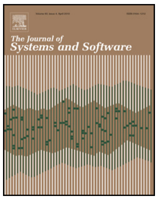
The Journal of Systems and Software xxx (2013) xxx–xxx

Contents lists available at SciVerse ScienceDirect

The Journal of Systems and Software

journal homepage: www.elsevier.com/locate/jss



# The social smart grid: Dealing with constrained energy resources through social coordination

Florian Skopik*

*AIT Austrian Institute of Technology, Safety and Security Department, 2444 Seibersdorf, Austria*

ARTICLE INFO

*Article history:*
Received 5 September 2012
Received in revised form 18 April 2013
Accepted 19 April 2013
Available online xxx

*Keywords:*
Social smart grid
Energy consumption balancing
Community-driven energy sharing

ABSTRACT

The smart grid promises to improve the efficiency and reliability of tomorrow's energy supply. One of the biggest achievements of future smart grids will be their distributed mode of operation which effectively eliminates vulnerable nodes causing single points of failures in the grid. However, due to the lack of centralized energy production and control, the coordination of energy consumption becomes first priority. Because there do not exist technologies to store energy at large-scale yet, all energy that is required must be produced at the same time. The biggest challenge of energy producers is therefore to reliably predict and provide the right amount of required energy to avoid shortages and breakdowns. In this paper, we propose a novel way to let smart grid stakeholders, i.e., energy producers and consumers, coordinate their energy demands themselves. For that purpose we combine traditional social network models and service-oriented computing concepts with the smart grid to allow consumers to form communities according to their energy consumption behavior. These communities enable them to interact with other grid stakeholders to coordinate energy consumption plans and set up private energy sharing alliances. This way, the utility provider and industrial energy producers can rely on a better predictable and a smoother energy demand of customers. We introduce a software framework, making use of widely adopted standards, demonstrate its feasibility with an agent-based simulation, and discuss its overall applicability.

## 1. Introduction

The electric power grid is by far the most important technical infrastructure used today and the basis for modern life. It is fundamental to all modern networked services, such as telephone, television, or the Internet. With the emergence of electric cars, the power grid will also ensure our mobility and thus increase its role. Reliable, dependable and secure energy supply is thus of paramount importance not only for the industry, but for the whole society. Changing requirements on the power grid in terms of supply capacity, load conditions, and reliability lead to an ongoing modernization and a major shift from a static public power grid to a more flexible one that can cope with today's challenges. The *Smart Grid* initiative aims at advancing the traditional power grid to an intelligent utility (Massoud and Wollenberg, 2005). As defined in ERGEG (2010), *a smart grid is an electricity network that can cost-efficiently integrate the behavior and actions of all users connected to it – generators, consumers, and those that are both – to ensure an economically efficient, sustainable power system with low losses and high levels of quality and security of supply and safety*. There are numerous advantages which come with the smart grid compared to the traditional power grid, such as remote meter reading, fast failure detection and recovery, intelligent and prioritized energy distribution, and the integration of home solar panels and private wind mills (so called micro-producers).

One of the biggest achievements of future smart grids will be their distributed mode of operation which effectively eliminates vulnerable nodes causing single points of failures in the grid. However, due to the lack of centralized energy production and control, the coordination of energy consumption becomes first priority. Because there do not exist technologies to store energy at large-scale yet, all energy that is required must be produced at the same time. The biggest challenge of energy producers is therefore to reliably predict and provide the right amount of required energy in the network to avoid shortages and breakdowns, e.g., by making use of pumped storage hydro power stations. In this paper, we propose a novel and innovative way to let smart grid stakeholders, i.e., energy producers and consumers, coordinate their energy demands themselves. For that purpose we combine traditional social network models and service-oriented computing concepts with the smart grid to allow consumers to form communities according to their energy consumption behavior. These communities enable them to interact with other grid stakeholders to coordinate energy consumption plans and set up private energy sharing alliances. This way, the utility provider and industrial energy producers can rely on a better predictable and a more smoothened energy

---

* Tel.: +43 664 8251495.
*E-mail addresses:* florian.skopik@ait.ac.at, florian.skopik@gmx.at

0164-1212/$ – see front matter © 2013 Elsevier Inc. All rights reserved.
http://dx.doi.org/10.1016/j.jss.2013.04.052

Please cite this article in press as: Skopik, F., The social smart grid: Dealing with constrained energy resources through social coordination. J. Syst. Software (2013), http://dx.doi.org/10.1016/j.jss.2013.04.052

demand of customers. Furthermore, being part of communities should strengthen their energy awareness by allowing comparisons of energy consumption data with other community members.

We propose a **social overlay network model** which conceptually resides on top of the smart grid infrastructure. This concept employs a wide variety of social networking approaches and service-oriented architecture (SOA) techniques, such as dynamic discovery of alliance partners, flexible interactions, run-time setup of contracts and agreements on delivered services, and opportunistic utility assessment (Skopik et al., 2011) to support wide-range interoperability and scalability. Furthermore, establishing security is an essential objective of this layer. This ensures the application of our model in a variety of novel community-driven use cases. In particular, the paper deals with the following contributions:

- *Smart grid social overlay and community formation model.* These mechanisms enable innovative application use cases by allowing personal interactions between smart grid stakeholders which is the basis for active energy consumption coordination. We adopt a popular community formation algorithm, shape it to be applied in the context of smart grids, and demonstrate its feasibility and an example configuration for smart grid communities.
- *SLA model for energy sharing communities.* We highlight a SOA-based model for contracts and service level agreements (SLAs) between community members. We further put particular emphasis on the application of secure protocols, such as the Web of Trust ontology, to protect the privacy of all participants and to facilitate the adoption of the presented concepts by end-users.
- *SOA architecture and evaluation.* We present a Web services-based prototype architecture and evaluate its feasibility in complex socio-technical environments, such as large-scale smart grid communities. In particular, we set up and run a complex agent-based simulation to show the potential of our proposed energy consumption coordination and sharing approach.

The rest of the paper is organized as follows. Section 2 highlights the motivation for our work and states the benefits of a social overlay network as well as future use cases. Section 3 introduces a conceptual layer model and the theoretical background, i.e., the application of a social formation algorithm and the definition of mutual sharing agreements between consumers. An architecture and prototype implementation, including used protocols and Web services standards, is discussed in Section 4. Section 5 deals with an evaluation of basic features for most use case scenarios and discussions on scalability of the prototype platform. Here, we also highlight the findings of an agent-based simulation. Related work is outlined in Section 6 and finally, Section 7 concludes the paper.

## 2. Motivation and use cases

Major goals of the smart grid initiative include (i) reducing costs (billing, accounting), (ii) increasing reliability through fast failure detection and recovery, as well as active load balancing, and (iii) reducing energy consumption. Since traditional energy consumers can switch their roles and become (at least for short periods) energy producers, managing the smart grid becomes challenging.

### 2.1. The smart grid stakeholders

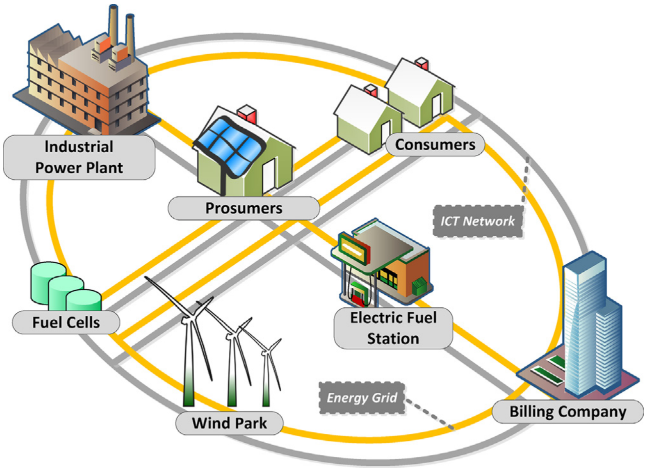


**Fig. 1.** Conceptual structure of a smart grid and its stakeholders.

In the future power grid, consisting of an energy grid and a tightly coupled ICT network as depicted in Fig. 1, we distinguish at least between the following stakeholders: (i) *energy producers* run various kinds of power plants; (ii) *energy consumers*, typically households and industry, as well as public facilities (e.g., hospitals) use energy; and (iii) the *electric utility* provides and maintains the public power grid for energy distribution and thus connects energy producers and consumers. Further separate organizations might be involved for managing meter readings, accounting and billing processes. An important aspect is that traditional consumers may integrate privately owned micro plants into the grid, e.g., wind or gas turbines as well as solar panels, and can become temporary producers. In that case, they are referred to ‘prosumers’.

### 2.2. What a social network can contribute

In the future power grid, we distinguish at least between the following stakeholders: (i) *energy producers* run various kinds of power plants; (ii) *energy consumers*, typically households and industry, as well as public facilities (e.g., hospitals) use energy; and (iii) the *electric utility* provides and maintains the public power grid for energy distribution and thus connects energy producers and consumers. Further separate organizations might be involved for managing meter readings, accounting and billing processes. An important aspect is that traditional consumers may integrate privately owned micro plants into the grid, e.g., wind or gas turbines as well as solar panels, and can become temporary producers. In that case, they are referred to ‘prosumers’.

We argue that strong and dedicated communities (Chima, 2011) considerably contribute to reaching ambitious energy saving goals. A social network supports social campaigns, discovery of reliable (energy sharing) partners, and interactions with other people. This is essential to create a sense of belonging and thus motivate people to act reliably and responsibly. There are various ways in which a social network which overlays the power network, will be beneficial:

- *Coordination of power consumption*: Allowing users to coordinate their energy consumption can help to balance energy consumption from a temporal point of view. For instance, consuming power outside of peak hours can be rewarded by energy providers. However, reliable active coordination requires people to announce their energy consumption plans centrally, which compromises privacy and introduces a major security threat. So, how can people discover trustworthy potential partners for distributed coordination?
- *Establishing a marketplace for privately generated energy*: With the increasing number of wind turbines and solar panels in the home area, households become micro power plants. Typically one will consume his own energy, however, in some situations s/he might produce more than needed, e.g., if not at home. In that case people can feed back energy into the public power grid. However,

Please cite this article in press as: Skopik, F., The social smart grid: Dealing with constrained energy resources through social coordination. J. Syst. Software (2013), http://dx.doi.org/10.1016/j.jss.2013.04.052

creating a community to enable direct selling of energy is beneficial for both, the consumer and the producer, who can negotiate individual conditions.

- *Providing a platform for energy traders*: In a more advanced scenario, two individuals might agree on utilizing each others' privately owned energy sources (e.g., *i* and *j* combine their produced solar energy, while *i* is using generated energy at 8 am and *j* at 9 am). However, in order to set up such an agreement they do not need to be neighbors, but can use the public power grid for transportation purposes (similar to public distributed computing, where personal computers are connected through the Internet to solve complex tasks while their particular owners do not utilize them). Even the infrastructure provider benefits from such agreements, since load capacities become predictable and there is no need to estimate future peak loads.
- *Enabling cooperative energy storage*: With an increasing number of electric cars on the market, everyone can get a dense energy storage for his home (Ipakchi and Albuyeh, 2009). However, cars are not needed every day and at peak hours their owners may allow others (or the central energy provider) to consume power from car batteries (at least partly) in order to avoid power blackouts in often unexpected generally high power consumption situations. Here, a social network alleviates the active coordination and setup of rules for energy abstraction within a community.
- *Strengthen energy consumption awareness*: This objective can be addressed through a multitude of initiatives. Social campaigns teach the public the wealth of energy, amount of produced carbon, and efficient power saving opportunities. Online platforms enable customers to compare their energy consumption behavior with others, and energy saving competitions (as hosted already today by OPower Company (2011)) attract people to actively participate.

### 2.3. Illustrative scenarios

Having these advantages in mind, we describe some illustrative scenarios and related challenges that we will address in the rest of this paper.

#### 2.3.1. Use case 1: distributed energy storage

One major drawback of today's energy supply is that electricity cannot be efficiently stored at large scale. As a matter of fact, electrical energy should be consumed right away when it is produced. Thus, predicting customer's consumption behavior is essential for energy providers in order to cope with peak loads. Recent research investigates novel concepts to relax this situation. With the anticipated broad acceptance of electrical cars, car batteries can be utilized as buffers. *Challenge.* Customers will have to specify if and under what conditions they allow abstraction of electricity from their cars' batteries which is fed back into the public grid on demand. In order to do that, they need to negotiate and set up service level agreements (SLAs) (Ludwig et al., 2003) with their power supplier which formalize and regulate this process.

#### 2.3.2. Use case 2: energy marketplace

With the recent emergence of private energy generators traditional energy consumers can temporarily become energy providers in the smart grid. For that purpose, people own energy generators, whose electricity they sell or share if not needed. Finally, they might want to share energy with some particular individuals only, e.g., building an 'energy alliance' with their neighbors, and like to specify special conditions for them. That case would need a supporting social network, i.e., a *marketplace*, where social links reflect trustworthy relations used to model the aforementioned SLAs in an intuitive manner. *Challenge.* We argue that the application of social networking has a similar potential as mechanisms described in the previous use case to effectively distribute energy and avoid peak loads. However, in order to allow distribution of self generated energy through a marketplace, people need the ability to discover potential energy sharing partners, and auction (temporarily) excessive energy.

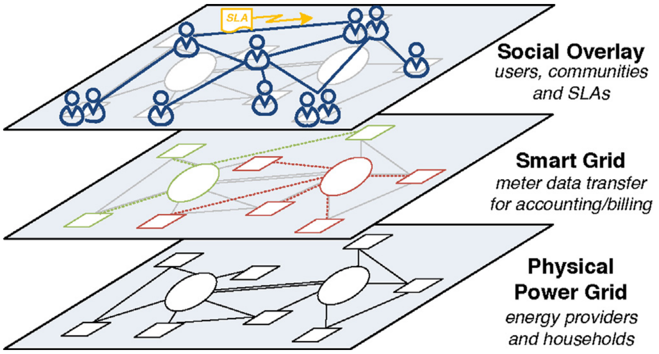


**Fig. 2.** Conceptual framework for smart grid social overlay networks.

#### 2.3.3. Use case 3: community-driven energy saving and consumption regulation

Both, active competition in a community as well as teaming up with people having same interests supports reaching ambitious goals. Energy saving competitions between single individuals or teams (e.g., groups having similar demographic background OPower Company, 2011) actively motivate community members to save energy. Furthermore, reaching a harmonic level of energy consumption with respect to the time of day through active coordination between energy consumers is a further goal. For that purpose people can populate their energy consumption plans and schemes (e.g., charging car batteries after coming home from work, heating up electric sauna on Friday evening) and get therefore rewarded with better price conditions by energy producers. *Challenge.* A critical mass of users needs to be attracted by a platform to make the whole concept taking off. The fundamental scientific concept is the application of coalitional game theoretic approaches (Osborne and Rubinstein, 1999) where the benefit for each individual increases through active collaboration.

## 3. Foundational models

Various components from the social networking domain, social formation algorithms, and contract negotiation models are required to realize the aforementioned use cases. These mechanisms reside on top of the actual smart grid infrastructure.

### 3.1. Social overlay network model

In order to establish a suitable social overlay, we propose a layered conceptual model as depicted in Fig. 2. This model consists of:

1. *Physical power grid*. The physically static infrastructure connects the energy producers (virtually every type of power plant) and energy consumers (households, industry, etc.). This layer represents a simplistic view on the state of the art.[1]
2. *Smart grid*. The smart grid deals with automatic meter reading (AMR), the ability of customers to integrate their own power

---

[1] We neglect all other complex grid mechanisms that are not related to the communication network of the smart grid; such as control networks for large-scale load balancing through the utility company.

Please cite this article in press as: Skopik, F., The social smart grid: Dealing with constrained energy resources through social coordination. J. Syst. Software (2013), http://dx.doi.org/10.1016/j.jss.2013.04.052

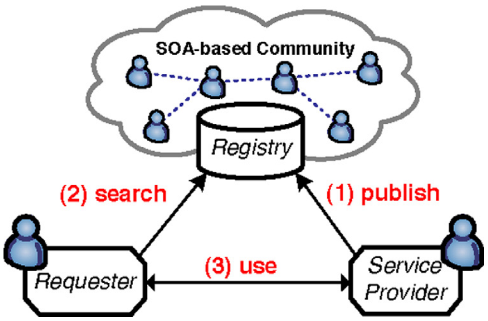


**Fig. 3.** Service-oriented architectures triangle.

sources into the public grid, and easy access to a liberal energy market, where customers can change their provider virtually instantaneously.

3. *Social overlay network*. Strong communities enable more sophisticated application scenarios, such as aforementioned marketplaces, social campaigns, and (perhaps most importantly) support to increase each individual's energy awareness. The focus of this paper is on the social overlay network layer.

### 3.2. SOA-based discovery

Web services play a fundamental role in supporting flexible collaboration and formation scenarios. The traditional 'SOA-triangle' approach (Alonso et al., 2004), enables a *requester* (client) to discover a *service* flexibly at run-time by querying a *service registry*. We adopt this concept and apply it to the social overlay of the smart grid. Here, requesters (e.g., of energy sharing opportunities, social campaign setups, etc.) can query for service providers (e.g., someone who provides energy from his own micro turbine) by querying the social network. By following the SOA paradigm, three essential steps are performed (see Fig. 3):

1. *Publish*. Users have the ability to create services and publish (announce) them within the community network using a registry. Publishing a service is as simple as posting a blog entry on the Web. It is the association of the user's profile with a service description (WSDL interface (Alonso et al., 2004)). Interfaces provide the needed metadata support for the discovery of suitable services.
2. *Search*. The service requester performs a criteria-based search (e.g., reflecting the energy demand and valid time slots) to find services. Ranking is performed to find the most relevant service based on, for example, the degree of matching or community feedback of a user-provided service.
3. *Use*. The framework supports automatic user interface generation using XML-Forms technology.[2] This way the details of an SLA are set up, for instance, the amount of provided energy and respective compensation for using the service.

### 3.3. Community formation model

#### 3.3.1. Basic model

Users of the smart grid can join and form communities based on their demographic background, interests and needs. We adopt[3] a well-known model for group formation from economic sciences (Jackson and Wolinsky, 1996; Jackson and Watts, 1999). The basic properties of the original use cases are similar to the situation in the smart grid. In particular, self-interested individuals accounting for their own payoff can create and cut links to others based on dynamically changing requirements. Periodic re-evaluation of a cost-benefit ratio exposes the network to constant flux and change.

**Table 1**
Description of symbols.

| Symbol | Description |
|---|---|
| $G$ | Social network $G$ with segments $g \in G$ |
| $u_i(g)$ | Utility of $i$ obtained from network segment $g$ |
| $ij$ | Direct link from node $i$ to node $j$ |
| $t(ij)$ | Path length from node $i$ to node $j$ |
| $\delta_j$ | Payoff committed by $j$ to neighbors (e.g., $i$) |
| $c_j$ | Costs (e.g., of node $i$) caused by node $j$ |

Each user is represented by a node in a graph-theoretical model $G = (N, E)$, (Wasserman and Faust, 1994) consisting of nodes $N$ and edges $E$. If a node $i$ is connected to a node $j$, we denote this edge with $ij$. Each node $i \in N$ receives a utility $u_i(g)$ when participating in group $g \in G$. In detail, $u_i(g)$ is calculated by accounting for the payoff $\delta\,(0 \le \delta \le 1)$ $i$ receives for being connected to other agents, and cost $c\,(0 \le c \le 1)$ for maintaining a link. Since nodes can be connected via several hops (transitive links), $t(ij)$ represents the number of edges on the shortest path from $i$ to $j$. The final algorithms is formulated as given in Eq. (1).

$$u_i(g) = \sum_{j \neq i} \delta^{t(ij)} - \sum_{j:ij \in g} c \tag{1}$$

#### 3.3.2. Personalized model

Although the model is widely applied in various works, there are some specifics that need to be considered and require slight adaptations. First, the payoff $\delta$ of a connection from one member to another member relies on a set of different factors. For instance, when sharing energy, the covered amount of the whole demand is a basic factor to rate the benefit of a social relation. Since the degree of coverage is different among members, we need to personalize $\delta$ for each particular neighbor by considering specific SLAs that are set up between pairs of individuals and enabled transitive relations. Second, costs emerge for each individual based on neighborhood size, i.e., the number of connections to maintain. These costs typically reflect coordination effort, such as maintaining SLAs (re-negotiating and discussing conditions of active SLAs), answering individual requests, and monitoring a partner's reliability (at least roughly, because this is supported by SOA monitoring techniques). Thus we slightly revise Eq. (1) and apply the model as given in Eq. (2). Symbols are explained in detail in Table 1.

$$u_i(g) = \sum_{j \neq i} \delta_j^{t(ij)} - \sum_{j:ij \in g} c_j \tag{2}$$

Links and model variables as well as their application are further visualized in Fig. 4. Social relations are established due to various reasons, for instance, typically before starting sharing energy. For that purpose, discovered community members (using SOA based service discovery mechanisms as discussed before), who promise highest benefit are linked as partners. After that, both parties

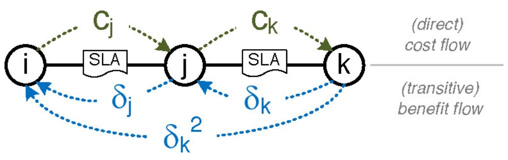


**Fig. 4.** Flows from $i$'s view in community formation processes.

---

[2] XML Forms: http://www.w3.org/MarkUp/Forms/.

[3] Notice, we slightly extended the model by allowing for different $\delta$ committed by nodes. This is reflected by an index $j$. This $\delta_j$ reflects the maximum possible benefit that $j$'s neighbors (such as $i$) can gain, which is further weakened based on the path length $t(ij)$ between $i$ and $j$.

Please cite this article in press as: Skopik, F., The social smart grid: Dealing with constrained energy resources through social coordination. J. Syst. Software (2013), http://dx.doi.org/10.1016/j.jss.2013.04.052

negotiate terms of sharing and set up a service level agreement (SLA). This SLA is a formal document which captures, for instance in the marketplace use case, how much energy is exchanged between[4] nodes and in which time slots. The actually gained utility is periodically assessed using Eq. (2).

### 3.3.3. *Transitive payoff propagation*

Indirect links allow for modeling a dampened propagation of $\delta$ (e.g., energy sharing opportunities). Thus, according to Fig. 4, $\delta_k^2$ can be interpreted as $j$ passes some payoff obtained from $k$ to $i$, so, in other words, $i$ benefits from $k$ indirectly via $j$. For example, when sharing energy, there are network members with highly volatile energy consumption behavior, e.g., a whole house block with several parties being uncoordinated in terms of energy consumption, they can potentially pass on excessive energy. In Fig. 4, this potential makes $j$ more valuable to $i$. However, costs rely on concrete SLAs and are thus accounted between directly connected pairs of nodes only (non-transitive).

In contrast to economic networks where physical goods are dealt, electric energy is a bit different:

- *Energy substitutability*. Energy cannot be shared directly between two network members. In contrast, one member feeds energy into the grid, while another one withdraws the same amount.
- *Hyperdynamics*. The smart grid environment has a super high volatility where load conditions and energy consumption change within seconds. Thus, in reality no stable networks emerge (cf. Jackson and Wolinsky, 1996: where all needs of network members (e.g., in terms of energy sharing) are satisfied).
- *Direct consumption*. Since energy cannot be stored at large-scale, consumers typically buy energy immediately when it is needed. The barely buy energy 'on stock' (exception: the batteries of an electric car can be seen as temporal buffer Ipakchi and Albuyeh, 2009).
- *No overhead costs*. Electric power from numerous sources can be easily combined, thus there is virtually no difference if someone purchases 100 kWh from one network member or 1 kWh each from 100 different sources.

The general utility of a network $G$ is defined as the sum of each single node's utility – see Eq. (3).

$$U(G) = \sum_{i=1}^{n} u_i(g) \tag{3}$$

An optimum[5] net usage is reached, if the sum of each node's utility in $G$ is maximized. Then the network $G^*$ is called efficient (Jackson and Wolinsky, 1996) – see Eq. (4).

$$G^* = \max_{G} \; U(G) \tag{4}$$

## 3.4. *Contract negotiation and SLA setup*

### 3.4.1. *SLA setup*

Some reasons exist for a network member to swap the consumer and producer roles repeatedly. For instance, one might cover a fraction of his own energy demand with solar power, however, then most energy is produced around noon. Therefore, one might need the full amount of generated energy only on weekends, but not from Monday to Friday. Another community member might offer produced wind power on weekends, because then s/he is at another place. Thus both offer there excessively produced energy for sale at the marketplace. For that purpose they need to publish what type of energy they have, the expected availability at some (periodically reoccurring) time frames, and the provided amount.[6] We adopt the WSLA standard (Ludwig et al., 2003), an SLA model which was initially developed for Web services in Serviced-oriented Architectures. We argue that both this model and SOA in general perfectly fit to a socially-enhanced smart grid environment, where consumers can dynamically become service (energy) providers and need to be discovered and 'utilized' on demand. The fundamental parts of WSLA are:

- Links and details to involved parties (*service provider* and *service consumer*)
- a *service definition* in form of an WSDL interface (Alonso et al., 2004), containing single Web services operations for purchasing energy, credit-based compensation, etc.
- *SLA parameters* and *metrics* that are attached to operations.
- *Service level objectives (SLO)* describing the terms of the agreement, such as values of SLA parameters to be reached, e.g., minimum amount of provided energy in predefined time slots.

Two nodes establish an SLA along their connecting social link which is created in the (periodically executed) formation process. A particular instance of an SLA is given in the implementation section of this paper.

### 3.4.2. *Re-negotiation*

Because of the mentioned high volatility of energy consumption behavior, each node will *periodically*[7] re-evaluation its obtained utility (energy, credits, support, etc.) from the network in order to decide about establishing new links and releasing existing ones respectively, or re-negotiate SLAs. Here one could argue that this re-evaluation can be performed per SLA only instead considering the combined net outcome by applying the previously introduced group formation algorithm. However, there are situations where one benefits above average from one partner because of being connected to a considerable amount of 'friends of friends' through transitive relations, which is not reflected by SLAs. This transitive relations however are essential for the discovery of future partners in the social network.

The variable $\overline{t_x}(u_i(g) > 0)$ describes the number of positive evaluations in a time span $\overline{t_x}$, e.g., $\overline{t_{24h}}$. In other words, this is the number of evaluation operations where the obtained utility was greater than zero. In contrast to that $\overline{t_x}(u_i(g) < 0)$ counts how often costs paid exceeded the obtained payoff and thus the overall utility $u_i(g)$ was less than zero.

So, if on average (e.g., over a week or month), costs exceed the payoff (cf. Eq. (5)), a node will attempt to release links with a low payoff-cost ratio ($\delta/c$) and form new ones.

$$\overline{t_x}(u_i(g) < 0) > \overline{t_x}(u_i(g) > 0) \tag{5}$$

A network is called balanced from $i$'s perspective if Eq. (6) applies. This is the minimum requirement for an opportunistic

---

[4] In the simplest case – and as already partially possible today – a community member sells excessive energy to a major energy provider at any time. However, we argue, that this typically does not pay off compared to privately set up contracts, because terms and objectives cannot be negotiated with an industrial partner. Thus sharing among individuals is much more beneficial for both of them.

[5] Notice, this definition does not account for a fair distribution between individuals but only refers to a generalized network perspective.

[6] Notice that this is only an estimation, because most green energy sources (wind, sun) are inherently unreliable. Thus, we introduce an uncertainty factor in the next section.

[7] This means in time intervals of a fixed length.

Please cite this article in press as: Skopik, F., The social smart grid: Dealing with constrained energy resources through social coordination. J. Syst. Software (2013), http://dx.doi.org/10.1016/j.jss.2013.04.052

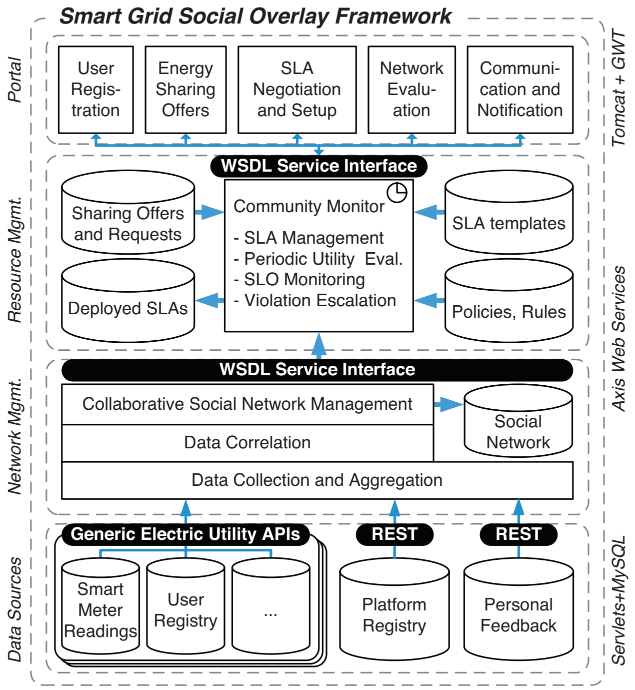


**Fig. 5.** Architectural view showing major components.

community member that does *not* require any actions (i.e., actively looking for new partners).

$$\overline{t_x}(u_i(g) < 0) \approx \overline{t_x}(u_i(g) > 0) \qquad (6)$$

## 4. Architecture and implementation

We basically discuss the mapping of introduced concepts to concrete technologies, and the implementation of a prototype system using state-of-the-art frameworks, components, and protocols.

### 4.1. Service-oriented architecture

The basic architecture as shown in Fig. 5 comprises four layers (1) The *Data Sources Layer* unifies platform specific data (e.g., user registry and profiles, or member recommendations) with data from electric utilities, such as smart meter readings, which are either available through generic Web interfaces or through third party platforms; e.g., PlotWatt.[8] (2) The *Network Management Layer* periodically pulls data from the sources underneath and aggregates them to create a graph model on which the network formation process relies. A crucial part here is data correlation, i.e., the matching of consumption data, user feedback, recommendations, etc. to unique user entities. This correlation needs to be specifically configured based on the actual services used, as well as particular data formats and schemes. A convenient technology to achieve flexible correlation and aggregation is the use of XSLT (Clark, 1999). (3) The *Resource Management Layer* supports the user with setting up SLAs, handles deployed SLAs, monitors respective SLOs, and reports violations by notifying concerned users. The centrally located Community Monitor periodically runs through these steps and applies the social formation algorithm to compare benefits and costs per user with respect to predefined thresholds (policies and rules). Here, the utility values for the single members are calculated. Additionally, $U(G)$ is refreshed in periodic time intervals to rate the

```
<?xml version="1.0"?>
<rdf:RDF xmlns:foaf="http://xmlns.com/foaf/0.1/"
         xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
         xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
         xmlns:foaf="http://xmlns.com/foaf/0.1/"
         xmlns:dc="http://purl.org/dc/elements/1.1/"
         xmlns:wot="http://xmlns.com/wot/0.1/"
<foaf:Person rdf:ID="me">
 <foaf:name>Florian Skopik</foaf:name>
 <foaf:nick>florian</foaf:nick>
 <foaf:mbox_sha1sum>12c683...</foaf:mbox_sha1sum>
 <wot:haskey rdf:nodeID="KeyFS" />
 <foaf:interest rdf:resource="http://..." />
 <foaf:currentProject>
  <foaf:Project>
   <dc:title>WindMillSharing</dc:title>
   <dc:description>green energy, wind, weekdays</dc:description>
   <dc:identifier rdf:resource="http://.../EnergyOffer#42"/>
  </foaf:Project>
 </foaf:currentProject>
 <foaf:knows>
  <foaf:Person>
   <foaf:mbox_sha1sum>73c479...</foaf:mbox_sha1sum>
   <foaf:name>Thomas Bleier</foaf:name>
  </foaf:Person>
 </foaf:knows>
</foaf:Person>
</rdf:RDF>
```

**Listing 1.** Example of a public FOAF profile.

overall efficiency and estimate energy savings through sharing. (4) The top located *Community Portal*, provides the user with various tools/apps for setting up SLAs, evaluating benefits and costs, and communicating with other community members.

The most complex block here is the *Community Monitor*. It basically implements an autonomic computing cycle (IBM, 2005), where based on monitored events and a thorough analysis, targeted actions are executed to remain in a stable mode of operation. This process is configured through various policies and rules, e.g., to notify the user, if a network partner violates SLAs, automatically dissolve links, or inform about new energy sharing offers in time slots of interest. The engine for SLA enforcement is discussed in detail in previous work (Psaier et al., 2011). Here, we apply this work in different context.

### 4.2. Implementation details

We utilize various well-established Web technologies from the area of Web services and the Semantic Web. In this section, we demonstrate major parts of our prototype framework. Notice that we took care to use major Web- and WS standards only and not any home-brewed proprietary protocols.

#### 4.2.1. Social network representation

In recent years, numerous models and protocols have been proposed to represent humans on the Web. One of the most widely used open approaches is Friend-Of-A-Friend (FOAF) (Brickley and Miller, 2010). Various attempts have been undertaken to secure FOAF, e.g., by combining it with SSL (Story et al., 2012) or access rights management (Kruk et al., 2006). In particular, we discussed in previous work (Skopik et al., 2011) an approach to the integration of FOAF and the Web of Trust ontology,[9] as well as public key infrastructures (PKI) (van Tilborg, 2005). Thus, we outline the basics here only.

Listing 1 shows a simplified example of a public FOAF profile, containing basic personal properties (`name`, `nick`, `interest`) and social relations (`knows`). The Web of Trust (WoT) RDF ontology is used to integrate concepts of a public key infrastructure into FOAF profiles, as demonstrated in Listing 2. The property `haskey`

---

[8] https://plotwatt.com/.

[9] http://xmlns.com/wot/0.1/.

```
<!-- restricted part of FOAF profile -->
 <rdfs:seeAlso>
  <foaf:Document rdf:about="http://.../foaf-private.rdf.asc">
   <wot:encryptedTo>
    <wot:PubKey wot:hex_id="34c5a421b" />
   </wot:encryptedTo>
  </foaf:Document>
 </rdfs:seeAlso>

 <!-- digital signature for this file -->
 <rdf:Description rdf:about="">
  <wot:assurance rdf:resource="foaf.rdf.asc" />
 </rdf:Description>

 <!-- public key of the owner/signer of this file -->
 <wot:PubKey rdf:nodeID="KeyFS">
  <wot:hex_id>3756EA0B</wot:hex_id>
  <wot:length>1024</wot:length>
  <wot:fingerprint>03f4...</wot:fingerprint>
  <wot:pubkeyAddress rdf:resource="http://.../key.asc"/>
  <wot:identity>
    <wot:User>
      <foaf:name>Florian Skopik</foaf:name>
      <foaf:mbox_sha1sum>12c683...</foaf:mbox_sha1sum>
    </wot:User>
  </wot:identity>
 </wot:PubKey>
```

**Listing 2.** Signing FOAFs (`wot:assurance`) and linking encrypted content (`rdfs:seeAlso`.

links a public key (`pubkey-Address`), `hex_id`, and `fingerprint` to a `person`. Furthermore, a person's private key is used to sign the own FOAF profile and therefore, to guarantee for integrity and authenticity[10] With respect to the smart grid community use case, members can link energy sharing offers (basically SLA drafts that are further negotiated with and accepted by one of the linked neighbors) to their profiles (see list of `foaf:Projects`).

Access to parts of a FOAF document may be restricted to certain users (whose public keys are used to encrypt those parts). We utilize this concept (Listing 3) in particular for (i) linked SLAs, which are encrypted to be kept confidential between concerned parties; (ii) private information, such as private phone numbers or chat accounts that can only be decrypted and used by close neighbors (connected via `knows`), and (iii) personal ratings to reward and punish behavior, e.g., in terms of reliability of energy sharing among connected neighbors.

```
<rdf:RDF xmlns:foaf="http://xmlns.com/foaf/0.1/"
         xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
         xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#">
 <foaf:Person>
  <!-- mbox_sha1sum links to public FOAF profile -->
  <foaf:mbox_sha1sum>12c683...</foaf:mbox_sha1sum>

  <!-- private contact details -->
  <foaf:mbox rdf:resource="mailto:florian.skopik@..."/>
  <foaf:phone>+43 xxxx xxxx</foaf:phone>

  <!-- private chat account -->
  <foaf:account>
   <foaf:OnlineAccount>
     <rdf:type rdf:resource="http://.../OnlineChatAccount" />
     <foaf:accountServiceHomepage rdf:resource="http://..../" />
     <foaf:accountName>florian_skopik</foaf:accountName>
   </foaf:OnlineAccount>
  </foaf:account>

  <!-- attach personalized ratings to known persons -->
  <foaf:knows>
   <foaf:Person>
    <foaf:mbox_sha1sum>73c479...</foaf:mbox_sha1sum>
    <foaf:tipjar rdf:resource="http://..." rdfs:label="ratings"/>
    <!-- attach SLAs with known persons -->
   </foaf:Person>
  </foaf:knows>

  <!-- link encrypted SLAs -->
  <foaf:Document rdf:about="http://.../SLA-73c479...xml">
   <dc:title>SLA-73c479...</dc:title>
   <wot:assurance>
     <wot:Endorsement rdf:about="http://.../SLA-73c479...asc">
       <dc:title>signature of SLA</dc:title>
       <wot:endorser rdf:nodeID="KeyFS"/>
     </wot:Endorsement>
   </wot:assurance>
  </foaf:Document>

  <!-- encryption information -->
  <wot:EncryptedDocument rdf:about="http://.../SLA-73c479...asc">
   <dc:title>friend47 private profile</dc:title>
   <wot:encryptedTo rdf:nodeID="KeyPartnerX"/>
   <wot:encrypter rdf:nodeID="KeyFS"/>
  </EncryptedDocument>

 </foaf:Person>
</rdf:RDF>
```

**Listing 3.** Private fragment of a FOAF profile.

### 4.2.2. *Energy provisioning and sharing through web services*

Each member who wants to offer a service, e.g., sharing of energy or coordination of energy consumption, deploys a customized Web service. Technically, the creation of these Web services is supported with predefined interface templates and rich user interfaces (XForms), so virtually no special technical skills are required for this step (see Schall et al., 2008; Schall, 2011 for details – out of scope here). The discovery of these services is supported by the social overlay model, where 'friend(s) of a friend' are recommended within the community. We roughly outline here the basic steps to enabling energy sharing through SOA from a provider's perspective as follows:

1 *Service suite creation*. Using predefined PortTypes, a set of services are deployed to (i) enable potential customers to retrieve offers (i.e., discovery of available energy in particular time slots); (ii) allow interested costumers to negotiate conditions (including price and agreed amount of energy)) and deploy SLAs; (iii) support the actual energy sharing actions, in particular, claiming subsequently guaranteed energy in certain time slots and providing compensation in form of credits.
2 *Service interaction*. Once a customer is bound to a provider's services, they are periodically utilized to trade energy against credits, change the requirements of customers (e.g., time slots of sharing), or extend offers.
3 *Service quality assessment*. In parallel, the behavior in terms of reliability (considering defined service level objectives) of involved parties is monitored. In case of violations an escalation strategy may be enforced, e.g., send warnings to the violating party or post negative rating on the community platform.

Listing 4 shows an excerpt of an energy provider's service interface.[11] Here, a customer who is bound to this service, can claim energy through one service operation and provide credits for compensation through a second one. For more details on dynamic service creation by humans, refer to Schall et al. (2008).

### 4.2.3. *SLA specification and enforcement*

The used agreement model is based on work from IBM and the GRAAP Working Group (see related work in Section 6). The overall structure, as given in the excerpts of SLAs in Listing 5 and SLOs in Listing 6 respectively, includes header, agreement items, and terms. The header of an SLA comprises involved parties details and contact

---

[10] Notice, the only guarantee regarding authenticity is that the FOAF signer is owner of the registered mail account that has been used to create the key pair.

[11] Expressed as fully compliant WSDL (web services description language).

Please cite this article in press as: Skopik, F., The social smart grid: Dealing with constrained energy resources through social coordination. J. Syst. Software (2013), http://dx.doi.org/10.1016/j.jss.2013.04.052

```
<!-- excerpt wsdl interface -->
<wsdl:portType name="EnergySharingPortType">
<wsdl:operation name="ClaimEnergy">
 <wsdl:input xmlns="http://www.w3.org/2006/05/addressing/wsdl"
   message="ClaimEnergyMsg" wsaw:Action="urn:ClaimEnergy">
 </wsdl:input>
 <wsdl:output message="AckClaimEnergy" />
</wsdl:operation>
<wsdl:operation name="PutCredit">
 <wsdl:input xmlns="http://www.w3.org/2006/05/addressing/wsdl"
   message="PutCreditMsg" wsaw:Action="urn:PutCredit">
 </wsdl:input>
 <wsdl:output message="AckPutCredit" />
</wsdl:operation>
</wsdl:portType>
<wsdl:binding name="HALSOAPBinding" type="EnergySharingPortType">
 <soap:binding style="document"
 transport="http://xmlsoap.org/soap/http"/>
</wsdl:binding>
```

**Listing 4.** Web service for member interactions.

```
<wsla:SLA
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:wsla="http://www.ibm.com/wsla"
name="SLA4711-WeekendBalance">
 <wsla:Parties>
  <wsla:ServiceProvider name="EnergyProvider">
    <!-- foaf:mbox_sha1sum ... -->
  </wsla:ServiceProvider>
  <wsla:ServiceConsumer name="EnergyConsumer">
    <!-- foaf:mbox_sha1sum ... -->
  </wsla:ServiceConsumer>
 </wsla:Parties>
 <wsla:ServiceDefinition name="ClaimEnergy">
  <wsla:Operation
    xsi:type="wsla:WSDLSOAPOperationDescriptionType"
    name="ClaimEnergy">
   <wsla:SLAParameter name="Energy" type="int" unit="kWh">
    <wsla:Metric>EnergyCount</wsla:Metric>
   </wsla:SLAParameter>
   <!-- further parameters: reliability, uncertainties ... -->
   <!-- config details: name, wsdl-location, binding -->
  </wsla:Operation>
  <wsla:Operation
    xsi:type="wsla:WSDLSOAPOperationDescriptionType"
    name="PutCredit">
   <wsla:SLAParameter name="Credit" type="int" unit="Credits">
    <wsla:Metric>EnergyCost</wsla:Metric>
   </wsla:SLAParameter>
   <!-- further parameters: reliability, uncertainties ... -->
   <!-- config details: name, wsdl-location, binding -->
  </wsla:Operation>
 </wsla:ServiceDefinition>
<!-- wsla Obligations -->
</wsla:SLA>
```

**Listing 5.** SLA excerpt.

```
<wsla:Obligations>
 <wsla:ServiceLevelObjective name="sloEp"
  serviceObject="ClaimEnergy">
  <wsla:Obliged>EnergyProvider</wsla:Obliged>
  <!-- Validity -->
   <wsla:Expression>
    <wsla:Predicate xsi:type="wsla:Equal">
     <wsla:SLAParameter>Energy</wsla:SLAParameter>
     <wsla:Value>75</wsla:Value>
    </wsla:Predicate>
   </wsla:Expression> <!-- evaluation weekly -->
 </wsla:ServiceLevelObjective>
 <wsla:ServiceLevelObjective name="sloEc"
        serviceObject="PutCredit">
  <wsla:Obliged>EnergyConsumer</wsla:Obliged>
  <!-- Validity -->
  <wsla:And>
  <wsla:Expression>
   <wsla:Predicate xsi:type="wsla:GreaterEqual">
    <wsla:SLAParameter>Credit</wsla:SLAParameter>
    <wsla:Value>50</wsla:Value>
   </wsla:Predicate>
  </wsla:Expression>
  <!-- expressions for reliability, uncertainties ... -->
  </wsla:And>
  <wsla:EvaluationEvent>TaskAssignment</wsla:EvaluationEvent>
 </wsla:ServiceLevelObjective>
 <wsla:QualifiedAction>
  <wsla:Party>CommunityBroker</wsla:Party>
  <wsla:Action actionName="violation" xsi:type="Notification">
    <wsla:NotificationType>Violation</wsla:NotificationType>
    <wsla:CausingGuarantee>sloEc</wsla:CausingGuarantee>
    <wsla:SLAParameter>EnergyCost</wsla:SLAParameter>
    <!-- expressions for reliability, uncertainties ... -->
  </wsla:Action>
 </wsla:QualifiedAction>
</wsla:Obligations>
```

**Listing 6.** SLO instance.

**Table 2**
Negotiable agreement attributes.

| Quality attributes | Description |
|---|---|
| Energy amount | Amount of delivered and consumed energy in kWh. |
| Credits | Compensation for delivering energy. |
| Availability | Predicted availability of energy provisioning depending on energy source and own fluctuating consumption in a predefined time span (e.g., a week or a month). |
| Production uncertainty | Fraction of the amount of energy that might not be delivered due to inherent production uncertainties or concurrent[a] SLAs. |
| Consumption uncertainty | Fraction of the amount of energy that might not be consumed due to inherent consumption uncertainties[b]. |

[a] One might decide to agree to more customers than s/he can actually serve – similar to flight companies which overbook their airplanes, because they know, some guests never show up.

[b] Who knows how much energy for an electric heater will be required in a few weeks?

information. In the contractual items the agreement subjects are listed. These include the service content (i.e., for Web-services the WSDL location, endpoint, and operation) along with metrics, their representation and method of measurement.

Finally, the terms provide the objectives, SLOs respectively (see Listing 6), and their validity period. Threshold values expresses the desired relation between objectives and metrics defined in the items.[12] In the given example, SLOs describe agreed amount of delivered energy and compensation through the customer, as well as escalation strategies in case of violations. An SLO consists of an `Obliged Party`, a validity period, and expressions that can be combined with logic expressions (e.g., `And`). The content of an expression connects the pool of SLAParameters of the items to a predicate (e.g, `GreaterEqual`) and a threshold value (`Value`). The final tag `QualifiedAction` defines the consequence of an SLO violation. In the example case, if a threshold of SLO `sloEc` is violated an action of type Notification is called.

We further provide a number of quality metrics that can be automatically monitored, determined, and enforced in our system, and thus, are aligned to the described protocol structures. This list can easily be extended to incorporate further rewarding and punishment mechanisms as discussed above, but is omitted here due to brevity. Instead, we focus on metrics for the energy sharing use case only (Table 2).

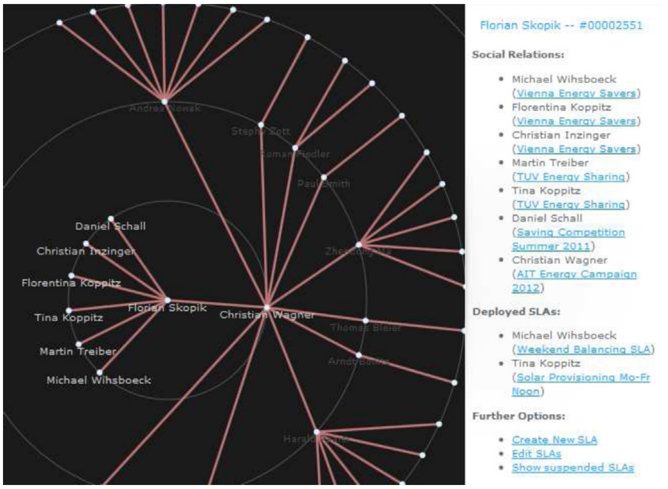
### 4.3. End-user perspective

On top of the social network layer and Web services based environment a collaboration portal supports the various features required in order to:

---

[12] Notice that mentioned WS operations (`ClaimEnergy`, `PutCredit`, etc.) match the interface given in the previous section.

Please cite this article in press as: Skopik, F., The social smart grid: Dealing with constrained energy resources through social coordination. J. Syst. Software (2013), http://dx.doi.org/10.1016/j.jss.2013.04.052



Florian Skopik -- #00002551

**Social Relations:**

- Michael Wihsboeck (Vienna Energy Savers)
- Florentina Koppitz (Vienna Energy Savers)
- Christian Inzinger (Vienna Energy Savers)
- Martin Treiber (TUV Energy Sharing)
- Tina Koppitz (TUV Energy Sharing)
- Daniel Schall (Saving Competition Summer 2011)
- Christian Wagner (AIT Energy Campaign 2012)

**Deployed SLAs:**

- Michael Wihsboeck (Weekend Balancing SLA)
- Tina Koppitz (Solar Provisioning Mo-Fr Noon)

**Further Options:**

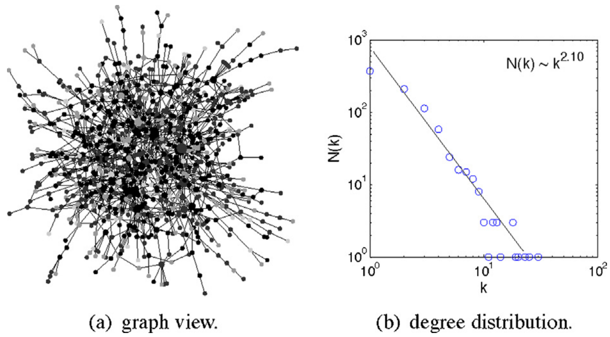
- Create New SLA
- Edit SLAs
- Show suspended SLAs

**Fig. 6.** Community member perspective.

- *Manage profiles*, such as register a new user, add further profile data (interests, typical consumption profiles, etc.). This is invaluable information for setting up energy sharing contracts with community members, however, one needs to act with caution not to compromise his privacy.
- *Discover new energy sharing partners*. We picture a platform that allows to offer energy sharing opportunities through an electronic marketplace. Currently, however, we do not employ formal mechanisms, but let people discuss their requirements in a threaded discussion forum. Although, here energy providers can post SLA templates that another party can agree with. Furthermore, in order to facilitate the discovery of new network partners, we employ recommendation mechanisms for friends-of-friends (transitive relations) who frequently offer excessive energy.
- *Set up SLAs*, once two members have agreed on the objectives. Here, we utilize XForms[13] to render an SLA and its objectives. So, involved members do not need to edit XML files but can conveniently fill in online forms.
- *Monitor and visualize the network*, including monitoring SLO violations to trigger escalations; and visualizing social connections[14] and deployed SLAs. Fig. 6 shows this feature from the perspective of the centered user (here: 'Florian Skopik'). Other users are one, two or three hops away.[15]
- *Send notification and enable escalations*. In case of SLO violations several options exist depending on the severity of the infringement; beginning with simple warnings, repeated violations may cause blacklisting a member or blocking her/him from the community portal at all.

## 5. Evaluation and discussion

We conducted several experiments to learn about the feasibility of our approach, chosen technologies, and the application of SOA concepts in the social smart grid environment. In the center of our experiments is an agent-based simulation which stresses the prototype implementation of the *Smart Grid Social Overlay Framework* in a realistic manner. In context of this simulation, we measure various performance aspects and rate the overall feasibility.



(a) graph view. (b) degree distribution.

**Fig. 7.** Created community network.

### 5.1. Experiment setup

#### 5.1.1. Technical infrastructure setup

Our evaluations[16] were gathered using the logging features of the Genesis2 framework (Juszczyk and Dustdar, 2010). Genesis2 has a management interface and a controllable runtime to deploy, simulate, and evaluate SOA designs and implementations. A collection of extensible elements for these environments are available such as models of services, clients, registries, and other SOA components. Each element can be set up individually with its own behavior, and steered during execution of a test case. For the experiments in this work, we deployed Genesis2 Backends to the *Amazon Elastic Compute Cloud*.[17] We launched, depending on the amount of involved service instances, two or three Community AMIs of the type High-Memory Extra Large Instance (17.1 GB of memory) running a Linux OS. In the following, we provided each instance with the same Genesis2 Backend snapshot via mountable volumes from the Elastic Block Store. Finally, we deployed the following environment setup from a local Genesis2 Frontend. It included SOA-based energy communities established by Genesis2 Web services equipped with simulated behavior and predefined relations to provide sharing channels and instantiate communities. Services act like community members when they establish social connections, set up SLAs, and invoke energy sharing WS operations (see `ClaimEnergy` and `PutCredit` in the previous section). Refer to the scenario description later in this section for more details on the behavior of these services (actually a kind of agent in an agent-based simulation using SOA technologies).

The Smart Grid Social Overlay Framework itself is deployed on an Apache Tomcat 6.0 which runs on a separate Windows 7 Enterprise (x64) machine with an Intel i7-2620 CPU@2.7 GHz and 8 GB RAM.

#### 5.1.2. Synthetic community network structure

Since we have not yet applied our approach in real large-scale environments, we do not have sufficient real testing data. Therefore, we generate artificial scale-free network structures that we would expect to emerge under realistic conditions in typical collaboration networks (Reka and Barabási, 2002) to test and discuss our framework. We utilize the preferential attachment model of Barabasi and Albert (Reka and Barabási, 2002) to create graphs with power-law distributed degrees depicted in Fig. 7. These network structures are the basis to conduct an agent-based simulation following realistic assumptions; for instance, estimating the amount of managed SLAs, number of social connections between

---

[13] http://www.w3.org/TR/xforms/.
[14] http://thejit.org/.
[15] Notice the circle layout.
[16] This is basically the same environment but with different configuration as first described in Schall et al. (2011).
[17] Amazon EC2: http://aws.amazon.com/ec2/.

Please cite this article in press as: Skopik, F., The social smart grid: Dealing with constrained energy resources through social coordination. J. Syst. Software (2013), http://dx.doi.org/10.1016/j.jss.2013.04.052

members, and realistic application of the group formation algorithms for neighbor assessment.

### 5.2. Agent-based simulation

We mainly run an agent-based simulation to stress the technical infrastructure, learn about performance requirements of most utilized features, and scalability of the overall platform. In this experiment, agents can feed back excessively produced energy (e.g., from private solar panels on sunny days) to the grid and energy sharing partners agree to withdraw the same amount from the grid at the same time; thus, effectively leading to a smoothened energy consumption without peak loads in the utility. Notice, although this simulation offers interesting insights and provides a good estimation for the achievable energy savings through community-based coordination and sharing, the real processes are far more complex and there are for sure influences and factors that have not been considered or oversimplified in this simulation. Nevertheless, we put best effort to set up a realistic future environment as described below.

#### 5.2.1. Agent population setup

Recently, the University of Stratclyde performed a thorough evaluation of existing user groups and their energy consumption behavior, and furthermore, developed an energy demand profile generator for further experiments (University of Stratclyde, 2007). We use this tool to model the agent behavior in our simulation. Private households can be classified according to their occupancy in a number of groups from which we picked the four most common ones. These groups cover around 75% of the population: (i) single adult, (ii) two adults, (iii) two adults with child, and (iv) single pensioner. Further profiles investigated in University of Stratclyde (2007) are two pensioners, two adults with pensioner, and three adults. The properties of groups relevant for the following simulation are summarized in Table 3. Corresponding energy consumption profiles are depicted in Fig. 8. These profiles show the energy consumption in kWh over the period of 24 h. Notice, the particular data to create these profiles was taken from University of Stratclyde (2007). Thus, energy demands of all households of the same group are aggregated here leading to a population of 767 agents (when using the four largest groups only).

#### 5.2.2. Agent behavior setup

We set up 767 agents[18] with different behavior profiles so that we get realistic community-wide conditions as observed in the mentioned real world study. Then we generate a scale-free social network (cf. Fig. 7) and assign agents randomly to the nodes of this graph. This is the starting point for the simulation.

We apply a number of assumptions to ease the simulation. First, we do not distinguish between summer and winter, however, we exclude light loads which minimizes the effect of this assumption. Second, we assume each household has an average set of electric appliances and there is virtually no difference in their consumption behavior averaged over one year. Third, there is of course a volatility of available self-generated energy (due to changing ‘weather conditions’). Because of different individual situations of households (location, energy sources, etc.) this volatility is individually modified in the simulation for each household. However, in reality, there is a correlation of amounts of self-generated energy (especially wind and solar) between geographically co-located network members because of similar weather conditions. Fourth, each household can cover between 5% and 40% of its demand with self-generated green energy from renewable energy sources (World Wind Energy Association, 2011). This is an optimistic assumption for today, however, we are investigating a future energy sharing platform, and expect green self-generated energy to rise considerably in the next years. Especially on uncommonly sunny or windy days the actually required level of self-generated energy is exceeded, which is a great opportunity for sharing (even buffered energy from an electric car). Notice, that we do not distinguish between energy source types nor simulate special weather conditions; but randomly set the amount of produced energy between 5% and 40% of the actually required amount and not synchronized to the actual consumption times; thus, there are slots of overproduction where energy can be shared. The volatility of the amount of generated energy is moderate with a change rate[19] of max. 10% per simulation round.

Further notice that the aim of energy sharing is that agents use *all* available energy from renewable energy sources offered through the social network. So, they just cover that amount of required energy which exceeds the available renewable energy with energy from industrial power plants (traditional energy producers). Hence, agents do not decide whether renewable energy from neighbors should be used or energy from industrial power plants (which might be cheaper in reality). Renewable energy is always favored; however, agents decide themselves from which neighbor they like to “buy”. To further ease simulations, we simply assume that the missing amounts of energy in the whole network are always provided by regular power plants without any limitations.

#### 5.2.3. SLA setup

We set up a round-based simulation, where each round represents 1 h real time. An SLA basically defines slots for energy sharing on either a daily basis (for instance `(round%24)+9` to `(round%24)+17` means from 9 am to 5 pm) or weekly basis (for example `(round%168)+(4*24)+20` to `(round%168)+(6*24)+18` means from Friday 8 pm to Sunday 6 pm).

Basically[20] we assume agents share produced energy in their respective unoccupancy times (see Table 3 and Fig. 8); additionally, pensioners share (wind or geothermal) energy starting with 10 pm (cf. energy demand profiles in Fig. 8). In particular, we apply the following feasible assumptions (University of Stratclyde, 2007) based on a weekly cycle: (i) single adults are not at home during weekdays from 9 am to 6 pm, and generally on weekends; (ii) two adults have a similar pattern on weekdays, but stay at home on weekends; (iii) pensioners always need energy during days, but can spend in the later evening starting with 10 pm; and (iv) two adults with children can only spend energy on weekdays from 9 am to 1 pm, however are reliable energy consumers.

Regarding the amount of shared energy, we set up that agents claim energy from social network neighbors whenever they cannot fully (100%) cover their demands. Agents provide only excessive energy which means that they first cover their own demand and sell energy which exceeds this limit. For that purpose we apply a price model where basically 1 kWh excessive energy is sold for 1 credit if own demand is just covered; and price decreases linearly if amount of excessive energy in the whole community is higher. We apply general uncertainty factors of 15%. In other words, energy providers can provide up to 15% less energy than agreed without any punishment (i.e., affecting consumers to renegotiate or release the SLA), and energy consumers can demand up to 15% less energy without any punishment from the provider. These are averaged

---

[18] Technically, these are Web services which run in the Genesis2 framework and access the community service depicted in Fig. 5.

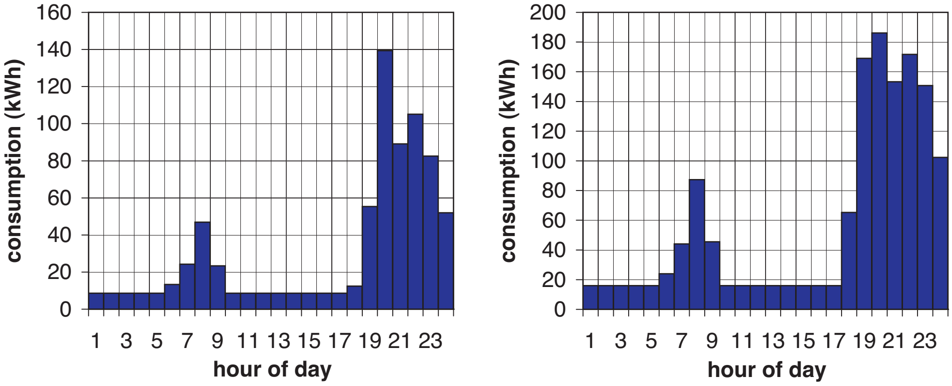
[19] Reasonable value determined through experiments for a mix of wind and solar energy.

[20] Notice, we set up fixed time slots and fixed SLAs for sharing depending on the agent group. Although this roughly matches reality, the sharing behavior will be more flexible for real users.

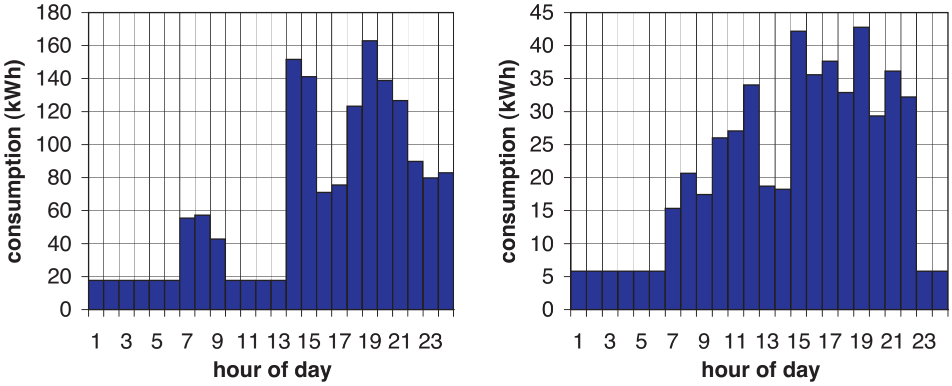
Please cite this article in press as: Skopik, F., The social smart grid: Dealing with constrained energy resources through social coordination. J. Syst. Software (2013), http://dx.doi.org/10.1016/j.jss.2013.04.052

**Table 3**
Properties of agent groups (from University of Stratclyde, 2007).

| House-hold | Unoccupancy times | Other assumptions |
|---|---|---|
| Single adult | 09:00 to 18:00 on weekdays | Occupied by a full time working adult; the average daily consumption of every appliance is distributed through out the day into two main periods, 6:00 till 9:00 and 18:00 till 01:00. |
| Two adult | 09:00 to 18:00 on weekdays | Usage pattern is similar to one adult household. |
| Two adult with children | 09:00 to 13:00 on weekdays | One member has a full time job; the second adult holds a part time job in the morning in order to take care of the children after school. |
| Single pensioner | Occupied all the time | Most loads are distributed through out the day in a random way and only what is related to cooking has a specified period (for lunch and dinner). |



(a) 234 single adult per day. (b) 218 two adults per day.



(c) 156 adults with children per day. (d) 159 single pensioner per day.

**Fig. 8.** Energy consumption profiles for different households.

values for predicting the availability of a balanced mix of sun, wind, biomass, and geothermal power.

### *5.2.4. SLO monitoring and formation setup*

Payoffs $\delta$ and cost $c$ of a node are dynamically set based on their behavior. For the given use case we configure the mappings as follows. From $i$'s perspective $\delta_j$ depends on $j$'s relative importance (in the network $g$) for the energy supply of $i$. Weights $w$ ($\sum_i w_i = 1$) determine the contribution of the partners to the coverage of $i$'s energy demand. Thus, the payoff of $j$ to $i$ is modeled as given in Eq. (7).

$$\delta_j = \frac{w_j}{\sum_{k:ik \in g} w_k} \tag{7}$$

The deviation $\Delta_{ij}$, averaged over SLOs in SLAs set up by $i$ and agreed by the obliged party $j$ are compared to real measured values. We argue that higher $\Delta$ cause higher $c$, because members need to concern and invest effort to actively monitor a partner's energy sharing behavior and – in case of high $\Delta$ – prepare to look for new energy sharing partners. While in reality further factors might influence costs, we argue that basically a simple stepwise cost function (Eq. (8)), which essentially depends on the deviation of measured metrics from agreed values only, is sufficient in many cases to reflect emerging costs with increasing unreliability of partners.

$$c_j = \begin{cases} 0 & \text{if } \Delta_{ij} = 0\% \\ 0.25 & \text{if } 0\% < \Delta_{ij} \leq 15\% \\ 0.5 & \text{if } 15\% < \Delta_{ij} \leq 25\% \\ 0.75 & \text{if } 25\% < \Delta_{ij} \leq 40\% \\ 1 & \text{if } \Delta_{ij} > 40\% \end{cases} \tag{8}$$

Please cite this article in press as: Skopik, F., The social smart grid: Dealing with constrained energy resources through social coordination. J. Syst. Software (2013), http://dx.doi.org/10.1016/j.jss.2013.04.052

Agents periodically re-evaluate obtained utility over a time frame of six weeks (168*6 rounds).[21] However, consider that in real life people have different ambitions regarding the involvement in (energy sharing) communities. Thus, we let only 5% of agents evaluate their utility on a daily basis (every 24th round, but considering the most recent 168*6 rounds), 10% every 48th round, 20% every 72nd round, 30% every 168th round (weekly), and 35% every 336th round (two weeks). Similarly, we enable 35% to maintain only 1 SLA at the same time; 30% maintain 2, 20% set up 3 concurrent SLAs, 10% have 5, and 5% have 10 concurrent SLAs. This distribution roughly matches the power-law node degree distribution of realistic scale-free social networks. Due to the lack of real data (and because we intended to avoid stereotypic assignments), we randomly assigned agent behavior with a uniform distribution over the four different agent groups.

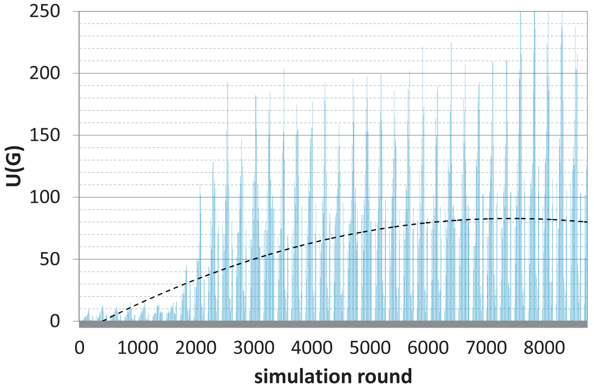
Since we need to reduce complexity of SLA setup and negotiation for the automatic agent-based simulation, we use only three sharing periods, which are fixed based on the agent type. In particular, in the time span from 9 am to 1 pm we let single adults, two adults, and adults with children provide energy; from 1 pm to 6 pm only single adults and two adults provide, while from 10 pm to 12 pm pensioners provide energy (cf. Fig. 8).

### 5.2.5. Basic scenario and simulation run

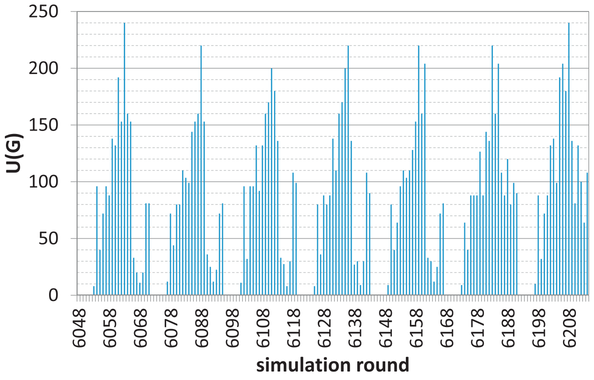
The round-based agent simulation basically covers the use case of a credit-based energy marketplace on-top of a community network. Excessive energy which is currently generated but not used is delivered to network members, who compensate this effort with virtual credits. These credits can be used later on to buy energy from other members and is thus a way of 'virtual energy storage'.

Initially, agents deploy only moderate[22] SLAs, so that 25% of an agents energy demand are covered by the sum of its neighbors. These SLAs are subsequently modified at predefined utility evaluation points. The controlled SLA adaptation is performed as follows: From a consumer's point of view, increase claimed energy by 1%[23] if the linked provider still had free capacities in the last evaluation period, and the consumer demand is not satisfied. From a producer's point of view, decrease claimed energy by 1% if provider was overutilized in the last evaluation period. Furthermore, in the simulation we ignore credit payment adaptation and apply the linear price model as mentioned in the setup phase. So compensation for energy transfer is dynamically set and upper limits in SLAs are not increased/decreased at run-time.

Consuming agents dynamically set their demands (cf. consumption profiles before), and producing agents dynamically set price thresholds based on energy availability and own needs. The price of energy is regulated based on availability and demand, e.g., widely available excessive energy at noon is cheaper than widely required energy in the morning or evening. Then, they periodically re-evaluate their utility (payoff vs. costs of network participation) to optimize their net outcome. Consuming agents can cut links if utility evaluates negative in a time period of 6 weeks (i.e., $\overline{t_{168*6}}(u_i(g) < 0) > \overline{t_{168*6}}(u_i(g) > 0)$; cf. Eq (5)). In case of higher energy demands, they try to add new links randomly[24] based on a transitive network member discovery process (basically an agent tries to add friends-of-friends if their utility is positive and the target agents accepts if s/he is able to offer excessive amounts of energy).



(a) utility over a year.



(b) utility in a settled week.

**Fig. 9.** Emergence and evolution of utility over a year; and within a (calendar) week when simulation is already settled.

### 5.2.6. Simulation discussion

We run the simulation for 365*24 rounds, which roughly reflects one year where one round corresponds to 1 h. Fig. 9 shows the overall evolution of the community-wide utility $U(G)$ (cf. Eq. (3)), first for a year and exemplarily for a month. In the beginning of the simulation, we experience a phase of utility emergence, where the obtained utility level slowly rises to its settled value – see Fig. 9(a) until approx. round 2500. The reason is that members are not optimally[25] connected and need to cute existing and establish new links respectively. Although utility of each single agent is calculated after *every* round, decisions for renegotiating SLAs, cutting or establishing relations, etc., are felt after a 6 weeks time window (every (168*6)th round) of observation (cf. Eq. (5)).

The observed utility is highly oscillating. Typically, most utility is achieved (globally) in the afternoon, because here two whole user groups are able to spend energy (single adults, and two adults), while others (adults with children and pensioners) reliably have a need for energy. At the end of each day there is a spike again, where single adults, two adults and adults with children benefit from pensioners who share energy. In the night no one has a need for energy (at least in our simplified simulation), while in the morning and in the evenings no one can afford to spend energy. In these time slots, utility is lowest, because no one benefits from each other.

Regarding the structure of the social network, we observed, that after the simulation, different agent groups are well mixed and interconnected; for instance single pensioners and single adults complement one another in terms of energy sharing, while agents

---

[21] This measure avoids oscillating link establishment on a daily basis.

[22] This would lead to a stable network where all demands of members are covered (Jackson and Watts, 1999) if there were no uncertainties in energy generation.

[23] Value determined through numerous experiment runs. This value reached the highest utility in $G^*$.

[24] Because of the long evaluation periods we do not maintain a history of neighbors. This means, that also former neighbors whose links have been cut recently, can become new neighbors again.

[25] Keep in mind, we connected them randomly following a realistic power-law distributed social network.

Please cite this article in press as: Skopik, F., The social smart grid: Dealing with constrained energy resources through social coordination. J. Syst. Software (2013), http://dx.doi.org/10.1016/j.jss.2013.04.052

from the same group and having similar demands are hardly connected.

According to our simulation set up, energy savings are higher when utility is higher. Basically utility just expresses to which degree members follow agreed SLAs, however, if a consumer's energy demand is not satisfied, s/he will try to increase the volume of energy to a possible top limit (typically until SLAs cannot be satisfied any longer; cf. SLA setup phase). Therefore, both utility and energy savings oscillate and correlate to a certain degree (highly variable due to many influencing factors in the simulation). In Fig. 10, we highlight the global energy savings in the whole network. For that purpose, after every 168 rounds (1 week) the amount of shared energy (kWh) is counted and compared to the overall consumption (according to consumption profiles discussed before). For the described setup, we experience energy savings of around 7% (moderate sharing), which simply means that *on average* households produce at some time slots 7% more energy than needed. But at the same time the demands of other community members can be satisfied with that energy and thus, they do not need purchase from industrial power plants. Further results (intensive sharing) show the maximum possible energy saving rate (around 9%) if we switch from opportunistic (Skopik et al., 2011) credit-based sharing to altruistic sharing, i.e., whenever someone generates excessive energy another member can use it free of charge. This case, however, is a variation of an open marketplace and it is therefore unlikely that a fair sharing community will emerge, whose fundament is mutual give and take.

We conclude that for the given configuration as described before, we were able to achieve a stable network over a long runtime, i.e., most network members gained significant utility from participation and had at least two neighbors with whom they shared energy and coordinated consumption behavior. This result makes us confident that this approach might also work in a real world setting. However, further studies within a test group are required to fine-tune the formation, SLA negotiation, utility evaluation and sharing processes.

#### 5.2.7. Framework scalability

With our tests, we found out that the most used features of the implemented framework, and thus the most critical operations in terms of performance and scalability, are (in this order) (i) ordinary Web services calls (claim energy, put credits, send messages and notifications), (ii) social graph accesses used for utility evaluation and recommendations, including read access and manipulations of FOAF, (iii) SLA management, including SLO monitoring, and (iv) utility evaluations. When designing and deploying a SOA-based social energy sharing platform, one needs to focus on these aspects to ensure smooth operation. Thus, we especially investigate the first two aspects in greater detail in the next section.

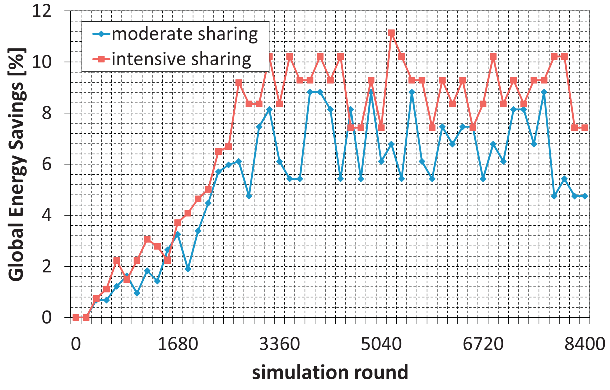


**Fig. 10.** Aggregated energy savings (in %) when sharing in whole *G*.

### 5.3. Discovery scalability and group formation

The social network is the 'backbone' of our whole approach and its properties need to be studied in detail. Especially the efficient discovery of energy sharing partners is crucial for the success of our system. To this end, a graph model is used to discover trustworthy community members, i.e., members that direct friends are linked to. In other words, a transitive relationship $t(ij) = 2$ from node $i$ to node $j$ with a single intermediate node $k$ can be emphasized as a recommendation of $k$ through $j$ towards $i$. The reasons for discovering new partners in the social network are manifold, for instance, long-term partners disappear or change their energy sharing behavior; or novel requirements emerge that can not be covered by current partners. From a technical point of view we measured how many indirect neighbors are reachable *on average*[26] in a network. The results heavily depend on two factors: (i) the interconnectedness of the graph, expressed here as the average node degree (*avgdeg*), i.e., the average number of neighbors one member has; and (ii) the maximum number of edges $t(ij)$ a transitive relation (path) from node $i$ to node $j$ can consist of.

Fig. 11(a) shows the average number of reachable nodes in a network of $N = 800$ for different graph densities ($avgdeg = (2, 5)$) when applying $t(ij) = (1, 6)$. Under realistic conditions networks have an average node degree between 2 and 3; however, this value can be increased by introducing synthetic relations based on common properties (such as matching interests or co-location). Fig. 11 demonstrates that only small $t(ij)$ are feasible to obtain a distinguished set of partners, while the majority of users is not recommended. Another interesting aspect is the number of social graph accesses when traversing the network and determining indirect neighbors for each single participant. For higher $t(ij)$ the number of required graph operations, and thus the computational effort, significantly[27] increases (see Fig. 11(b)). These operations are carried out through Web services, each call requiring parsing FOAF files and thus causing costs in terms of execution time.

Fig. 12 shows the scalability of member discovery for smaller and larger graphs up to 10000 nodes. Here we apply a fixed $t(ij) = 2$, equal to the natural notion of recommendation with one intermediate hop, and investigate the number of discovered network partners and number of required graph operations for different network sizes. We found out that while the number of discovered nodes sharply increases when enlarging the network, the number of graph operations does not, which means a good scalability of the discovery approach (applying $t(ij) = 2$).

### 5.4. Evaluation of security mechanisms

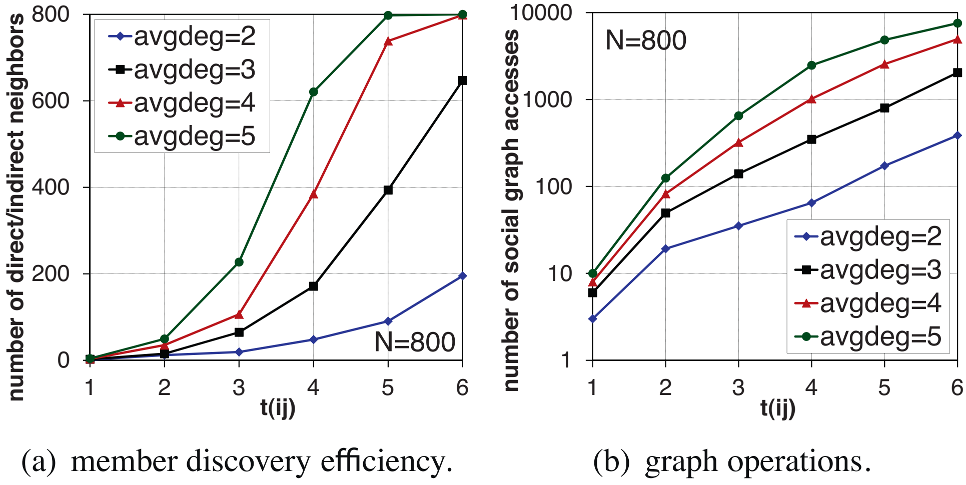
For privacy and information security reasons, personally identifiable data sent over the communication channel needs to be protected from unauthorized access or modifications. Whereas the transport layer is easily protected by employing HTTPS/SSL, end-to-end security in the proposed SOA environment is achieved by implementing the WS-Security (OASIS, 2006) standard. By this means the content of the exchanged SOAP messages is directly encrypted and digitally signed. Due to the additional SSL handshake, the impact of HTTPS/SSL communication is greater for shorter messages (Sosnoski, 2009), but not very significant in general. In internal testing we have found out, that the response time is on average 10 times higher when WS-Security is used. We have used GlassFish 3.1.1 as application server, a very simple Web service, and soapUI 4.0.1 for load testing. Server and client were located in the same organization-wide local area network.

---

[26] Notice, that we are talking about power-law distributed node degrees.

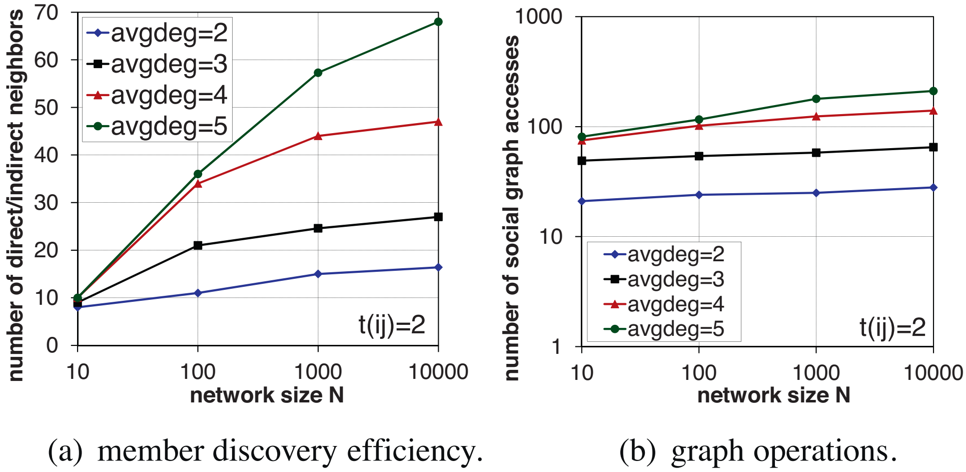
[27] Notice the logarithmic scale.

Please cite this article in press as: Skopik, F., The social smart grid: Dealing with constrained energy resources through social coordination. J. Syst. Software (2013), http://dx.doi.org/10.1016/j.jss.2013.04.052



(a) member discovery efficiency. (b) graph operations.

**Fig. 11.** Fixed size graph and varying transitive path lengths.



(a) member discovery efficiency. (b) graph operations.

**Fig. 12.** Fixed transitive path length and varying graph size.

We omitted concrete evaluation results because these are usually highly dependent on the concrete evaluation setup, system load and usage behavior. Furthermore, the obtained results had a high variety which is the reason why we stick to general conclusions: Our findings are compliant to (Novakouski et al., 2010) and show that adding message-level security to SOAP conversations has significant impact on the system performance in terms of message throughput. However, this is not a drawback, because sophisticated technologies from the Web systems domain exist to enable load balancing – and thus to combine both security and satisfactory performance.

## 6. Related work

Establishing Smart Grids (Ipakchi and Albuyeh, 2009) is a major aim of the European Union (ERGEG, 2010), as well as covering a considerable amount of the energy demand with green energy (World Wind Energy Association, 2011). While the basic technologies are currently under development, applications far beyond automatic meter reading are lively discussed; for instance social networking for smart grids is motivated by Chima (2011). For that purpose coalitional and cooperative game theory (Osborne and Rubinstein, 1999; Jackson and Wolinsky, 1996) is a promising approach to form strong communities. The availability of rich and plentiful data on human interactions in social networks has closed an important loop (Kleinberg, 2008), allowing one to model social phenomena and to use these models in the design of new computing applications such as Web-based collaboration environments. Semantic Web service communities as introduced by Medjahed and Bouguettaya (2005) foster the creation of structured social compositions with predefined community interfaces and functionality. However, ontology structures are not well suited for human communities, because these structures emerge bottom up and are difficult to capture with regard to functionality and interactions between members. Also, value networks (Allee, 2000) are of interest when business aspects are investigated in collaboration settings, i.e., the value that can be generated by such networks based on offered capabilities and goods. In our work, we selected the FOAF protocol as the technical basis to model social networks (Wasserman and Faust, 1994). However, security (van Tilborg, 2005) and privacy (Yokoo et al., 2005) concerns must be properly addressed when using these technologies. Thus, various extensions exist, such as FOAF-SSL (Story et al., 2012) and D-FOAF (Kruk et al., 2006) to ensure secure social networking.

Service-oriented computing (SOC) promises a world of cooperating services loosely connected, creating dynamic business processes and agile applications that span organizations and platforms, but also – as in a Smart Grid context – people across boundaries (Georgakopoulos and Papazoglou, 2008). Service-oriented architectures (SOA) have emerged as the defacto standard

Please cite this article in press as: Skopik, F., The social smart grid: Dealing with constrained energy resources through social coordination. J. Syst. Software (2013), http://dx.doi.org/10.1016/j.jss.2013.04.052

to design and implement professional large-scale collaboration systems on the Web. They allow for loose coupling between single components and enable sophisticated discovery mechanisms based on functional (e.g., supported features) and non-functional (e.g., QoS) properties. Web service technology (Alonso et al., 2003) enables cross-boundary interactions in collaborative networks (Camarinha-Matos and Afsarmanesh, 2006). Major software vendors have been working on standards addressing the lack of human interaction support in service-oriented systems. WS-HumanTask (Amend et al., 2007a) and Bpel4People (Agrawal et al., 2007b) were released to address the emergent need for human interactions in business processes. These standards specify languages to model human interactions, the lifecycle of human tasks, and generic role models. Role-based access models (Amend et al., 2007a) are used to model responsibilities and potential task assignees in processes. While Bpel4People-based applications focus on top-down modeling of business processes, *service-oriented crowds* target flexible interactions and compositions of human-provided services (Schall et al., 2008). This approach is aligned with the vision of the Web 2.0, where people can actively contribute services. In such networks, humans may participate and provide services in a uniform way by using the HPS framework (Schall et al., 2008). In our work, we combine SOA concepts and social principles. We consider *open service-oriented communities* wherein services (such as energy provisioning, but also energy awareness campaigns, or energy consumption coordination) can be added at any point in time. Following the open world assumption, humans actively shape the availability of services. The concept of human-provided services (HPS) (Schall et al., 2008) supports flexible service-oriented collaborations across multiple locations and domains.

There has been substantial research on translations of service level agreements (SLAs) to a Web-service applicable standard (Andrieux et al., 2007; Ludwig et al., 2003). These approaches present similar XML-based SLA models, however, differ in the details. IBM's WSLA focuses on defining agreement objectives, their constraints and combination. For this purpose parameters can be linked to SLOs together with thresholds. In our work we reuse the parameter schema to define our quality attributes. In the last years, SOA and Web services (Alonso et al., 2004) have been in the focus of both academia and industry research. Convenient technologies allow for easy interoperability and automation. Especially when combining SOA with SLAs (Psaier et al., 2011) powerful applications can be realized with minimal or completely without human intervention. Here, human-provided services (Schall, 2011) are a further building block of service networks, such as energy sharing communities.

## 7. Conclusion

This paper described a SOA-based framework to enable efficient and secure social networking among smart grid stakeholders using open standards, such as PKI and the Web of Trust ontology. The implementation is fully compliant to Web services standards, SLA models and Web communication protocols, enabling a seamless integration of social network members and their provided services. The demonstration of these technologies in context of smart grid communities, an innovative new application area, is an important contribution of this paper. We investigated the scalability of our prototype implementation for communities of varying sizes. Sophisticated security mechanisms, such as the application of the Web-of-Trust concept, are important to enable customer trust in our platform. Additionally, we run an agent-based simulation following realistic assumptions and studied the feasibility of proposed community formation, SLA negotiation and monitoring mechanisms. In particular we found out that, given that the agent profiles and behavior match real community structures, our approach has great potential to solve two main research questions regarding smart grid operations:

1. How to coordinate energy consumption in highly distributed large-scale grids in order to allow for a more precise energy consumption estimation through utilities?
2. How to make common consumers integral players of the smart grid on order to raise awareness for the wealth of energy and ultimately to achieve considerable energy savings?

Future work includes a demonstration set up for a small real community. With the continuously proceeding roll out of smart meters, people will be able to monitor their energy demand in fine grained time intervals. We plan to study different ways on how to map these data into online platforms in order to finally apply the described framework in a real environment. However, major privacy concerns arise here, which need to be addressed, e.g., through pseudonymization.

## Acknowledgments

This work is considerably based on previous joint research with Daniel Schall and Harald Psaier. The paper is an extended version of Skopik and Wagner (2012). The author likes to thank all people who contributed to the original work. Furthermore we want to express our gratitude to Lukasz Juszczyk for developing and providing the Genesis2 framework.

## References

Agrawal, A., et al., 2007b]. WS-BPEL Extension for People (bpel4people), Version 1.0.
Allee, V., 2000]. Reconfiguring the value network. Journal of Business Strategy 21 (August (4)).
Alonso, G., Casati, F., Kuno, H., Machiraju, V., October 2003. Web Services – Concepts, Architectures and Applications. Springer.
Alonso, G., Casati, F., Kuno, H., Machiraju, V., 2004]. Web Services: Concepts, Architectures and Applications. Springer, Berlin.
Amend, M., et al., 2007a]. Web Services Human Task (WS-HumanTask), Version 1.0.
Andrieux, A., et al., 2007]. Web services agreement specification (WS-agreement). Open Grid Forum.
Brickley, D., Miller, L., FOAF vocabulary specification 0.98, 2010. http://xmlns.com/foaf/spec
Camarinha-Matos, L.M., Afsarmanesh, H., 2006]. Collaborative networks. In: PRO-LAMAT., pp. 26–40.
Chima, C., How social media will make the smart energy grid more efficient, 2011. http://mashable.com/2011/02/08/smart-grid-social-media/
Clark, J., Xsl transformations (xslt) version 1.0, W3C Recommendation, Tech. Rep., 11, 1999. [Online]. Available: http://www.w3.org/TR/xslt
European Regulators Group for Electricity and Gas, Position paper on smart grids no. e09-eqs-30-04, Berlin, 2010.
Georgakopoulos, D., Papazoglou, M.P. (Eds.), 2008]. Service-Oriented Computing. MIT Press, Cambridge, MA.
IBM, An architectural blueprint for autonomic computing, Whitepaper, 2005.
Ipakchi, A., Albuyeh, F., 2009]. Grid of the future. IEEE Power and Energy Magazine 7 (2), 52–62.
Jackson, M.O., Watts, A., 1999]. The evolution of social and economic networks. Journal of Economic Theory 106, 265–295.
Jackson, M., Wolinsky, A., 1996]. A strategic model of social and economic networks. Journal of Economic Theory 71, 44–74.
Juszczyk, L., Dustdar, S., 2010]. Script-based generation of dynamic testbeds for SOA. In: ICWS., pp. 195–202.
Kleinberg, J., 2008]. The convergence of social and technological networks. Communications of the ACM 51 (11), 66–72.
Kruk, S.R., Grzonkowski, S., Gzella, A., Woroniecki, T., Choi, H.-C., 2006]. D-FOAF: distributed identity management with access rights delegation. In: ASWC., pp. 140–154.
Ludwig, H., et al., Web service level agreement (WSLA) language specification, version 1.0, 2003.
Massoud, Wollenberg, B.F., 2005]. Toward a smart grid: power delivery for the 21st century. IEEE Power and Energy Magazine 3 (September (5)), 34–41.
Medjahed, B., Bouguettaya, A., 2005]. A dynamic foundational architecture for semantic web services. Distributed and Parallel Databases 17 (2), 179–206.

Novakouski, M., Simanta, S., Peterson, G., Morris, E., Lewis, G., Performance analysis of WS-security mechanisms in soap-based web services, Berlin, 2010.
Organization for the Advancement of Structured Information Standards (OASIS), WS-Security 1.1, 2006.
OPower Company, 2011. http://opower.com/
Osborne, M.J., Rubinstein, A., 1999]. A Course in Game Theory. MIT Press.
Psaier, H., Skopik, F., Schall, D., Dustdar, S., 2011]. Resource and agreement management in dynamic crowdcomputing environments. In: EDOC., pp. 193–202.
Reka, A., Barabási, 2002]. Statistical mechanics of complex networks. Reviews of Modern Physics vol.74, 47–97.
Schall, D., Truong, H.-L., Dustdar, S., 2008]. Unifying human and software services in web-scale collaborations. Internet Computing 12 (3), 62–68.
Schall, D., Truong, H.L., Dustdar, S., 2008]. Unifying human and software services in web-scale collaborations. IEEE Internet Computing 12 (3), 62–68.
Schall, D., Skopik, F., Psaier, H., Dustdar, S., 2011]. Bridging socially-enhanced virtual communities. In: SAC., pp. 792–799.
Schall, D., 2011]. A human-centric runtime framework for mixed service-oriented systems. Distributed and Parallel Databases, 1–28.
Skopik, F., Wagner, C., 2012]. Novel energy saving opportunities in smart grids using a secure social networking layer. In: COMPSAC., pp. 557–566.
Skopik, F., Schall, D., Dustdar, S., 2011]. Opportunistic information flows through strategic social link establishment. In: WI. IEEE., pp. 308–315.
Skopik, F., Schall, D., Dustdar, S., 2011]. Managing social overlay networks in semantic open enterprise systems. In: WIMS, pp. 50:1–50:12.
Sosnoski, D., 2009]. Java Web Services: The High Cost of (ws-)Security.
Story, H., Harbulot, B., Jacobi, I., Jones, M., Foaf+ssl: Restful authentication for the social web, last access: 2012. http://esw.w3.org/Foaf+ssl
University of Stratclyde, Energy demand profile generator (last access: November 2011), 2007. http://www.esru.strath.ac.uk
van Tilborg, H.C.A. (Ed.), 2005]. Encyclopedia of Cryptography and Security. Springer.
Wasserman, S., Faust, K., 1994]. Social Network Analysis: Methods and Applications, 1st ed., ser. Structural Analysis in the Social Sciences. Cambridge University Press, no. 8.
World Wind Energy Association, World wind energy report 2010, February 2011.
Yokoo, M., Conitzer, V., Sandholm, T., Ohta, N., Iwasaki, A., 2005]. Coalitional games in open anonymous environments. In: AAAI., pp. 509–514.

**Florian Skopik** joined the Austrian Institute of Technology (AIT) in 2011 and is currently working in the research program ICT Security, focusing on diverse security aspects of distributed systems and service-oriented architectures. Current research interests include secure smart grids and the security of critical infrastructures, especially in course of national cyber defense. Before joining AIT, Florian received a bachelor degree in Computer Science 2006, and master degrees in Computer Science Management and Software Engineering from the Vienna University of Technology in 2007. He was with the Distributed Systems Group at the Vienna University of Technology as a research assistant and post-doctoral research scientist from 2007 to 2011, where he was involved in a number of international research projects on enterprise collaboration. In context of these projects, he also finished his PhD studies on dynamic trust in social networks. Florian further spent a sabbatical at IBM Research India in Bangalore for several months. He published around 50 scientific conference papers and journal articles, and is member of various conference program committees and editorial boards.

Please cite this article in press as: Skopik, F., The social smart grid: Dealing with constrained energy resources through social coordination. J. Syst. Software (2013), http://dx.doi.org/10.1016/j.jss.2013.04.052