. However, the details of how this is done will not be considered further in this project.

Combined heat and power (CHP) plants burn fuel to generate both heat and electric power. In reality, there is a limit to how much they can vary the amount of electricity produced independently from the amount of heat. In order to simplify the SmartCoDe optimisation process, these constraints will be ignored and any CHP plants will be modelled as independent heat and electricity sources for now.

4 Wind and solar power models

This sections describes mappings between the environmental conditions and the amount of power that wind and solar-PV systems will produce. These models will be used in the SmartCoDe network to predict the power output of wind turbines and photovoltaic generators. Another application is to provide realistic data on wind and solar power to the SmartCoDe simulation environment.

4.1 Wind turbine power prediction

The power output from a wind turbine varies dramatically with time. However, this variation is not random: the power output is a deterministic function of the weather conditions. If the windspeed, air density and other parameters are known, then the power output of a wind turbine can be accurately predicted.

Within SmartCoDe we are only required to forecast the average power output for a period of ten minutes or more; second-by-second power predictions are not required. Fortunately, ten minutes is also the recommended "burst period" for both meteorological reasons and because it is widely used throughout the wind industry [1]. This means that many turbine manufacturers provide "power curves" for their turbines based on the average values from ten-minute periods - exactly the information that SmartCoDe requires. This section therefore focusses on predicting electrical power output for a ten-minute time period.

This section presents a generic model for predicting the power output from a wind turbine. The model has three parts:

- An equation for the amount of kinetic energy in the wind flowing through the turbine
- A simple model for the efficiency of a turbine at converting this kinetic energy into useful electricity
- Models of actions that the turbine control system may take to affect the power output

These three parts can be combined into a "system power curve" mapping windspeed to power output, with optional corrections for air density and gustiness. The system power curve allows the power output to be predicted for a given set of weather condition without having to calculate the details of the turbine's operation every time.

4.1.1 Kinetic energy in the wind

The amount of kinetic power in the air flowing through a wind turbine is given by the following equation:

$$P_{wind} = \frac{1}{2}\rho A U_{\infty}^{3} \quad (1)$$

Where A is the turbine's swept area, ρ is the density of the air and U_{∞} is the windspeed.

The windspeed is the main factor affecting the amount of power available. The wind will vary during the ten minute period, but as a first-order approximation the power output from the turbine can be predicted using only the mean windspeed during the period, $\overline{U}_{10}$. The cubic relationship between windspeed and power means that this will underestimate the kinetic energy if the wind speed is variable ("gusty").

However, although gusty wind may in theory contain more energy than steady wind for a given $\overline{U}_{10}$, all wind turbines struggle to extract all the power from gusty wind to some extent. This is due to the turbine constantly having to adjust to the changing wind conditions. The effect that this has on the power output is different for every turbine, so the power output of a real-world turbine could be either higher or lower in gusty conditions compared to a steady wind with the same mean speed.

Equation 1 shows that the power in the wind is directly proportional to the density of the air. This in turn depends on the air temperature, pressure and humidity. The density of dry air can be calculated from the ideal gas law:

$$\rho = \frac{P_{atm}}{R \cdot T}$$

where P_{atm} is the atmospheric pressure, T is the temperature of the air in Kelvin, and R is the specific gas constant for dry air:

$$R = 287.05\ Jkg^{-1}K^{-1}$$

The air density at a wind turbine site will vary day-to-day depending on the weather conditions. Usual values are between 1.0 kg/m^3 and 1.4 kg/m^3, depending on latitude and altitude. The International Standard Atmosphere has a density of 1.225 kg/m^3 at sea level.

4.1.2 Control system actions

Strong winds happen relatively rarely, but the cubic relationship between windspeed and kinetic energy means that they could potentially give very high power outputs from a wind turbine as shown in Figure 3. However, if the turbine was built to cope with these very high power levels it would become very expensive despite the high power levels only occurring on a few days per year.

The solution used in almost all wind turbine designs is to intentionally reduce the power output of the turbine in strong winds. This "throttling" can be done using a variety of methods such as pitching (feathering) the blades, yawing the turbine or actively controlling the speed using the electrical system.

The aim is to extract maximum power from light and medium winds ("Region II"), and to cap the power at a constant level in strong winds ("Region III"). This is illustrated in Figure 3.

Extremely strong winds may cause excessive structural loads on a turbine even if the power is being limited, so most turbines have an absolute maximum windspeed above which they will shut down. This is known as the "Cut-out windspeed" and is marked as "(4)" in Figure 3.

In very light winds there is almost no energy in the wind. Most turbines enter a low-power standby state when the wind is below the "Cut-in windspeed" in order to reduce energy losses. This is marked as "(1)" in Figure 3. In practice the cut-in and cut-out decisions use information about the recent wind history as well as the current windspeed in order to avoid continuously starting and stopping in marginal conditions.

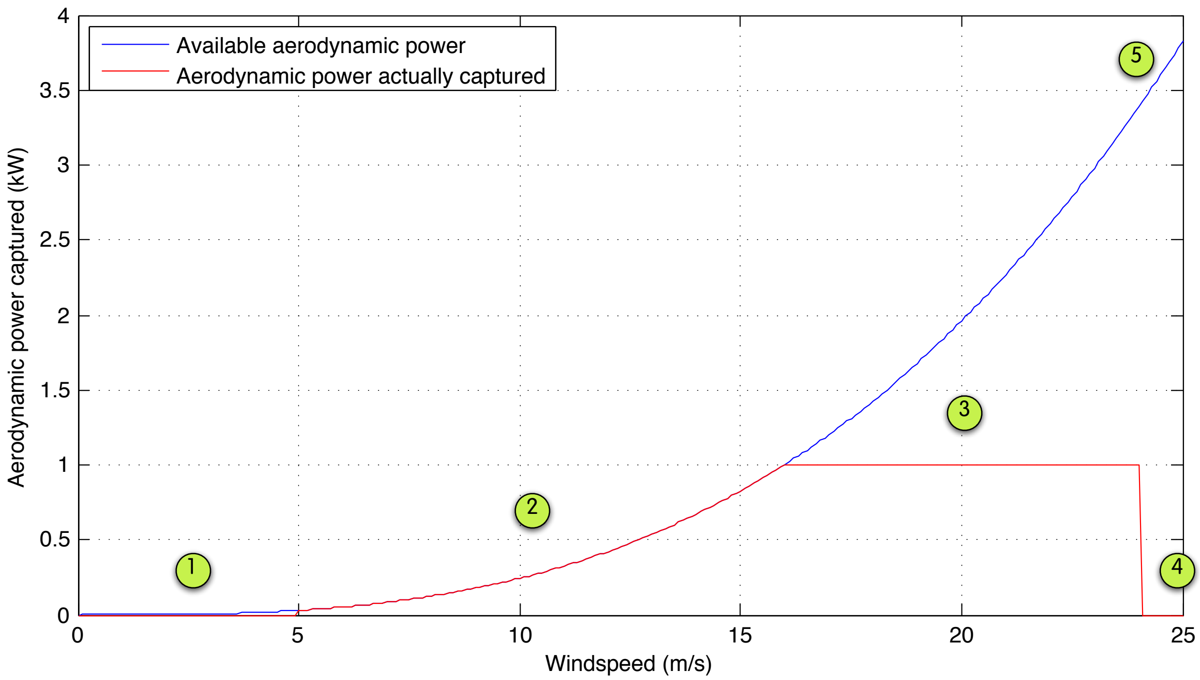

Figure 3: Operating regions for most wind turbines.

We require a prediction of the mean power output from the turbine over a ten minute period, which naturally depends on whether the turbine is running or not. When $\overline{U}_{10}$ is close to either the cut-in or cut-out windspeeds then the turbine could start or stop during the ten minute period, so will only be running for a fraction of the time. This "cut-in fraction" will depend on the algorithm used to make cut-in decisions as well as the exact wind profile preceding and during the ten-minute period.

However, the cut-in decisions for many turbines can be approximated as follows: during the ten-minute period, the short term average windspeed typically used for cut-in and cut-out decisions follows a Gaussian distribution around $\overline{U}_{10}$, with a typical standard deviation of around 2-10% of $\overline{U}_{10}$. The cut-in fraction can be estimated as the fraction of this Gaussian distribution which is above the cut-in threshold and below the cut-out threshold.

The throttling system used to limit power output in strong winds can be modelled using a very similar approach: the instantaeous windspeed follows a Gaussian distribution around $\overline{U}_{10}$, with a typical standard deviation of around 5-20% of $\overline{U}_{10}$. This can be multiplied by the red curve in Figure 3 and integrated to find the average power level.

Note that all the models presented in this report assume 100% availability. Predicting downtime due to faults or scheduled maintenance is outside the scope of SmartCoDe.

4.1.3 Efficiency

Equation 1 defines the amount of kinetic power in the air flowing through the rotor. However, the laws of physics limit the fraction of this power that can be extracted by even an ideal turbine: if all the kinetic energy was extracted from the air then it would become stationary and no more air would reach the turbine. Wind turbines typically extract around 40% of the energy from the air flowing through them.

The turbine converts the captured energy into mechanical and then electrical energy, before processing the electricity to produce a form suitable for connection to the local grid. Each of these stages have losses. A simplistic model that assumes that the energy conversion process has a fixed efficiency η and an additional fixed constant loss fits many turbines surprisingly well:

$$P_{out} = \eta \times P_{wind} - P_{overhead} \tag{2}$$

The overhead can be divided into two terms, the power consumed all the time (even when the turbine is idle in light winds) and the additional overhead incurred only when the turbine is spinning.

4.1.4 System power curve

The detailed models of wind turbine operation presented above can all be simplified into a single graph such as Figure 4. This "system power curve" maps the ten-minute-mean windspeed $\overline{U}_{10}$ to the average amount of electric power delivered, greatly simplifying the process of forecasting power output for the SmartCoDe network.

To generate the curves, each possible windspeed (and optionally air density and gustiness) is examined in turn. The models are used to predict the power output in each state, and the fraction of the time which the turbine is likely to spend in each state. The expected power output can then be calculated.

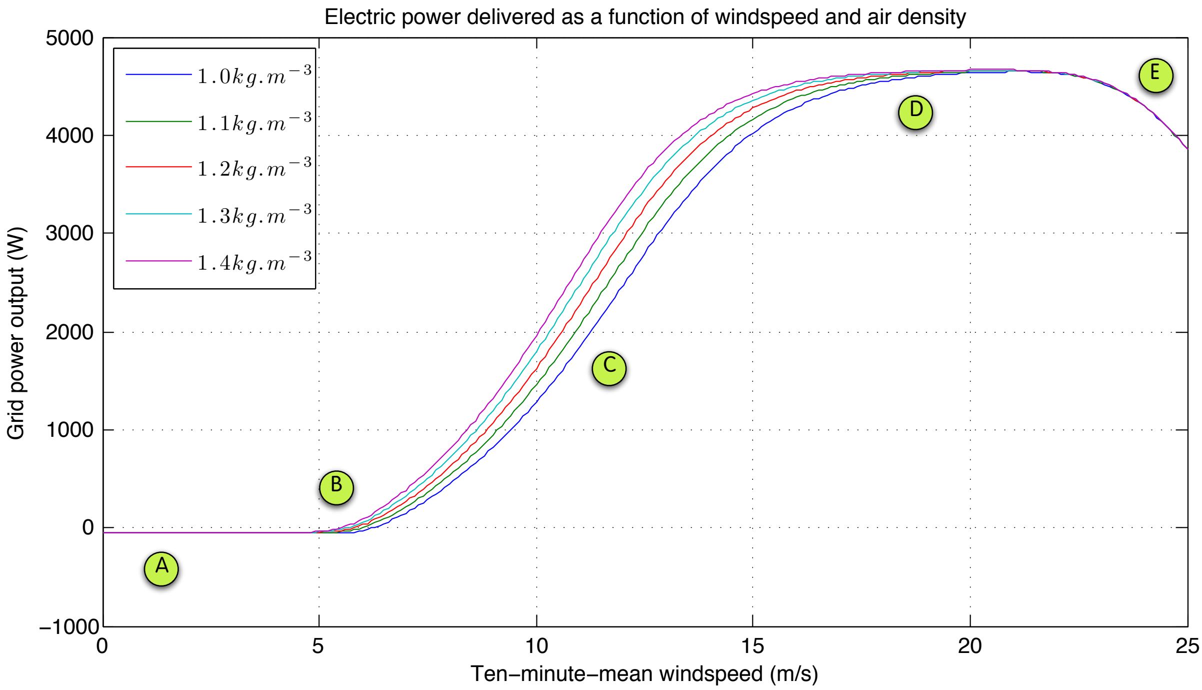

Figure 4: System power curves for a typical small wind turbine. They are based on average power output over a ten-minute period, so includes factors such as parking in light winds and throttling in strong gusts as well as the system's efficiency.

Each curve in Figure 4 has the following features of interest:

A Wind is too light, turbine is in standby state. Note that there is a small constant power consumption to power sensors etc.

B In this region the turbine is likely to be starting and stopping in light winds. The curve gives the power output averaged over ten minutes, taking into account the fact that the turbine may not be running all the time.

C Turbine is running in "Region II": maximising power output.

D Turbine enters "Region III" for at least some of the ten-minute period. This intentionally limits the power that is produced in order to control the loads on the turbine.

E With a ten-minute-mean windspeed of 25m/s, there is a chance that the "strong wind cut-out" conditions will have been triggered so the average power output is reduced.

Once a system power curve has been computed for a certain model of turbine, it becomes very simple to turn weather forecasts into predictions of power output. The difficult problem is that of producing the weather forecasts, specifically in predicting the exact windspeed at the turbine. This is a problem that is being addressed in detail later in the SmartCoDe project.

4.2 Solar PV power prediction

The power output from an array of photo-voltaic panels depends on many factors, which can be grouped into two categories:

- The amount of energy incident on the array:
  - The area of the panels in the array.

  - The angle of incidence of sunlight relative to each panel's surface. This depends on the latitude, time of year, local solar time, and the orientation of the panels, including the effect of any solar trackers fitted.
  - The amount, thickness and distribution of cloud cover.
  - Attenuation of the sunlight by passing through the earth's atmosphere, particularly when the sun is low in the sky.
  - Shading of any panels by structures in the environment (such as adjacent buildings) or by debris on the panels themselves.

- The performance of the array:
  - The nameplate efficiency of the panels.
  - The performance of the panels under diffuse lighting.
  - The air temperature, wind and cooling coefficient of the panels: PV panels generate heat but work better when cool.
  - The efficiency of the inverters, including the performance of any peak-power-tracking system.
  - Any imbalance in power output between panels within a string (can dramatically reduce power output from that string).

This shows that producing a detailed generic model suitable for all Solar PV systems is a large task! However, it can be considerably simplified if we use a similar "system power curve" to that described for the wind turbines above. In the case of a solar PV system the system power curve could map the insolation (watts of sunlight energy striking each square metre of PV panel) to the amount of "useful" (i.e. grid-compatible) electric power that is produced. This curve is relatively simple to measure for a given installation and hides many of the above parameters, making a detailed model of them all unnecessary.

Figure 5 shows an example of how this system power curve could be used to predict the power produced from a solar PV array. Geometric calculations are used to find the angle at which the sun will strike the array based on the position of the sun in the sky and the orientation of the panel. Then the effect of the predicted cloud cover is computed, giving the actual insolation, which can then be used with the system power curve to find the power output. If necessary, a family of power curves could be plotted for different air temperatures to improve the accuracy of the power forecasts.

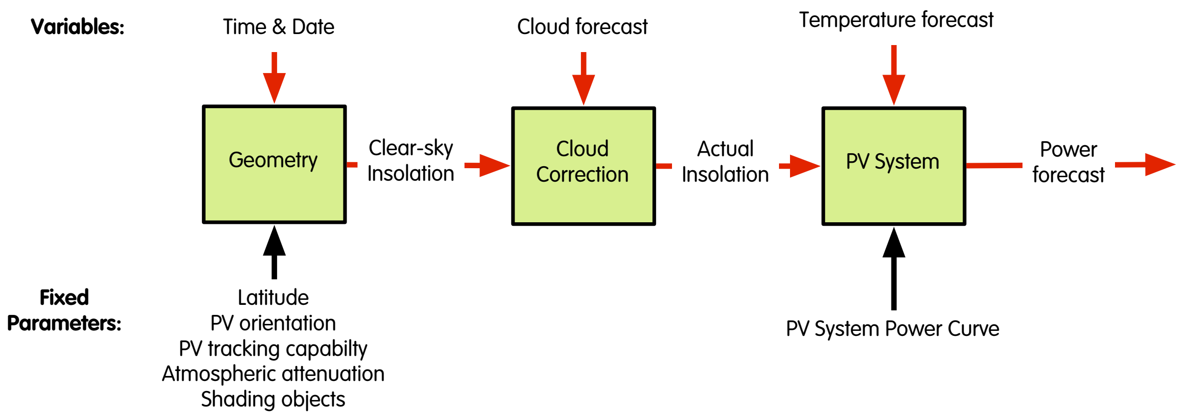

Figure 5: Outline of generic PV model mapping weather forecast to a predicted power output.

The model presented in Figure 5 is not perfect. In particular, it fails to capture the highly non-linear effects of partial shading on the power output, or the difference between diffuse and weak sunlight. However, the only way to accurately capture these effects would be to build a full computer model of all the cells in the array, predicting the voltage and current output from each one and simulating the actions of the various controllers. This is probably unnecessarily complicated for power forecasts within a SmartCoDe VOLAEP node, although the SmartCoDe interfaces are sufficiently general to allow such a model to be used if desired.

4.3 Probabilistic model inputs

The previous sections describe mappings between the environmental conditions and the amount of power that wind and solar-PV systems will produce. These models would be used in a SmartCoDe network to predict the power output of these nodes. However, the weather forecasts used to drive these predictions are always uncertain to some extent. The SmartCoDe network requires the power output forecasts to be accompanied by a measure of uncertainty, typically in the form of a variance estimate. This means that we have to propagate the uncertainty of the weather forecast through the model and into the power estimates.

The models presented above are deterministic, meaning that they do not add any uncertainty of their own. However, they are often highly non-linear so conventional linear and locally-linear techniques for propagating uncertainty are unlikely to give accurate values of the variance in the power output values. They are also often not analytically tractable (or based on tables of measured data) making it difficult to calculate all the partial differential terms.

One option for avoiding these problems is to propagate the uncertainty covariance using an "unscented transform" [3]. This uses a deterministic sampling process to produce a set of "sigma points" representing the input distribution. Each point represents a possible combination of weather variables, and together they capture the probability of different weather occurring. These points are then individually propagated through the model, each producing a predicted power output. The variance of these power predictions gives a good approximation to the true uncertainty in the predicted power output.

The unscented transform works best with smooth continuous probability densities: if the input distribution does not have these properties then a Monte-Carlo sampling approach would potentially capture the distribution better. However, given that only the first two moments (mean and variance) of the output distribution will be used, it is probably unnecessary to capture the shape of distributions in great detail. The unscented transform is much more efficient than a Monte-Carlo sampling approach as it only requires a few carefully-chosen points rather than many points chosen at random.

The predicted outputs can be compared with reality and the differences used to enhance future predictions through Kalman filtering or similar.

5 Operation of the SmartCoDe network

The basic operation of the SmartCoDe network is straightforward to describe:

- The EMU gets information on power supply from the LEPs (e.g. grid tariff, supply forecast for wind turbine)
- It also gets information on power consumption by the EuPs
- The EMU controls the EuPs and LEPs in the network using this information with the goal of energy saving / cost reduction / load balancing /CO_2 reduction.

How the "using this information" part works will be determined in Task 1.4, but one idea is to control the network to match a pre-determined "load profile" of power vs. time. This load profile might be issued by the customer to the grid operator as described in Section 3.3, but could also be issued by the grid operator implicitly via time-dependent tariffs. It might even change over time due to dynamic tariffs and / or automated negotiations by both parties. Assuming that a network load profile is sent to the grid therefore covers many existing and potential business models. Different optimisation goals can be formulated in terms of target load profiles, for example optimizing for cost reduction might result in different load profiles to optimizing for energy saving. Figure 6 shows an overview of this scenario.

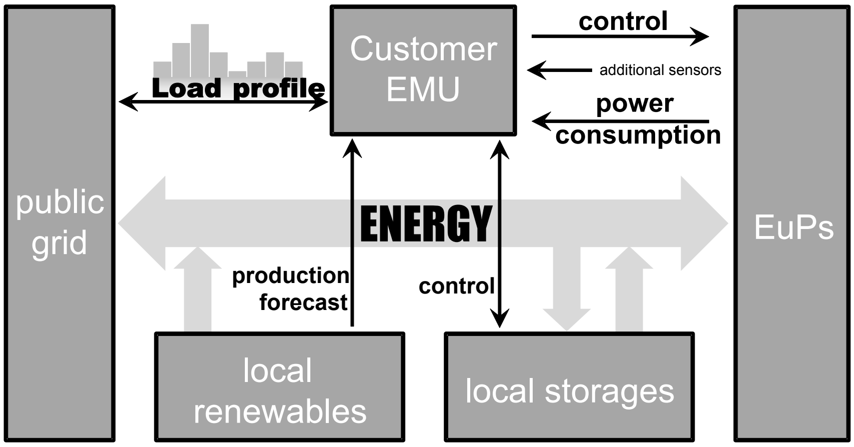

Figure 6: Information and Energy flow in the SmartCoDe network

Regarding the question of how the EMU controls the EuPs, there are basically two management approaches: A direct centralised approach and an indirect decentralized approach.

5.1 Centralized versus decentralized approach

In a **centralized approach**, the EMU would control every EuP in the network directly. That is, it would tell every EuP to switch on, switch off, or go into some level in between (e.g. dimming). The SmartCoDe nodes would merely be subordinates in this setting, simply passing on the command issued by the micromanaging EMU to its attached EuP (see Figure 7), while also passing on the sensor data and the power consumption data from the EuP to the EMU.

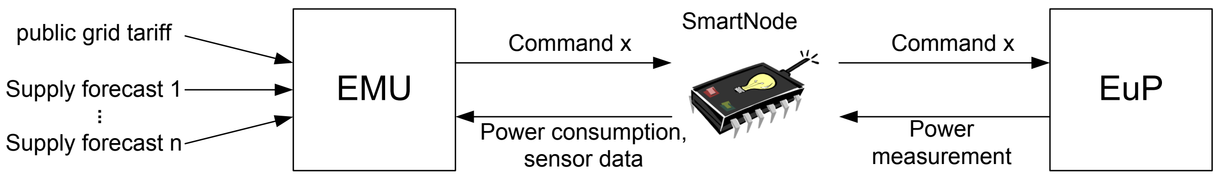

Figure 7: Principle of a centralized SmartCoDe operation

Advantages:

- In terms of communication, this case could be covered with the standard ZigBee profiles
- The design of the SmartCoDe node would therefore be simpler
- All the power management decisions are made in the EMU. Therefore, interdependencies in the network can be considered more easily since the EMU has complete knowledge and control.

Disadvantages:

- Huge communication overhead for certain applications, since every single EuP is micromanaged, and also the EuP's sensor values have to be transmitted. Above a certain number of SmartCoDe nodes in the network we might get serious bandwidth problems.
- Above a certain number of managed EuPs, the EMU might not be able to handle the workload of controlling them all individually.
- For some of the EuP classes (VARSVC, VSTSVC), the SmartCoDe operation will basically constitute a control loop. It is unclear whether the remote wireless nature of this control loop might cause problems.

- If the EMU crashes, or the communication to the EMU is corrupted, the EuPs are basically headless, which is also a serious issue regarding the aforementioned control loops.

In a **decentralized approach**, the EMU would not control every EuP directly, but would issue (for example via broadcasts) general demand side management directives, which the SmartCoDe nodes would use to decide autonomously how to control their attached EuP. A possible form for these directives would be an abstract *cost function* for a certain time interval into the future which is computed by the EMU out of the grid tariff info, supply forecasts issued by the LEPs (see Figure 8), or additional information like consumption forecasts out of a local user behaviour data base.

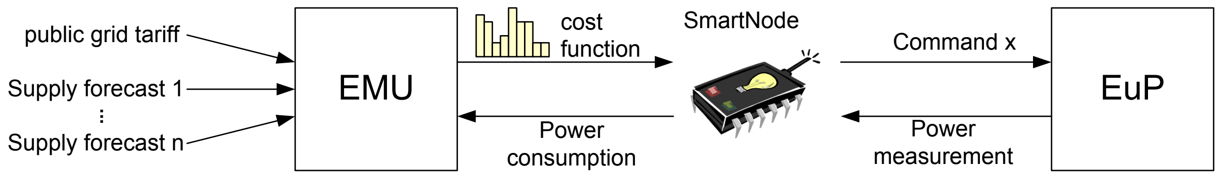

Figure 8: Principle of a decentralized SmartCoDe operation

Advantages:

- Considerably less communication overhead:
  - No need to send EuP sensor data to the EMU, although additional sensor data like outside temperature or wind speed might still be processed
  - The cost functions would need to be transmitted only if there is a change in the underlying forecasts. While we currently think about issuing cost functions every ten minutes, once every hour might be enough.
- Control loops are local and not vulnerable to EMU crash or communication breakdown to EMU.
- The EMU needs to do less work since it does not need to address (or even know) every EuP in the SmartCoDe network individually as in the centralized approach. This makes the energy management in the decentralised approach much more applicable to larger networks.
- Micromanaging is still possible: For certain EuPs, the decentralized approach might make not that much sense or the SmartNodes might not be able to handle the computations involved. A certain number of such EuPs can still be handled like in the centralized approach.

Disadvantages:

- The SmartCoDe node design might be more challenging regarding the software
- Computing the abstract directives might be pretty expensive and/or challenging. For example there might be NP-complete problems involved, as this is often the case in optimization.
- While the EMU does not need to consider every EuP in the network explicitly any more, it also can't account for interdependencies as easily, e.g. with respect to load balancing. This also makes it difficult to guarantee stability.

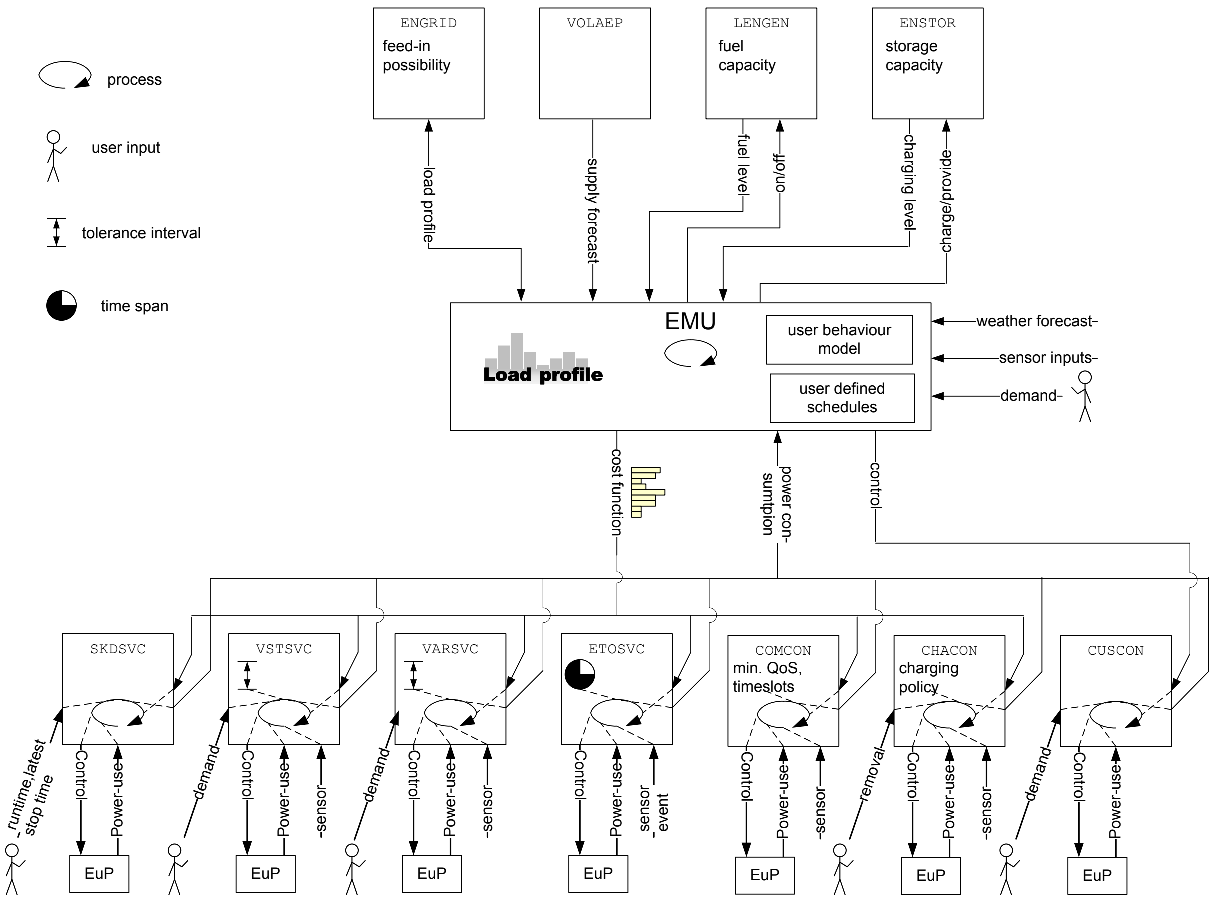

Figure 9: SmartCoDe network overview

Altogether, this approach seems to be more advantageous: The problems involved might be challenging, but seem to be solvable. The centralized approach, on the other hand, presents some serious, probably not solvable problems regarding the communication overhead and the management of a large number of EuPs. Figure 9 shows an overview of the decentralised approach with all classes of EuPs and LEPs discussed so far. It also shows that the user interaction can handled locally pretty well and therefore does not contribute to the network traffic.

Some details have been omitted from Figure 9 for clarity. For example, the various internal parameters of the respective EuPs like the tolerance intervals of VSTSVC and VARSVC can be mostly manipulated by the EMU.

The choice of approach has to be finalised in Task 1.4. Workpackage 2 has provided a simulation environment that will allow investigation of several aspects of the decentralised approach (see D-2.2).

6 Conclusion and future work

This report presents the current status regarding modelling the energy consumers and producers in a SmartCoDe network. Regarding Energy using Products (EuPs), a classification has been developed which is based on the nature of the EuPs service, its interfaces and its characteristics regarding energy management. However, to capture all aspects of EuPs relevant for SmartCoDe, user behaviour has also considered; how much so has yet to be determined.

The Local Energy Providers (LEP) classification is more complete in this sense since user interaction plays virtually no role here. Detailed models for the energy output of wind turbines and photovoltaic generators were presented. They will be used both in simulation and to create real-world energy forecasts for use in energy management algorithms.

Some thoughts on the general communication infrastructure and energy management approach have been presented. Currently, a decentralised approach is favoured where the Energy Management Unit (EMU) issues cost functions with the goal to keep the overall network power consumption within a given load profile. The details will be finalised in Task 1.4.

References

[1] T. Bertényi. Optimal burst period for energy forecasting. Technical report, quietrevolution technical note, 2010.
[2] Deutscher Verein des Gas- und Wasserfaches e.V. Trinkwassererwärmungs- und Trinkwasserleitungsanlagen; Technische Maßnahmen zur Verminderung des Legionellenwachstums; Planung, Errichtung, Betrieb und Sanierung von Trinkwasser-Installationen. Technical report, Arbeitsblatt W 551, 2004.
[3] S. J. Julier and J. K. Uhlmann. A new extension of the kalman filter to nonlinear systems. pages 182–193, 1997.
[4] H. Kim and K. G. Shin. Scheduling of battery charge, discharge, and rest. In *Proceedings of the 2009 30th IEEE Real-Time Systems Symposium*, RTSS '09, pages 13–22, Washington, DC, USA, 2009. IEEE Computer Society.
[5] F. Kupzog and C. Roesener. A closer look on load management. In *Proceedings of the 5th IEEE International Conference on Industrial Informatics, 2007*, pages 1151–1156. IEEE Computer Society, IEEE Computer Society, 2007.

Abbreviations and Definitions

A	Wind turbine's swept area
CHACON	charge control
COMCON	complete control
CUSCON	custom control
EMU	Energy Management Unit
ENGRID	public energy grid
ENSTOR	energy storage
ETOSVC	event-timeout controlled service
EuP	Energy Using Product
HVAC	Heating, Ventilating and Air Conditioning
K	Kelvin
LENGEN	local energy generator
LEP	Local Energy Provider
Local Grid	The part of the SmartCoDe Network which lies behind the metering point of the public grid power supplier
ρ	density of the air
P_{atm}	atmospheric pressure
PC	Personal Computer
public grid	Network and power supply before the costumers metering point
PV	Photovoltaic
R	specific gas constant
SKDSVC	schedulable service
T	Temperature
U_∞	Windspeed
VARSVC	variable service
VOLAEP	volatile energy source
VSTSVC	virtual storable service