Arrowhead Compliant Virtual Market of Energy

Luis Lino Ferreira[1], Laurynas Siksnys[2], Per Pedersen[3], Petr Stluka[4], Christos Chrysoulas[5], Thibaut le Guilly[6], Michele Albano[7], Arne Skou[8], Cesar Teixeira[9], Torben Pedersen[10]
Instituto Superior de Engenharia do Porto[1, 5, 6, 7, 9], Aalborg University[2, 6, 8, 10], Neogrid[3], Honeywell[4]
{llf[1], chech[5], mialb[7], crdst[9]}@isep.ipp.pt; {siksnys[2], Thibaut[6], ask[8], tbp[10]}@cs.aau.dk, pdp@neogrid.dk[3],
Petr.Stluka@honeywell.com[4]

Abstract— **Industrial processes use energy to transform raw materials and intermediate goods into final products. Many efforts have been done on the minimization of energy costs in industrial plants. Apart from working on "how" an industrial process is implemented, it is possible to reduce the energy costs by focusing on "when" it is performed. Although, some manufacturing plants (e.g. refining or petrochemical plants) can be inflexible with respect to time due to interdependencies in processes that must be respected for performance and safety reasons, there are other industrial segments, such as alumina plants or discrete manufacturing, with more degrees of flexibility. These manufacturing plants can consider a more flexible scheduling of the most energy-intensive processes in response to dynamic prices and overall condition of the electricity market. In this scenario, requests for energy can be encoded by means of a formal structure called flex-offers, then aggregated (joining several flex-offers into a bigger one) and sent to the market, scheduled, disaggregated and transformed into consumption plans, and eventually, into production schedules for given industrial plant. In this paper, we describe the flex-offer concept and how it can be applied to industrial and home automation scenarios. The architecture proposed in this paper aims to be adaptable to multiples scenarios (industrial, home and building automation, etc.), thus providing the foundations for different concept implementations using multiple technologies or supporting various kinds of devices.**

Keywords— Service-Oriented Architectures, Smart Grids;

I. INTRODUCTION

Energy consumption and cost reduction are two important parameters of the sustainability of all human related activities, including industry. In the past few years, the rising of the *smart-grid* paradigm has enabled energy saving approaches based on controlling *how* and *when* energy is consumed [1].

The cost of energy, both its economic cost and its production impact on the environment, is a function of how the energy is produced and consumed. Until now, energy has been provided mainly by bulky plants, which are always on, plus a minor contribution was given by renewable energy sources. To assure the continuous energy supply during the "hot spots", i.e.: when the demand is higher, peaking power plants are used, which are fast to switch on and off, but, on the other hand, are much more expensive and polluting, since peaking power plants are designed to operate only for a limited time per year and thus many expensive optimizations on the energy production are not part of their design.

Thus, instead of considering "how" the industrial processes are implemented, we argue that the scheduling of energy consumption can reduce the energy costs of the industrial process. Our proposed approach can help to flatten out energy consumption and refrain, or at least minimize, the usage of the peaking power plants. In some industrial plants it is possible to temporarily reduce the most energy-intensive processes as is the case of smelters in alumina plants, which consume massive amounts of electrical energy. The smelting process can be interrupted for some time under assumption that the cryolite will be kept unfrozen. This gives the alumina plants a possibility to participate in the energy market. In other industrial plants, such as discrete manufacturing, the flexibility of operation is usually higher and cost of energy can be directly integrated into the plant optimization objective function.

We consider scheduling energy consumption by allocating it through a virtual market of energy, which communicates with appliances through flex-offers, the latter being the name of both, this approach, and of the formal structures encoding energy requests with associated flexibilities, and schedules. An efficient participation in the market relies on mechanisms based on high performance ICT for managing data and automatically scheduling energy consumption. Service Oriented Architecture (SOA) has the potential of providing high flexibility and loose coupling, needed for a scalable and robust solution. A SOA is basically a collection of software services and applications coordinating (orchestrating) their activities. SOA nowadays are evolving in order to cope with the challenging requirements for a real-time and high-security operation and interactions with resource constrained devices.

The flex-offer approach involves industrial energy *consumers*, *aggregators*, and a so-called *flexibility market*. Sensors collect energy consumption data from machines involved in industrial processes. Then, this data is transformed into device-specific energy requests that are encoded using flex-offers [2]. Flex-offers contain information about the consumption profile of the device and its time flexibility. Aggregators are specialized entities that join (several) smaller flex-offers into (a smaller number of) larger flex-offers. The aggregated flex-offers are sent to a market. The market minimizes total costs by scheduling the energy consumptions while respecting the constraints contained in the flex-offers (minimum/maximum power, earliest/latest start of energy consumption, etc.).

Finally, the plan with scheduled energy consumption is disaggregated by the aggregators to build up multiple individual plans, which are then sent to respective controllers that drive the machines involved in the industrial process accordingly.

The ARROWHEAD project [22] aims at providing an interoperable and adaptable SOA substrate to enhance industrial processes, to allow faster production cycles, energy savings, rapid reconfiguration of industrial processes, better data collection on physical events occurring during the industrial process, better interoperability with legacy systems from different vendors, re-engineering of the highest-level interaction between the systems for easier maintenance of the (software) systems themselves and simpler evolvability. Among other objectives, the ARROWHEAD project aims at applying the virtual market of energy concept, for a smarter yet automated scheduling of energy consumption.

In this paper we describe the architectural design of the ARROWHEAD Virtual Market of Energy, primarily focusing on its embedded part – the components of the Flex-Offer Agent (FOA), which generate and control the execution of flex-offers.

The remainder of the paper is organized as follows. Section 2 gives a short introduction to the flex-offer concept as developed by the EU FP7 MIRABEL [2] project; section 3 describes the overall architecture of the ARROWHEAD virtual market of energy; section 4 provides a detailed design of the Flex-Offer Agent architecture, whereas section 5 presents usage scenarios for the market. Finally, sections 6 and 7 present related work and conclusion.

II. THE FLEX-OFFER CONCEPT AND THE VIRTUAL MARKET OF ENERGY

A. The Flex-offer Concept

Flex-offer is the concept has been developed in the EU FP7 project MIRABEL [2]. It allows exposing demand and supply loads with associated flexibilities in time and amount for energy trading, load balancing, and other use-cases. Flex-offers are generic entities, and can accommodate various types of consumers (e.g., electric vehicles, heat pumps, household equipment, industry, etc.) and producers (de-charging electric vehicles, solar panels, etc.). It is currently undergoing a European standardization process through CEN/CENELEC CWA.

In its simplest form, a flex-offer specifies an amount of energy, a duration, an earliest start time, a latest end time, and a price, e.g., "I need 10 KWh over 2 hours between 1 AM and 5 AM, for a price of 0.35 DKK/kWh". A more complete example is shown in Fig. 1. In the figure, we represent the window of the consumption starting moment in red. The power requirements are expressed in variable-length intervals, during which minimum and maximum energy amounts are additionally specified (the blue color expresses the difference between maximum and minimum amounts). These limits are used to better accommodate a real situation, where the energy consumption exhibits variations in adjacent time intervals.

Flex-offers for supply are also possible, e.g., "I can deliver between 10 and 20 MWh, between 1 PM and 5 PM, for a price of 0.35 DKK/kWh". Note that the amount of energy represented in a flex-offers can be small, e.g., for household equipment, or large, e.g., for industry.
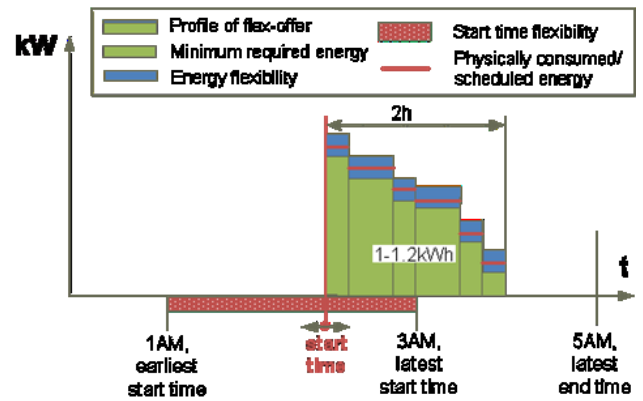

Fig. 1. Flex-offer example

Flex-offers can be aggregated and disaggregated irrespectively to a type of consumption or production they represent. Both aggregated and non-aggregated flex-offers can be mixed and dealt uniformly.

A flex-offer can be seen as a kind of "option" that a consumer/producer puts out on a market. The flex-offer may be rejected, for example if the price is not right. If the option is accepted, the flex-offer is given an initial schedule, e.g., the flex-offer is scheduled at 2 AM, and the consumer control system is notified. On the simplest case, the schedule is carried out as specified. However, one of the strengths of the concept only comes into play when things do not go as foreseen, for instance due to a sudden drop in wind energy. In this case, the flex-offer can be rescheduled, shifted to 3 AM, when the wind has returned.

B. Virtual Maket of Energy

For managing flex-offers, the TotalFlex [4] project proposes the use of the general Virtual Market of Energy system (see Fig. 2) that, by providing a set of Service Oriented Architecture (SOA) interfaces, interconnects several (existing and new) European Electricity Market Actors, elaborated below.

Prosumers are entities, aka. Distributed Energy Resources (DERs) that can both consume and produce electricity. Examples of Prosumers are residential houses, commercial buildings, manufacturing, and process industries. These generate flex-offers and consume schedules.

Aggregators are specialized entities capable of aggregating several (micro) flex-offers from Prosumers into larger (macro) flex-offers. It is also capable of disaggregating (macro) flex-offer schedules, e.g., received from the Electricity Market. An aggregator might be an integrated part of a Balance Responsible Party (BRP).

Balance Responsible Parties are European electricity market entities that secure the balance in a logical sub-domain within the grid, i.e. ensure that consumption is equal to production. It utilizes the aggregated flex-offers from Aggregators for an internal energy balancing and placing flex-offers on a so-called Flexibility Market for trading with other BRPs or Distribution System Operators (DSOs).

Distribution System Operators are entities responsible for uninterrupted supply of energy in the distribution grid. Flexibilities represented by flex-offers and offered on the market enable DSOs new ways to smoothen loads on the distribution grid by buying and then controlling the timing of loads.

Flexibility Market offers BRPs and DSOs the common place for trading flex-offers. It minimizes total costs by scheduling energy loads while respecting the constraints contained in the flex-offers (minimum/maximum power, earliest/latest start of energy consumption, etc.). Flexibility Market may also interface other traditional markets of energy.
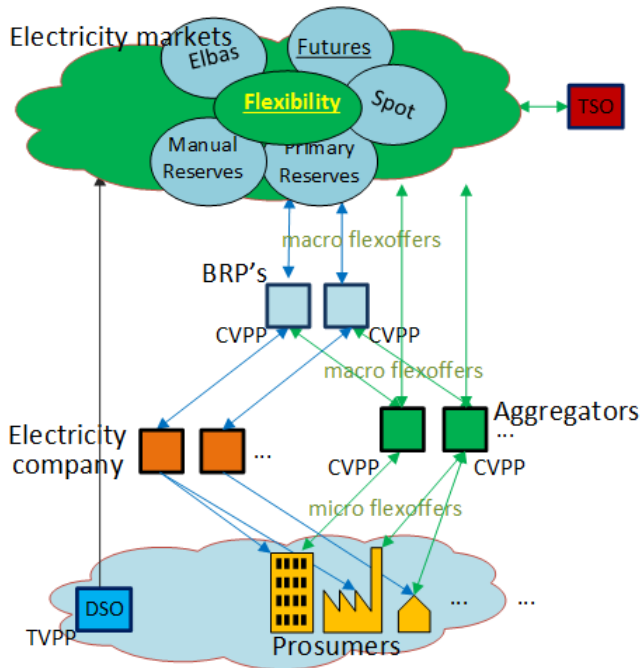

Fig. 2. Virtual Market of Energy main actors and operations.

By focusing on the interface between prosumers and Aggregators solely, the ARROWHEAD project further extends this generic Virtual Market of Energy architecture by proposing a number of software-based extensions which allow equipping various types of prosumers with the support for flex-offers in a generic and unified way. These extensions are presented next.

III.   SYSTEM ARCHITECTURE AND COMMUNICATION INTERFACES

The block diagram in Fig. 3 introduces the main actors and shows the operation of Arrowhead's Virtual Market of Energy. The architecture being proposed in this paper is built upon the Arrowhead Common Framework, which consists of a number of core services that support the development of generic SOA systems. The Arrowhead Common Framework thus acts as an enabler for systems from different areas (e.g.: industrial automation, airplane maintenance, energy production, home automation, smart grids) to facilitate their interaction with each other and exchange information. This multi-area approach can enable large savings in terms of energy, efficiency and interoperability.

The proposed architecture is structured upon five modules, where three belong to the core Arrowhead framework – Service Registry, Authorization, and Orchestration, and the other two modules exchange business logic data – the Aggregator and the Flex-offer Agent (FOA).
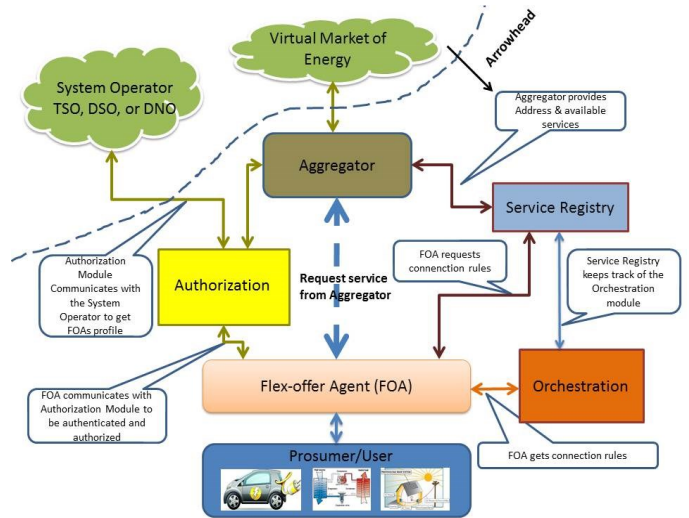

Fig. 3. High level architecture for the Virtual Market of energy

Flex-offer Agents are basically software modules that offer the main functionalities to support the flex-offer concept. Its architecture (described in detail in Section 4) already provides functionalities for getting information about the power consumption profile of specific devices, generate a flex-offer, and control the execution of a scheduled flex-offer. The design of this software also provides the necessary means to adapt to any platform by developing adequate interfaces among: i) local (in Section IV we show that the FOA design supports several types of implementations) and remote FOA modules; ii) with the controlled devices' hardware; iii) with other needed devices (e.g. a remote power meter) through a network; and iv) with external services (e.g. to obtain weather forecasts). Flex-offer Agents are very flexible, its design allows for its total implementation to be running on a single device or distributed through several devices. As an example, the company providing the flex-offer service might give the user a specific hardware device, exclusively for the FOA, alternatively the FOA modules can be executed on existing devices on the prosumer premises. The main objective of the design being presented in this paper is to enable a high level of flexibility on the FOA implementation.

The Aggregators work by receiving flex-offers from FOAs, aggregating with flex-offers from other FOAs, into larger macro flex-offers and placing them to the Virtual Market of Energy. Note that only flex-offers larger than certain amount can be negotiated on existing electricity

markets. Afterwards, the Aggregator receives a response from the Virtual market of Energy, disaggregates the response and sends a consumption schedule to the FOA. Several types of Aggregators might exist, and some Aggregators can be more specific for the control of electric motors while others can be more adequate for the control of heating systems. Additionally, choosing the most adequate Aggregator also depends on the geographical area.

To obtain the address of a proper Aggregator, a FOA uses the Service Registry module, to register itself, and the Orchestration module (to obtain an Aggregator that matches its criteria), both services are provided by the Arrowhead framework. The Service Discovery service has already two implementations, one using DNS Service Discovery (DNS-SD) [6] and another using Berkeley Internet Name Domain (BIND) [7].

Security services are provided through the Authorization module. This module is managed by the electric System Operator, with whom the prosumer is supposed to have signed a contract to benefit from flex-offers. Authentication is implemented by a combination of Public key Infrastructure (PKI) and X.509 certificates over REepresentational State Transfer (REST) [8] protocol. All communications can be encrypted by using XMPP [5] over Transport Layer Security (TLS). Therefore, the FOA and the Aggregator must first obtain its security certificates through the Authorization module. Only after, the modules are allowed to connect.

The FOA obtains the address of the Orchestration module using the Service Registry module which stores information regarding all of the available service producers, their location and characteristics. In order to be discovered any Arrowhead compliant module must first communicate with the Service Registry module and register itself by providing information regarding its address and its offered services. That is also the case of the Orchestration module.

The Orchestration module contains information about connection rules and system configuration, thus allowing the FOA to obtain the address of the most adequate Aggregator for the devices controlled by it. The Aggregator or a system administrator should previously have configured the connection rules for the Aggregator considering its geographical location and the kind of devices that it is enabled to receive flex-offers from. This feature allows FOAs to connect to Aggregators which implement specialized algorithms capable of optimizing the results for specific types of devices being controlled. The Orchestration has been implemented using REST.

The communication between the Aggregator and the FOA is implemented using XMPP, and exploits the existing Arrowhead framework services and their specific protocols to establish connection.

The main advantage of XMPP relies on its capabilities to support the Publish/Subscribe communication paradigm, which provides an asynchronous and highly scalable many-to-many communication model. The resulting decoupling between Publishers and Subscribers, in time, space and synchronization, simplifies the implementation of its associated software. Additionally, XMPP is also in a process of being standardized as a protocol for the control of Demand Response applications for OpenADR [9] and on the ISO/IEC/IEEE 21451-1-4 [10] standards.

IV. FLEX-OFFER AGENT ARCHITECTURE

The Flex-Offer Agent is a critical component for the success of the flex-offer concept; therefore it requires a careful design in order to be used in multiple usage scenarios. In particular the FOA should allow the device's control algorithms to be implemented by the system builder. Also, the algorithms used to generate flex-offers should be adapted to any particular case, e.g. by considering different configuration parameters, which depend on the device type. The proposed architecture allows for a local or distributed implementation of the Flex-Offer Agent.

A. Architectural Components

The architecture of the Flex-Offer Agent has been defined using an object oriented and component-based methodology, which resulted in the component diagram shown in Fig. 4.

The first step in defining the system architecture has been to analyze the problem domain. By doing so, we were able to identify three main elements, the Distributed Energy Resource (DER), the FlexOfferManager, and the Aggregator.

The DER represents an element that consumes or produces energy, usually a specific device using the flex-offer concept. A DER representation must contain the necessary information to generate its flex-offers. Since this information varies depending on the type of DER, specialization of DER can be defined, such as a heat pump or a collection of heat pumps.

Note that a DER can itself contain another DER. We can take as an example a set of sixty heat pumps that are combined to be represented as a unique DER, each heat pump being a DER as well. It is assumed that for any type of DER, a profile (using the Profile object) can be defined, that contains the current configuration of the DER. This profile could for example contain the comfort settings for a heat pump or the maximum temperature variation for a aluminum smelter.

A common feature of any DER is that they consume and produce energy based on a schedule. Therefore, we attach to a DER a schedule (Schedule object), representing the future plan for its energy consumption/production, as well as measurements, that represent its past consumption. This plan is then executed by a Controller on the ControllerManager, which sends the appropriate control signals to the physical DER. The connection between the DERManager and the ControllerManager can be done locally or through an adequate interface.

Finally, to each DER is attached a set of flex-offers produced by it, as well as a set of Flex-Offer Schedule objects associated with its schedule.

The FlexOfferManager manages a set of DERs and provides the main functionalities of the system. Besides, a FlexOfferManager is associated with an Arrowhead subsystem that enables it to advertise its services through the Arrowhead service discovery and manage security.
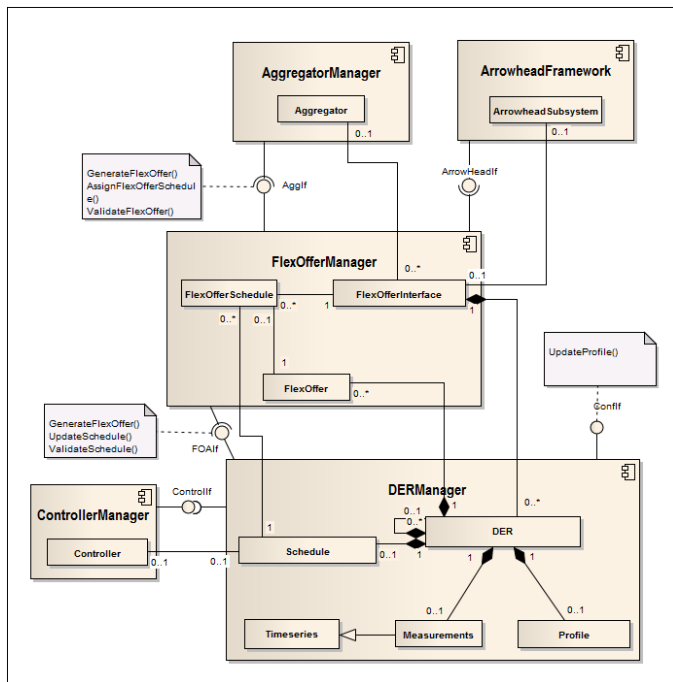

Fig. 4.   Flex-Offer Agent architecture.

Finally, the last element of the system is the Aggregator. As already mentioned, it aggregates a set of flex-offers from different FOAs to make it possible to be place a bid on existing energy markets. Note that low power devices (in households, as an example) do not consume enough energy to be allowed to place orders on the energy markets. An Aggregator can also be specialized into certain kinds of devices, for example on heat pumps. Such specialization allows it to run aggregation and disaggregation algorithms which are more efficient with those kinds of devices. Also note that an Aggregator is supposed to be associated with one or more FOAs.

For the whole system it is possible to identify four main functionalities: i) the generation of flex-offers; ii) the assignment of flex-offer schedules; iii) the validation of the flex-offer schedules execution (i.e., if the device has consumed the amount of energy and on the timings stated on the received flex-offer schedule) and finally; iv) the configuration of the DERs. The first three functionalities are required by the Aggregator, since it needs to get flex-offers from FOAs, assign flex-offer schedules and validate that they were executed correctly. The configuration of a DER can be done by different parties, depending on the type of DER.

The FOA is the logical entity that incorporates the FlexOfferManager, the DERManager, the ControllerManager and the ArrowheadFramework, with which an Aggregator communicates. The functions required by the Aggregator are provided by the FlexOfferInterfaces. However, as it is a DER representation that contains the necessary information to generate a flex-offer (to be detailed in Section IV.B), the function to generate flex-offers is implemented by the DER. The FlexOfferManager will thus require the DER to generate its flex-offers, which are sent to the Aggregator through the Aggregator interfaces (AggIf). The request triggering the generation of a flex-offer can be initiated by the Aggregator itself or internally by the DER. After an initial validation of a received flex-offer, an Aggregator places it on the market.

The FlexOfferManager also provides the functionality to assign flex-offer schedules to a Controller after receiving a response to its flex-offer request. A FlexOfferManager then updates its DER schedules. Finally, a FlexOfferManager provides an Aggregator with the possibility to validate if past flex-offer assignments were executed correctly. The actual validation is performed by the aggregator, after receiving the information necessary for its validation, i.e. the power consumption of the device during the execution of a flex-offer.

To improve the separation of concerns, we dispatch each logical entity into components. We group the three main elements of the system into three components, an AggregatorManager to handle Aggregator functionalities, a FlexOfferManager for interfacing a overall control functionalities, and a DERManager for DER functionalities. The Controller is also assigned to its own component manager, the ControllerManager, to adapt to different physical architectures. An AggregatorManager and a FlexOfferManager communicate through an AggIf interface, while a FlexOfferManager and DERManager communicate through a FOAIf interface implemented by a DER. Finally, the DERManager provides a ConfIf interface for updating its profile, and provides the DER schedule to the Controller through a ControlIf interface.

By defining the FOA architecture as decoupled components, we aim at facilitating different deployment strategies, either local, distributed or a mix of both, and concrete interaction mechanisms to satisfy different constrains, depending on the type of DER. Therefore, this architecture allows each manufacturer of a device to implement its own control algorithms on its own "control box". The control algorithms might vary from sending simple turn on and turn off commands to more elaborate ones, which depend on obtaining readings from sensors and sending commands to actuators. It is also important to note that this architecture can be adapt to proprietary open implementation, supported or not supported by the Arrowhead framework.

B.   *Flex-offer Generation*
As mentioned earlier, the DER representation contains all the necessary information to generate flex-offers. For the simplest DERs (e.g. dishwashers), the DER representation captures only the pre-defined flex-offer template with concrete manufacturer-given energy amounts specified in the profile (see Fig. 1). This template is then instantiated based on an active user profile, e.g., specifying latest end time.

For more complex DERs (e.g., multiple heat pumps), the DER representation may capture i) the current state of the DER; ii) a detailed mathematical model of DER; and iii) other context information (e.g., weather-forecasts and spot-prices). The generation of flex-offers in such cases is a complex optimization problem (based on the DER representation) which needs to be solved in order to obtain the expected energy amounts and flexibilities to be represented in flex-offers.

When generated flex-offers are assigned by the Aggregator, the Aggregator sends back the respective flex-offer schedules, which have to be mapped (converted) to schedules for a DER. Note that the generation of these schedules depends on the DER operation characteristics. However, the DER schedules might not always be possible to be build (from the flex-offer schedules) without violating constraints, and for the Controller to follow. Therefore, the flex-offer generation procedure has to be designed so that generated flex-offers results into flex-offer schedules that can always be mapped to DER schedules and followed by a DER under study. Consequently, the DER representation has to capture up-to-date information so that the generation of "schedulable flex-offers" is feasible.

C. *Interfaces*

To enable the communication between components, a set of interfaces have been defined. The AggregatorManager provides the AggIf interface which delivers a set of services to interact with the FlexOfferManager. These services are used by the FlexOfferManager to send a flex-offer request to the Aggregator, to receive a flex-offer schedule from the Aggregator and to report on the successful/unsuccessful execution of a flex-offer schedule (i.e. if the device has consumed the amount of energy and on the timings stated on the associated flex-offer schedule).

The FlexOfferManager component provides an interface FOAIf which allows the communication between FlexOfferManager and DERManager. This interface implements the needed features for: i) the DERManager to generate a flex-offer request, which is processed by the FlexOfferManager; ii) the FlexOfferManager to request the validation of a flex-offer schedule; iii) assigning a DERManager a flex-offer schedule.

A DERManager provides a ConfIf interface that allows the definition of a DER profile, which might be accessed through a user interface (e.g., a web browser).

A DERManager and a ControllerManager communicate through the ControlIf interface. This interface is application specific, enabling the DERManager to send commands to actuators (e.g., to execute a flex-offer plan) or collecting data from sensors.

Finally, the ArrowHeadIf interface allows the FlexOfferManger to connect with the Arrowhead Framework core services.

V. USAGE SCENARIOS

A. *Industrial Case – Aluminum Plant*

Metal processing plants, such as aluminum or steel plants, are large consumers of electricity and, at the same time, they have flexibility for making trade-offs between plant production objectives and electricity consumption. This can have a strong impact in costs and on the environmental footprint of the factory.

In the case of aluminum plants it is possible to let the smelters cool down for some time, while respecting temporal constraints given by the process where the cryolite must be kept unfrozen.

Possible solution for the application of the flex-offer concept to an aluminum plant can be implemented by connecting the plant-level energy management and optimization systems to the Aggregator through the Flex-Offer Agent (FOA). The split of responsibilities between individual components can be made as follows:

Plant-level Energy Management System (EMS) has responsibility for real-time monitoring of energy consumption and its functions can be extended towards interacting with the process-level control systems in order to adjust production schedules and targets for individual smelters. In our scenario, EMS will monitor the status of smelters and quantify available flexibility for interruption of the process. This will take into account process constraints and will have to be approved by the responsible plant personnel. EMS will communicate all approved flexibilities the by implementing its own specific DERManager module. Then the FlexOfferManager implements the interface between the industrial plant and the Aggregator, correctly encoding the flexibilities identified inside the plant allowing for further processing by the Aggregator.

A specific implementation of the Aggregator can specialize on flex-offers generated by aluminum smelters and then communicated back to the FlexOfferManager which assigns the received flex-offer schedule to the DERManager, which will have to communicate with the plant-level EMS and with the ControllerManager to execute the Schedule, generated by the DER.

B. *Residential Case – Heat pump*

This section describes an example showing how to apply the flex-offer concept to a residential house equipped with a heat pump. In this example (Fig. 5) the residential heat pump connect with the FOA module, which, by its turn, connects with the Aggregator.

In this case, we are also assuming the usage of a specialized Aggregator capable of controlling several houses with heat pumps and able to pool a large number of devices, logically grouped into a Virtual Power Plant (VPP). This kind of grouping enables access to a large variety of energy markets (e.g. day-ahead and intra-day), leading to higher possibilities of issuing successful flex-offers, where the main

objectives are to reduce the energy cost associated with the heat pump and to reduce its environmental impact.

The implementation of the DER is complex since it needs internal data about the heat pump and also needs to connect to external data sources. Flex-offers are generated based on: i) a model of the system; ii) past heat pump power measurements; iii) weather forecast; iv) operational constraints (e.g. user comfort levels); and v) heat pump settings. These flex-offers are then sent to the Aggregator, which aggregates flex-offers from different devices, and potentially waits until being able to construct a flex-offer with the necessary requirements to be sent to the energy market.
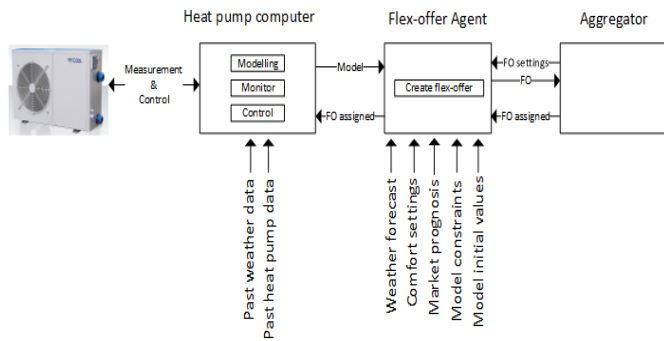

Fig. 5. Flex-offer enabled heat pump architecture

Finally, it is important to note that a heat pump FOA connects to an Aggregator based on a set of rules, implemented by the Arrowhead framework Orchestration component, which can be its geographical location and device type.

VI. RELATED WORK

There have been numerous research efforts with respect to the vision of how Smart Grids and SOAs can join forces in order to deliver a really powerful Smart Grid architecture.

In [11] the need for a more flexible information system to let utilities operate more efficiently is presented. The authors stress out the need for a more flexible information system, since many previous attempts in the power grid sector were not really functional. The integration were made manually or in a time consuming way. The presented solution is also part of the IEC-61970 / IEC-61968 standard, and uses the Common Information Model (CIM) for devices and objects representation. It can be characterized as a predecessor of modern SOAs.

The issues of data integration, platform sharing and how SOA can address and sort out these issues are presented in [12]. Implementing a web service SOA based on IEC standards provides many advantages as can be seen in [13-14]. Advantages such as: easier integration between companies and systems, and easier reuse of existing infrastructure by offering reduced configuration costs. This is feasible since the approach leads to a more automated way for reconfiguring the system.

SOA is indicated as vital in the Smart Grid. Its central role is present in the household environment, in order to easily succeed interaction among heterogeneous devices. An example can be found in [15]. In this example the way of how a service-based architecture can be useful in a Smart House is presented in detail. Internet and the use of web-services are considered to be the means to enable interaction among the smart devices of the house and the supply companies, in order to exchange bids and other related functionalities (e.g. Demand-Response functionality). Therefore SOA is a cornerstone to succeed interaction between smart devices and the market of energy. An ontology is also needed for different actors to seamlessly interact. Such an ontology is presented in [16]. The term Service Oriented Grid (SOG) is used by the authors to the above ontology based approach. Semantics are important in a SOA based approach in order to achieve interoperability across the grid of energy.

Karnouskos in [17] is proposing a SOA-ready feature for each Smart Grid device. In this way each actor will be in position to advertise what services he can provide and at the same time he will be in position to dynamically discover services that other actors care providing by using Web Service Dynamic Discovery specifications. By adopting the aforementioned approach, better energy management and efficiency is succeeded. Thus the usage of web services for the collaboration among different actors, like consumers, energy resources, and energy markets, is proposed.

Coax and Considine [18] point out the importance of collaboration or else loyal cooperation in Smart Grids. Tight collaboration among the many actors involved (even hostile to each other, due to financial transactions) is believed to be the driving force for an efficient implementation of the Smart Grid. Characteristics like transparency, composition, extensibility and loose coupling are also present in the NIST and are the basis in SOAs.

Possible scenarios for participation of the industry in the smart grid environment were studied and documented in the following papers.

Alcoa Power Generation, Inc. participates in the wholesale market by providing regulation of up to 25 MW as an ancillary service through control of smelter loads at Alcoa's Warwick Plant [19], and similar scenario can be applied at other alumina plants.

An optimal control model for the load shifting problem in energy management and its application in a South African colliery [20] highlighted the possibility to shift the loads by the dynamic control of conveyor belts.

A mixed-integer approach to dynamic minimization of electricity costs by scheduling the loads and satisfying the process, production, and maximum demand constraints in a steel rolling mill indicated a possibility to save more than 5% of electricity costs [21].

VII. Conclusions

Dynamic, heterogeneous and distributively owned resources and service environments present unique challenges for resource and service representation, allocation and management.

In this paper we present the flex-offer concept, which allows any kind of electric energy powered device, capable of allowing some time flexibility on its power consumption to benefit from reduced energy prices, if its consumption is executed according to the convenience of the overall electric grid.

One practical implementation of the concept is being done within the aims of the Arrowhead project, where the flex-offer concept is integrated a broad field of applications, structured into a multiprotocol platform – the Arrowhead Framework. This framework, already available in its first generation prototype, supports some of the most important services for implementation of a Service Oriented Application. We show in this paper how this framework services can be used to fulfill the needs of the flex-offer concept.

Finally, we propose a flexible and highly adaptable architecture to implement the Energy Market, which is based on two main actors: Aggregators and Flex-Offer Agents. The Aggregator is capable of receiving flex-offers related to specific devices and of handling the placement of this request, on an aggregated form, into the real energy markets. The Flex-Offer Agent generates flex-offers based on data from the devices and on prediction models, sends them to the Aggregator and executes the schedules send by the Aggregator. We also highlight how the concept can be used on an aluminum or steel industrial scenario and on the control of heat pumps on several households.

References

[1] M. Albano et al, "The ENCOURAGE ICT architecture for heterogeneous smart grids", in IEEE EUROCON 2013, July 1-4, Zagreb, Croatia, July 2013.

[2] M. Boehm, L. Dannecker, A. Doms, E. Dovgan, B. Filipic, U. Fischer, W. Lehner, T. B. Pedersen, Y. Pitarch, L. Siksnys, and T. Tusar, "Data management in the mirabel smart grid system," in EnDM, 2012, pp.95-102.

[3] O. Abarrategui, J. Marti, A. Gonzalez, "Constructing the Active European Power Grid", Proceedings of WCPEE09, Cairo, Egypt, 2009, pp. 1-4.

[4] TotalFlex project, link: http://www.totalflex.dk/

[5] Extensible Messaging and Presence Protocol, link: http://www.ietf.org/rfc/rfc3920.txt

[6] E. Guttman, "Autoconfiguration for IP Networking: Enabling Local Communication", IEEE Internet Computing 5 (3), 2001, pp. 81–86.

[7] D. B. Terry, M. Painter, D. W. Riggle and S. Zhou, "The Berkeley Internet Name Domain Server", Proceedings of USENIX Summer Conference, Salt Lake City, Utah, 1984, pp. 23-31.

[8] R. T. Fielding, R. N. Taylor, "Principled Design of the Modern Web Architecture", ACM Transactions on Internet Technology (TOIT) (2): 2002, pp. 115–150.

[9] OpenADR Alliance, link: http://www.openadr.org/.

[10] "ISO/IEC/IEEE P21451-1-4 Standard for a Smart Transducer Interface for Sensors, Actuators, and Devices based on the eXtensible Messaging and Presence Protocol (XMPP) for Networked Device Communication," Available online: http://wiki.xmpp.org/web/Tech_pages/IoT_Sensei, accessed April 2014.

[11] D. Becker, H. Falk, J. Gillerman, S. Mauser, R. Podmore, and L. Schneberger, "Standards-based approach integrates utility applications," IEEE Computer Applications in Power, vol. 13, no. 4, 2000, pp. 13–20.

[12] W. Kehe, S. Qianhui, D. Cheng, and Y. Haiyani, "The research of power grid data integration and sharing platform based on soa," Computer Science and Engineering, 2009. WCSE '09. Second Int. Ws on, vol. 1, pp. 106 –109, 2009, pp. 28-30.

[13] R. Mackiewicz, "The Benefits of Standardized Web Services Based on the IEC 61970 Generic Interface Definition for Electric Utility Control Center Application Integration," in IEEE Power Systems Conf. and Expo., 2006, pp. 491–494.

[14] Q. Chen, H. Ghenniwa, and W. Shen, "Web-services infrastructure for information integration in power systems," presented at the IEEE Power Eng. Soc. General Meeting, Montreal, QC, Canada, Jul. 2006

[15] C. Warmer, K. Kok, S. Karnouskos, A. Weidlich, and D. Nestle, "Web services for integration of smart houses in the smart grid," in Grid-Interop 2009, 2009, pp. 1–5.

[16] T. Considine, "Ontological requirements of the service oriented grid," in Grid-Interop - The road to an interoperable grid, Atlanta, Georgia, USA, November 2008.

[17] S. Karnouskos, "The cooperative internet of things enabled smart grid," in Proceedings of the 14th IEEE International Symposium on Consumer Electronics (ISCE2010), June 07-10, Braunschweig, Germany, June 2010.

[18] W. T. Cox and T. Considine, "Architecturally significant interfaces for the smart grid," in Grid-Interop - The road to an interoperable grid, Denver, Colorado, USA, 17-19 November 2009.

[19] D. Todd, M. Caufield, and B. Helms, "Providing Reliability Services through Demand Response: A Preliminary Evaluation of the Demand Response Capabilities of Alcoa Inc.", ORNL report, January 2009.

[20] A. Middelberg, J. Zhang, and X. Xia, "An optimal control model for load shifting – With application in the energy management of a colliery", Applied Energy 86 (2009), pp. 1266-1276.

[21] S. Ashok, "PLC-based Load Management in Steel Rolling Mills", Energy Engineering 107 (2010), pp. 38-53.

[22] Arrowhead Website, http://www.arrowhead.eu/.