

Article

## Applying a Dynamic Resource Supply Model in a Smart Grid

Kaiyu Wan <sup>1,\*</sup>, Yuji Dong <sup>1,2</sup>, Qian Chang <sup>1</sup> and Tengfei Qian <sup>1</sup>

<sup>1</sup> Department of Computer Science and Software Engineering, Xi'an Jiaotong-Liverpool University, Suzhou 215123, China; E-Mails: yuji.dong@liverpool.ac.uk (Y.D.); qian.chang10@student.xjtlu.edu.cn (Q.C.); tengfei.qian10@student.xjtlu.edu.cn (T.Q.)

<sup>2</sup> Department of Computer Science, University of Liverpool, Liverpool L69 3BX, UK

\* Author to whom correspondence should be addressed; E-Mail: kaiyu.wan@xjtlu.edu.cn; Tel.: +86-512-8816-1507.

Received: 9 June 2014; in revised form: 29 August 2014 / Accepted: 5 September 2014 /

Published: 22 September 2014

---

**Abstract:** Dynamic resource supply is a complex issue to resolve in a cyber-physical system (CPS). In our previous work, a resource model called the dynamic resource supply model (DRSM) has been proposed to handle resources specification, management and allocation in CPS. In this paper, we are integrating the DRSM with service-oriented architecture and applying it to a smart grid (SG), one of the most complex CPS examples. We give the detailed design of the SG for electricity charging request and electricity allocation between plug-in hybrid electric vehicles (PHEV) and DRSM through the Android system. In the design, we explain a mechanism for electricity consumption with data collection and re-allocation through ZigBee network. In this design, we verify the correctness of this resource model for expected electricity allocation.

**Keywords:** cyber-physical system; dynamic resource supply model; service-oriented architecture; smart grid; web service; ZigBee network

---

### 1. Introduction

Cyber-physical systems integrate physics-based and digital world models. A seamless integration of these two worlds will contribute immeasurably to a broad range of industries, including energy, infrastructure, healthcare, manufacturing and military [1–3]. Although many researchers have been working on different aspects of CPS design in these application domains, there are many open

challenges in integrating science and technology foundations towards design methodology, security and system composition. One of them is resource generation and allocation in CPS. In traditional computer science applications, the operating system manages hardware and software resources to handle all computation [4]. It may use resource models, such as RDF (Resource Description Framework) and REST (Representational State Transfer) [5–7]. In CPS, resource is not restricted to traditional computation and memory resources. Resource, in a generic sense, denotes any entity that is relevant to both the production and consumption of a service. In a CPS, physical devices are resources, which are hence first class entities. Services may be either generated or consumed by physical devices, which might, in turn, be consumed by cyber computational resources, such as communication protocols. Software services may be generated by the computational resources that reside in either a static or dynamic host computer in a CPS network and may be consumed by other physical devices to make changes in the environment. In general, a CPS resource might offer many services; a CPS service might require several resources; a CPS resource might use other resources; and a complex service in CPS may be produced by combining several services and resources. Thus, the service-oriented view of a CPS is more rich and complex than the service-oriented view for traditional business applications, as discussed in the SOC literature [8].

In this paper, resource generation and allocation mechanisms in a CPS are first investigated. The “resource supply” is investigated separately, and a basic resource supporting model is proposed to deal with resource generation and supply. In Section 2, we discuss the dynamic resource supply model (DRSM). It is built from both the description and utilization points of view. Resources in the CPS are described by introducing two concepts: Resource Supply Description Template (RSDT) and Resource Instance Notation (RIN). RSDT is used to describe the resources with static properties, and the RIN is used for describing the dynamic changing of attributes. The management of resources is handled by a resource operating decision machine (RODM). In Section 3, the related mechanisms are explained by illustrating how DRSM can be used in a CPS with a three-layer service-oriented architecture. In Section 4, DRSM is applied to a smart grid environment in order to achieve a cost-effective electric power distribution, whose twin goals are wireless monitoring and optimized power distribution. Section 5 discusses the mechanism and design of the interaction between plug-in hybrid electric vehicles (PHEV) and DRSM through the Android system. Section 6 discusses a design and a mechanism to automatically collect data on electricity consumption and to re-allocate power supply through ZigBee network. In Section 7, our work is summarized and further work is proposed.

## 2. Resource Model

One of the most important challenges in CPS is to conjoin abstractions in modelling physical processes with abstractions in computing. Most of the large system builders just simply treat the design issue of CPS, which involves capturing the interaction between the fundamentally different nature of physical and computational characteristics as a management problem instead of getting support from science or engineering [9]. To fill in the gap between cyber and physical services, RSDM tries to provide a separate layer for modelling different types of resources. Below, we will explain in detail this dynamic resource supply model, which includes both the resource descriptions and resource allocation mechanisms.

### 2.1. Resource Description

Since the resources accessed by the CPS components, both physical and cyber, could be of different types, we regard the CPS as a complex heterogeneous environment. To reduce the complexity of handling resources in CPS, a resource accessed or produced by CPS should declare some of its significant properties in order that property-based matching and management are possible. In our model, Resource Supply Description Template (RSDT) and Resource Instance Notation (RIN) describe the resources and their properties. In a CPS, the resource provider (RP) will generate and hold the RSDT as a contract for other systems. If one RP decides to join the CPS and support resources, the RSDT will register into the middleware, and this middleware has the methods and rights to monitor resource supply conditions. While the CPS is running by the support from RPs as described at RSDTs, the real-time resource supply situation can be indicated by RIN.

The RSDT is based on the Resource Description Template (RDT), which was proposed in [10]. In [11], the resources in a CPS are categorized into several different types, including human resources, biological resources, natural resources, man-made resources and virtual resources. The description of one type of resource is expressed as an XML version model-based formal specification. The resource providers can modify their RDTs to construct new RDTs and publish them. This process is independent from the final use of resources when a service is provided by a service provider. However, that model is tightly bound to assist a three-layer service-oriented cyber-physical system, and the RDT is just a description at the physical layer. When the resources are consumed in the system, many resource attributes in the RDT are not related to the running system. As shown in the robot example in [10], the availability notation is necessary, because if the resource requester is not in China, the robot resource located in China is not available. However, when the robot is being used in the system, the real location of the robot does not affect the system. To solve this problem, in our new resource model, the description can be either from the resource provider or the resource itself. The RSDT will describe the static properties, and the RIN will handle the changing attributes. It will be published by the resource provider to describe the resource information.

**Table 1.** Resource Supply Description Template (RSDT).

|   |  |
|---|--|
| <b>Template Holder</b>                      | the resource provider                              |
| <b>Template Unify ID</b>                    | the identity of this template                      |
| <b>Resource Name</b>                        | the provided resource name                         |
| <b>Characters</b>                           | to help the system decide how to use this resource |
| <b>Attributes</b>                           | can be inherited to the resource instances         |
| <b>Cost</b>                                 | cost per unit                                      |
| <b>Availability</b>                         | a set of constraints like time, location and etc.  |
| <b>Legal Rules for Supply</b>               | URI to a website                                   |
| <b>Other Resource in the Context of Use</b> | a set of context resource dependencies             |

Table 1 indicates a basic RSDT, and the semantics of it is given below:

(1) The template holder is the resource provider who publishes this RSDT. This holder is responsible for providing the resources described in this template.

(2) The template unify ID is the identity of this RSDT. It is defined as a number, and an IPv6 address is used to express this identity.

(3) The section “characters” contains many properties, which will be used by the system to decide how to use this kind of resource. For instance, the resource might be continuous, so the system has to quantize its information to calculate the supply and predicate further requirements. All characters will be read by the resource operating decision machine (RODM) and register at this machine. The decision about how to use the resources generated from the RSDT is decided by the RODM.

(4) The section “attributes” contains all attributes that will get a real value when the system is running. The template does not have real-time values, so when the resources are consumed in the system, the influenced attributes will be inherited by the RIN from all attributes described in the RSDT. Thus, the system can dynamically determine the resource supply situation by monitoring the RINs.

(5) The section “cost” is used for stating the price for a unit of resource.

(6) The section “availability” states the constraint related to the availability of this type of resources. As an example “this resource can only be provided on weekdays” restricts the resource availability for weekdays.

(7) The section “legal rules for supply” enumerates the legal aspects binding the availability and use of the resource.

(8) The section “other resource in the context of use” declares the types of other resources required to use this resources.

**Table 2.** RSDT published by a wind power plant in a power grid.

|   |  |
|---|--|
| <b>Template Holder</b>                      | XYZ Wind Power Plant   |
| <b>Template Unify ID</b>                    | 2001:0db8:85a3:08d3:1319:8a2e:0370:7344  |
| <b>Resource Name</b>                        | Alternating Current  |
| <b>Characters</b>                           | continuous resource<br>discretized by second<br>passively get resource<br>estimate<br>predicate          |
| <b>Attributes</b>                           | Inherited: Voltage<br>Inherited: Current<br>Inherited: Frequency<br>Inherited: Phase<br>Inherited: Power |
| <b>Cost</b>                                 | 5 U.S. cents/kWh   |
| <b>Availability</b>                         | Location: YYY (longitude, latitude)<br>Time: 80% supply/per year   |
| <b>Legal Rules for Supply</b>               | URI to <a href="http://www.XYZWindPowerPlant.com">www.XYZWindPowerPlant.com</a>                          |
| <b>Other Resource in the Context of Use</b> | electricity transformers<br>electricity transmission group<br>electricity protection group               |

In Table 2, we give the RSDT model for the “alternating current” resource supported by a wind power plant in a power grid.

In a dynamic resource supporting environment, the attributes of resources usually change at any moment. Therefore, the template published by the resource provider is not suitable for expressing the changing attributes and supporting the related operations. The RIN uses a simple key-value method to indicate the values at run-time. In order to make RIN mechanism as simple as possible, we have designed a resource operating decision machine (RODM). RODM will manage context information, and other associations, and carry out computations for related decision making. To improve the efficiency of dynamic resource support, we let the RIN be initialized by the resource itself at run-time. Thus, the latency time is decreased. Table 3 illustrates an instance of a resource generated from the RSDT shown in Table 2.

In this RIN sample, the resource has a unique ID, and five other attributes are inherited from the RSDT.

**Table 3.** A Resource Instance Notation (RIN) sample generated from the RSDT.

|                      |                              |
|----------------------|------------------------------|
| <b>Resource Name</b> | Alternating Current          |
| <b>Resource ID</b>   | 2013.12.1.1546               |
|                      | Voltage: 6.3 kV              |
|                      | Current: ?                   |
| <b>Attributes</b>    | Frequency: 50.02 Hz          |
|                      | Phase: correct               |
|                      | Power: 25,000 kilowatts/hour |

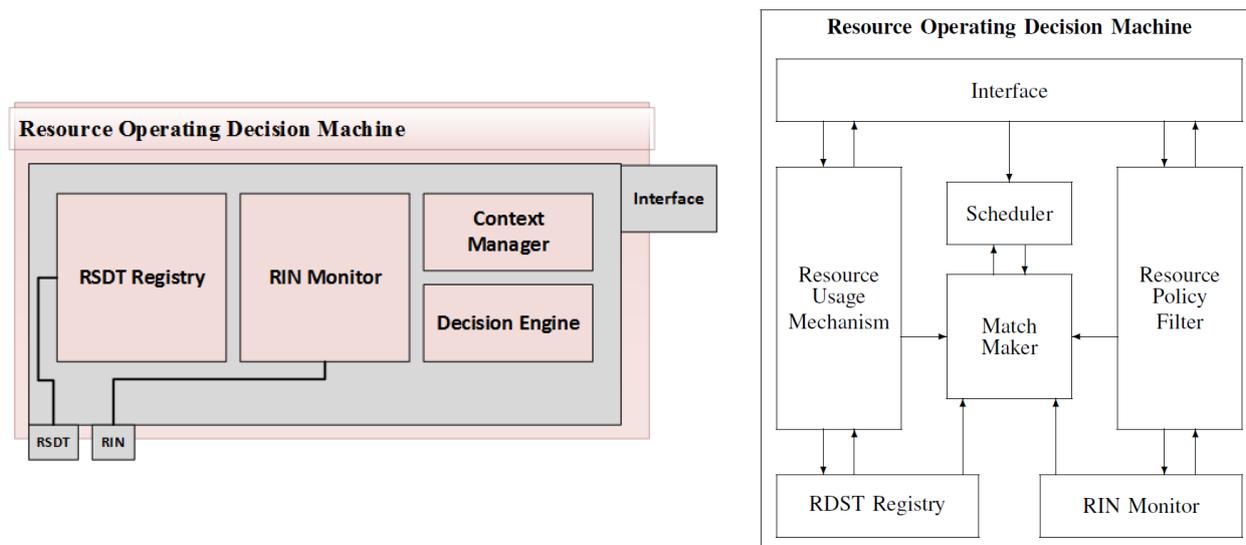
## 2.2. Resource Allocation Mechanism

In the RSDT definition, we have included characters, attributes and contexts. RODM will automatically decide the operations on them. All of the configurations and decisions made by the RODM are part of the contract between the resource provider and the cyber-physical system. RODM has the following four important responsibilities:

1. It has to read the information included in the characters section of the RSDT and decided how to use the resources from this template. It has to hold the RSDT description, as well as the information on resource providers.
2. It has to use the attributes information in the RIN in order to achieve a real-time effect in the system.
3. It has to make decisions about how to manage and allocate resources.
4. It has to provide an interface to the CPS that can manage resources.

From the above stated requirements, we have designed the RODM shown in Figure 1.

It is clear that the RODM has three connectors, respectively, for the RSDT, RIN and interface to the CPS. All of the RSDTs have to register with RODM when they want to join the cyber-physical system. All of the generated RINs are monitored at run-time. The resource requesters have limited rights to interact with this system through the interface.

**Figure 1.** Structure of the resource operating decision machine (RODM).

Besides acting as an interface between the resource and service, the decision engine is divided into four parts. Together, they make high-level decisions based on the real-time context. The four parts are the following:

(1) The resource usage mechanism is the module that analyses the information from an RSDT and decides how to use the resources. Because the RODM machine has to make decisions at a high level, the resource usage mechanism module needs a special Resource Usage Specification Language in order to load the resource information from the registered RSDT. Some researchers [12,13] have proposed a resource usage specification language for distributed systems or virtual machines. Projects like Globus [14] and Condor [15] have their own solutions for resource management and allocation. In Globus, a “Resource Specification Language” (RSL) has been designed to describe the request and communicate information regarding the jobs [16]. In Condor [15], the “classified ad” mechanism can define constraints and mechanisms within the resource descriptions. In our approach, within RODM, we are trying to modify the syntax and mechanism of RSL to specify the usage mechanism of resources in CPS.

(2) The resource policy filter is the module connected to the RIN monitor and that supports the exception handler for the RIN monitor. This module is a critical, important module in RODM to verify whether the support resources are correct for the tasks in the cyber-physical system. It is used to control the resource quality and then guarantee the security of resource management model. If a resource instance cannot pass this resource policy filter, the RODM machine will start the exception handler to handle the errors.

(3) The scheduler is the connector between the cyber-physical system and this resource management RODM machine. All resources managed by the RODM machine need to provide some specified task from the cyber-physical system through the interface. Because both the cyber-physical system and resources used by the system could be distributed, large-scale and heterogeneous, the tasks need to be arranged to correctly allocate suitable resources. This scheduler module is designed so that it can arrange a large amount of different tasks coming from the interface.

(4) The match maker is the core decision engine to choose the correct resources to support related tasks. Different tasks need different resources, and this module will analyse, calculate and decide the eventual resource allocations. It has connections to all other modules to support complicated decisions. Most of the decision problems here can be treated as optimization problems and can be solved by many existing algorithms or machine learning methods.

### 2.3. Analysis and Discussion of DRSM

A number of resource models and related systems have been proposed. Some examples are typical operating systems [4], Resource Description Framework (RDF) [5], Representational State Transfer (REST) [6], the resource space model (RSM) [7], entity-relationship model [17] and the Resource Description Template (RDT) [10]. In all models, the designers try to express all attributes by describing the resources in different views. In addition, most of them also provide mechanisms to manipulate the resources. However, cyber-physical systems involve a heterogeneous collection of resources and have different requirements about their specifications and management. The DRSM is designed for dealing with such complex resources types. A detailed comparison of DRSM with the other approaches is given below.

(1) In an operating system, the CPU, cache and memory resources are extremely fine-tuned to support higher level software. However, the operating system is a special middleware program that just needs to consider limited types of resources, like CPU and memory. Therefore, it does not support a generic resource description and related operational approach.

(2) RDF [5] is a framework to represent resources in the web using the Uniform Resource Identifier (URI). This resource model only describes the information resources and is therefore not suitable for describing physical resources or a more generic resource model.

(3) REST [6] is an architecture style for resource description, describing the sources of specific information on the web. Compared with RDF, it is extended to manipulate the resources by transferring the representational states of the applications. However, this model still aims at the web environment and is only suited for virtual resources.

(4) RSM [7] is a special resource description method that treats any resource as an entity located in a multi-dimensional space, where each dimension is one resource attribute. This resource model is at the logical layer to describe the resources and suitable for applications, like search engines. However, our resource model supports not only static description, but also dynamic manipulation.

(5) The resource-explicit service model (RESM) proposed in [17] can be treated as an example of the entity-relationship model. In their model, the physical devices are considered as resources to support services. The resources and service contexts are modelled as a bundle in a single ER diagram. The resource in this model is not a concept independent from services, but just an important property of services. The resource and service models are coupled too tightly.

(6) The Resource Description Template (RDT) separates the concerns between resources and services [10,11]. However, this resource model is still part of the resource-centric service in CPS, which means that the RDT is tightly coupled with the service model. In our resource model, the RDT is extended and separated from services.

Resource specifications in DRSM can adapt to different types of resources. In complex CPS, there are usually many thousands of resources and related dynamically changing data. The idea of central control, as we have designed in RODM, is obviously not enough to manage all resources. Therefore, we are currently working to make this model work in a distributed environment. The basic idea is to design an appropriate mechanism to build a multi-agent system. Then, any RODM in this whole system can make decisions from their own knowledge and requests.

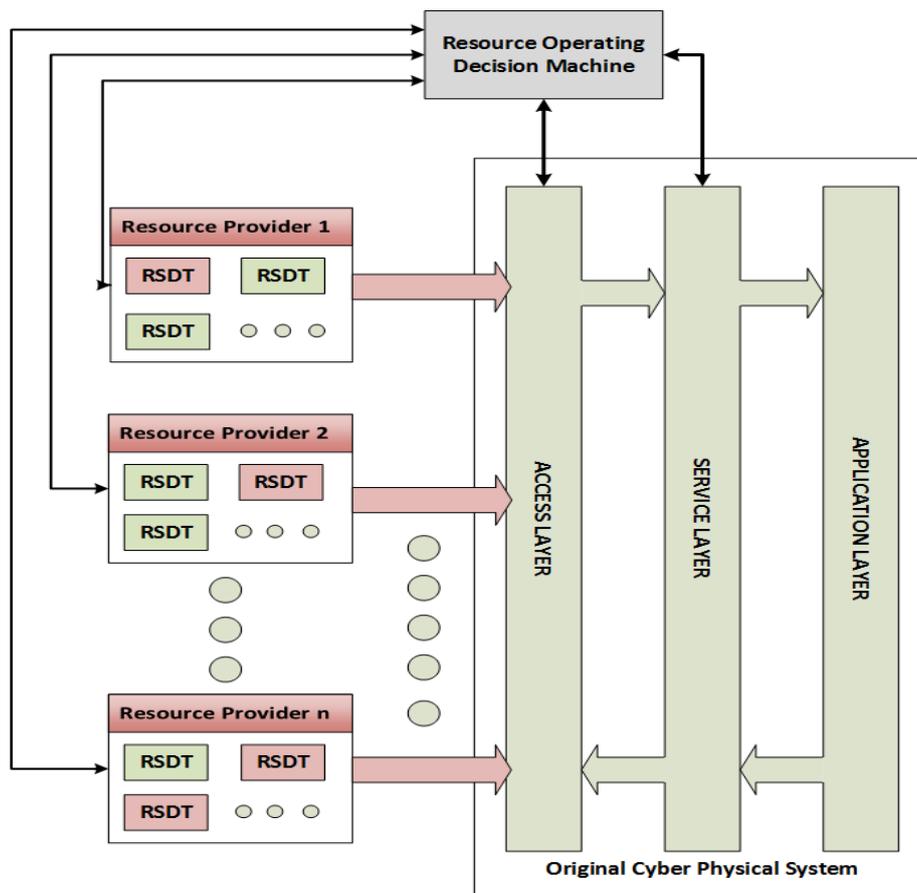
### 3. Resource Supply Model to Service-Oriented Architecture

In this section, we will briefly discuss how to use this model in a cyber-physical system with a service-oriented architecture (SOA). In the SOA, the system always needs to consume some resources to support the services. The services might need other services with other resources, and then, they usually have complex dependencies and associations in the system, because the resources might be produced from other services or the resource might rely on other resources. Therefore, the service-oriented view for CPS is much more complicated than the software requirements for traditional business applications [8].

The RSDM is designed to support resource specification, management and allocation in CPS. Therefore, the final implementation of the model is supposed to be a middleware to integrate with CPS. In most of software architectures, there are some technical solutions to support the middleware. However, the SOA is naturally very easy to interface with other middleware. In this paper, we choose a service-oriented architecture in a cyber-physical system to integrate with our resource model, because it provides better loose coupling, location transparency and protocol independence [18]. In [19], the three layers “access layer”, “service layer” and “application layer”, have been introduced in SOA towards achieving a solution to integrate SOA with CPS. In our approach, we also have these three layers, but we integrate DRSM with the layers to arrive at a new SOA for CPS. The goal is to separate the concerns between the resources and services. That is, the system does not need to consider resource dependencies on the resources and services. Instead, all resources provided for services are supported and managed by the resource models, while the contexts handling will be integrated in the middleware. Figure 2 shows our model, explicitly showing how the resource model collaborates with the main system.

For us, the service providers are the RPs, and the resources are located in the access layer. In Figure 2 there, are  $n$  RPs that provide their resources to a CPS through the access layer. Each RP holds several different RSDTs. Some of the RSDTs provide resources at run-time, and others may be just in sleep time. An RP has to register into the RODM, and the RODM has to monitor the resource providing situation. Therefore, both the RPs and the access layer in the CPS have connections to RODM. Finally, the services have the rights to check the resources and manage related contexts between the resources and services. Therefore, RODM also has a connection to the service layer in the CPS.

**Figure 2.** Applying the dynamic resource supply model (DRSM) into a cyber-physical system.



#### 4. Applying the Resource Supply Model to a Smart Grid Environment

The smart grid is an emerging new technology, which makes use of modern computation and communications technology to improve the efficiency and reliability of power transmission and power distribution in an electric grid. In general, the electric grid is basically made up by the electricity power transmission system (TS), the electricity distribution system (DS) and the central demand management system [20]. The origin of the term “smart grid” appears in the article “Reliability demands will drive automation investments” [21]. The smart grid transfers electricity from electric stations (suppliers) to personal electricity users and industrial consumers using modernized informatics and communication technology to save energy and increase sustainability and efficiency. In [22], the electric grid is stated as arguably the world’s largest engineered system, which is vital to significantly improve human life and is the motor of the economy and the major driver of progress.

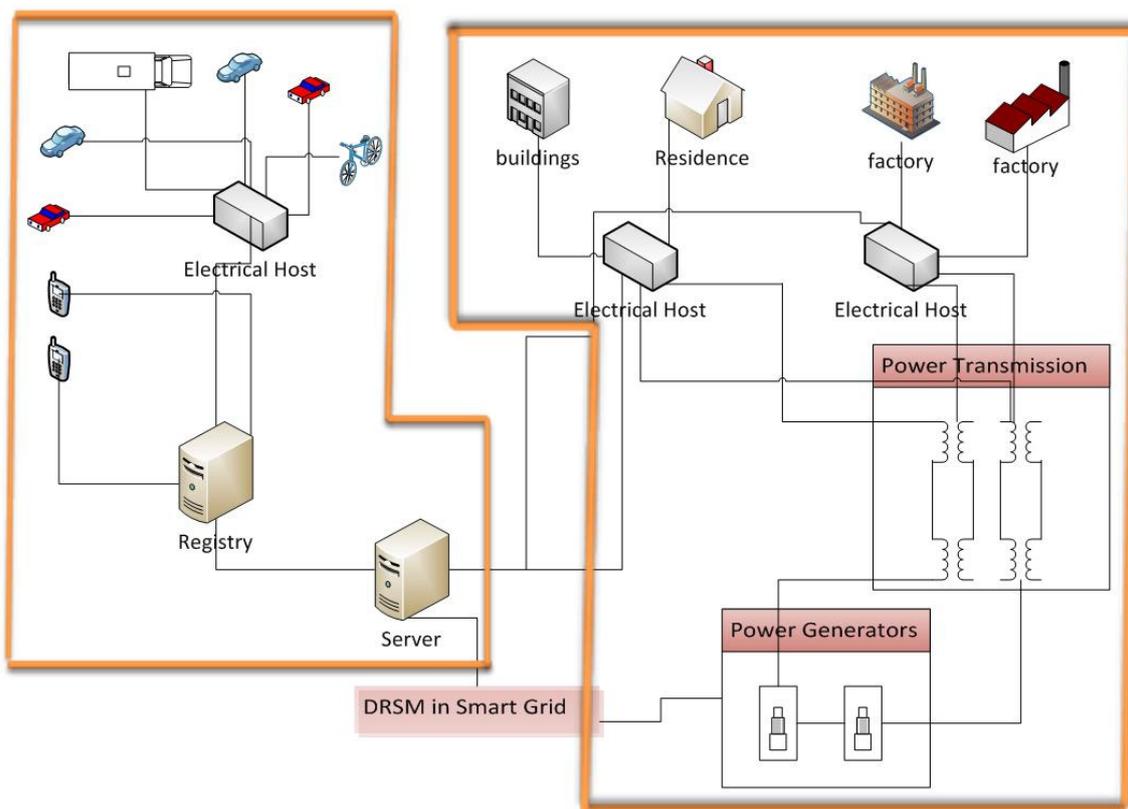
In this paper, we apply DRSM to the smart grid as a case study. Specifically, DRSM is applied to manage the resource, *i.e.*, electricity in this case, and provides an optimization-based decision making mechanism for the smart grid electricity transmission and distribution processes. Figure 3 illustrates the whole structure of the scenario. The system consists of three major components:

- (1) DRSM is shown at the left bottom of Figure 3. DRSM will make smart decisions to distribute electric power and charge the price of power through available data.

(2) A charging request of plug-in hybrid electric vehicles (CR-PHEV) through a web service is shown in the left block of Figure 3. This part deals with communication between the PHEV and DRSM through the Android system in order to request for the electric power pricing.

(3) Electricity consumption data collection and allocation (ECDCA) through a ZigBee network is shown in the right block of Figure 3. ECDCA deals with communication between normal electric devices and DRSM in the smart grid. Electricity consumed by devices and consumption data are sent to DRSM. The decisions made by DRSM are sent back to the electric devices.

**Figure 3.** Whole structure of the smart grid scenario.



#### 4.1. Design of Charging Request of PHEVs in the Smart Grid (SG) Environment

The wireless communication technologies are enabled for SG [23]. Multiple supporting communications technologies are local area networking, mobile broadband communications and the cellular network. SG highly relies on these robust and secure communication methods. Using a mobile platform can be a suitable choice to add mobility to SG. Warner *et al.* [24] have also investigated a web service based on SG models. They stress three requirements for integrating web services into a smart grid. These are the end-user feedback, automated decentralized control of distributed and demand response and control of the grid stability and islanding operation. Omar Asad *et al.* [25] use the sensor web services to manage the charging of plug-in hybrid electric vehicles (PHEVs). Their study shows that continuous and uncontrolled charging increase of the load during peak hours, while off-peak charging provides the most promising results. However, off-peak charging will not be always successful, as the public charging stations in shopping malls or at business centres are not always available. In [26], the authors have proposed a prediction-based PHEV charging for a smart grid,

employing a special real-time price setting system. The time period of charging is determined by communications with the smart grid in order to decrease consumer expenses.

Currently, the two protocols that can be used for web services are SOAP and RESTfull. Their relative performances have been investigated in [27]. RESTfull was firstly investigated by Roy Fielding [6]. The efficiency and power consumption of web services using RESTfull have been shown to be more efficient than SOAP. However SOAP is widely used in web services and is shown to have better compatibility with other applications than RESTfull [28,29]. Based on these investigations, we were convinced to use SOAP.

The integrated structure inside the orange square of Figure 3 is designed for our work. We consider a variety of electric vehicles that need to be charged in the selected electrical station. The electricity purchase decision is made with the assistance of SG-CPS after taking into account some occurrence of electricity shortfalls and the cost of grid electricity. The cloud represents the registry that withholds some information, which has not been decided yet. Whenever doable, the location of the charging station also plays a role in making optimal selections of a charging station.

In Figure 3, charging request of plug-in hybrid electric vehicles (CR-PHEV) through the web service are shown in the left orange block. Below is the detailed scenario for our design of CR-PHEVs. When the owner of the electric vehicle wants to charge his car, he can use an app on his Android smart phone to ask for a vehicle charging schedule. The main scenario is that the owner uses his smart phone to send a request to the registry, and then with the feedback from the registry, the client sends the request to the connection layer of the management system. In reality, the location of the car owner is also a factor for SG to make its decision. In our further research, the Google Map positioning system will be included for providing current location information to the management system. A resource list is temporally defined as XML files that contain information of electricity charging stations and other necessary information. The resources can be registered at RODM; thus, the charging places are at the access layer, and the charging requests and scheduling are at the service layer of our whole system. Then, the electricity charging stations may hold their own RSDT and some run-time data of RIN. Table 4 and Table 5 specify a simple example of an electricity charging stations.

Figure 4 illustrates the design of the CR-PHEV through the web service. There is a customer manager module that contains relevant personal information of users, such as the user ID, account number and account balance. The user interface module may obtain a variety of platforms supporting web services and other portable communications. After checking the user availability, the client terminal asks the current states for services in case some resources are unavailable. The query is delivered to the service manager from the service agent. The current service state and some resource description samples of some services are listed inside. Then, the query is sent to the resource agent, which is the centre piece of the structure interacting with other four modules. It not only determines the resources corresponding to the selected service, but also returns the service state at that time together with the function of the resource manager module and knowledge library module. The knowledge library is designed for holding algorithms that control the relationship between service and physical resources. There may be one accessible service and resource currently. However, more than one resource can be chosen for a service implementation, and there can also be many physical devices with same properties and functionalities. The module “knowledge lib” is a necessary part of our design, because it helps the resource agent to make decisions. The “resource manager” shown at the left corner

of the bottom in Figure 4 is designed for holding the resource RDT and states. A database structure is considered temporarily for the resource manager. A special module called “user privilege check” is constructed for the user credit check or special services requiring different privileges.

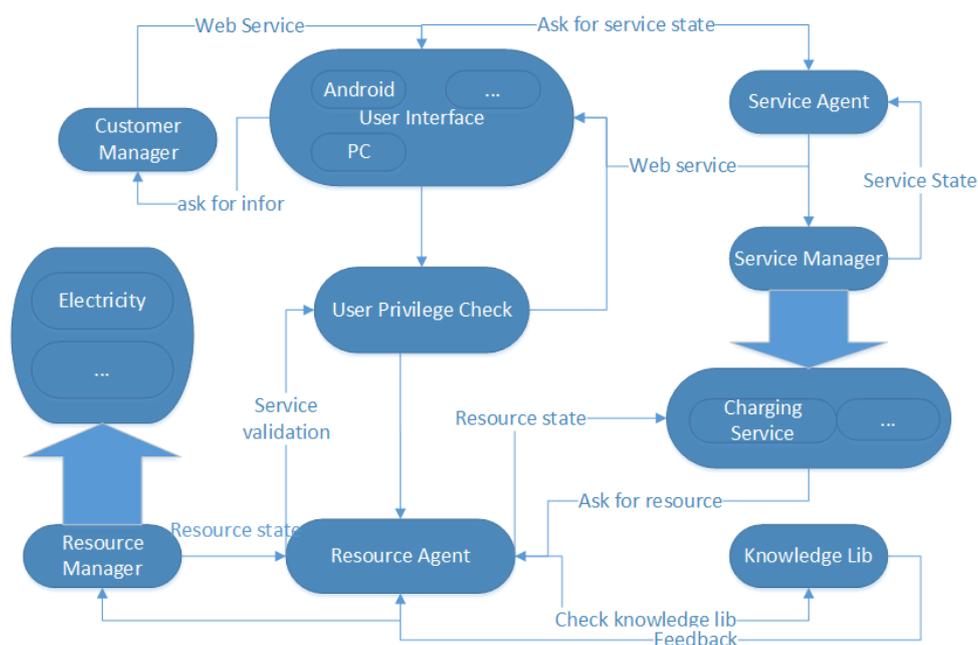
**Table 4.** RSDT published by an electricity charging station.

|   |   |
|---|---|
| <b>Template Holder</b>                      | XYZ Electricity Charging Station  |
| <b>Template Unify ID</b>                    | fe80:8814:c1f1:90e9:1319:8a2e:0370:7344   |
| <b>Resource Name</b>                        | Electricity Charging Positions (AC)   |
| <b>Characters</b>                           | discrete resources<br>exclusive   |
| <b>Attributes</b>                           | Non-Inherited: Voltage = 230 VAC<br>Non-Inherited: Current = 16 A<br>Non-Inherited: Power = 3.3 kW<br>Inherited: Places |
| <b>Cost</b>                                 | 20 U.S. cents/kWh   |
| <b>Availability</b>                         | Location: ZZZ (longitude, latitude)<br>Time: 6 am–12 pm   |
| <b>Legal Rules for Supply</b>               | URI to www.XYZWindPowerPlant.com  |
| <b>Other Resource in the Context of Use</b> | electricity supply  |

**Table 5.** An RIN sample generated from the RSDT in Table 4.

|                      |                                     |
|----------------------|-------------------------------------|
| <b>Resource Name</b> | Electricity Charging Positions (AC) |
| <b>Resource ID</b>   | 2014.8.28.1946                      |
| <b>Attributes</b>    | Places: 19                          |

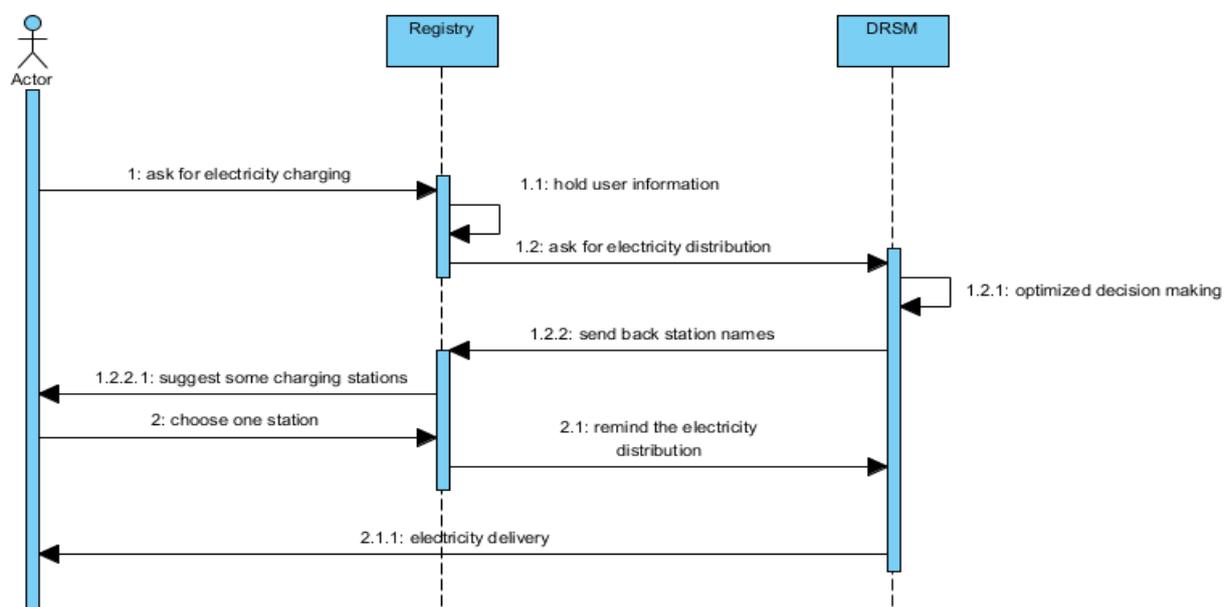
**Figure 4.** Design of the charging request of plug-in hybrid electric vehicles (CR-PHEV).



In Figure 5, the SR (service requester) is the “actor”, SP (service provider) is the “registry” and RP (resource provider) is DRSM. The communication process is specific to the service type for electricity

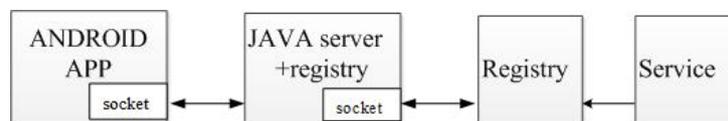
distribution. At first, the car owner asks the registry for the car charging schedule. The registry will hold some charging station information and then asks the connection layer of the management system. Then management system will give an optimized decision on the charging schedule and send back the decisions to its client. The register will list some electricity stations for the car owner to choose. The customer (actor) can choose to go directly to the charging station. Then, the electricity will be transmitted and distributed to the station. If necessary, the car owner is able to choose one station to go to, and then, the registry will inform it of the electricity distribution of the management system. This will benefit the efficiency of TS for less power consumption.

**Figure 5.** The sequence diagram for web service in the smart grid (SG).

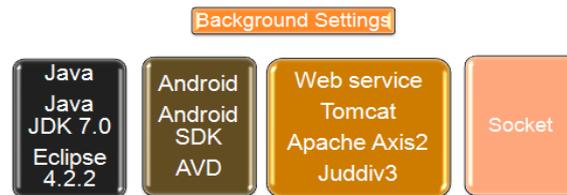


The whole process, shown in Figure 6, is designed in this way. The Java program, as a server, first asks the UDDI registry for the web service registration address. Then, it passes the address to the Android client app. Then, the Android app will access the web service by using the service address.

**Figure 6.** The whole process for the web service in SG.



The class diagram of the web service is given in [30]. Following the detailed design, the implementation was completed. Figure 7 shows the basic software components in the implementation. The detailed implementation and test results appear in [31].

**Figure 7.** The diagram for background settings.

#### 4.2. Design of ECDC in the Smart Grid (SG) Environment

The smart grid is a modern electric power control system. It uses automated control and modern communication to improve the efficiency of power distribution and makes the electric grid safer. In the smart grid, communication between users and the central power station is very important and fundamental. The monitoring and communication in a traditional electric power distribution system have been achieved in some areas; however, they are implemented through a wired network, which requires expensive communication cables and regular maintenance. As a result, this technology has not been widely implemented, due to the high cost [32]. Hence, it is important to develop a cost-effective electric power distribution system based on wireless monitoring and, thereby, allow the system to improve the efficiency by optimizing the power distribution.

The main motivation for using the ZigBee network in our work comes from the following analysis. ZigBee is a wireless network standard, which is designed for sensors and control devices. The requirements of sensors and controls are low latency and low energy consumption. Compared with the protocol enter times for Wi-Fi (3 s) and Bluetooth (3–10 s), the ZigBee protocol node enter time is only 15 ms. ZigBee node can work for 6–24 months with two AA batteries. However, Bluetooth can only work for several weeks, and Wi-Fi can only work for several hours. Because the ZigBee protocol is designed for small data transmission, it does not need a high bandwidth. The physical layer of ZigBee uses a direct sequence, so that the analogous circuitry is simple and tolerant. The implementation is cheap. The media access control (MAC) layer does not need large amounts of ROM or RAM. The main requirement of MAC is that is designed to handle a large numbers of devices. The network layer is designed to allow the network to grow spatially, and the layer can handle very large amounts (maximum 65,535) of nodes in a low latency [33]. Usually, there are three kinds of equipment. These are the coordinator, router and end device. In this project, one coordinator and several end devices form a network representing electricity consumption place.

The smart grid usually contains three major subparts, and in each subpart, power generation, power delivery and power utilization are done. We use the ZigBee network to collect electric power data and receive the feedback command to change the actual power provision of an electric device. The ZigBee device includes a coordinator and end devices. Several end devices and one coordinator form a ZigBee network to simulate electric power consumption in a particular area, such as in one house or in one room. Several ZigBee networks can be connected together to simulate electric power consumption in a particular region.

In Figure 8, different end devices simulate different kinds of electric power consumption of electric devices, and the data will be transmitted to the coordinator through the ZigBee network. After the coordinator receives the data, it will upload the data to the computer through USB, which will be

published to the DRSM. Then, the DRSM will calculate the data and give feedback to the coordinator. The feedback message will be finally sent to end devices through the ZigBee network by the coordinator.

In Figure 8, the electric devices are connected to the ZigBee devices using some particular sensors. These sensors are used to measure the actual electric power of electric devices, and these ZigBee devices that are connected to the electric devices are called end devices. A coordinator creates a ZigBee network. The end devices can join the ZigBee network and, thus, send data to the coordinator and receive data from the coordinator through the ZigBee network. The coordinator is connected to a computer using USB, and the computer is connected to the DRSM through the web service. Accordingly, the data flow of the designed ZigBee network is illustrated in Figure 9.

Figure 8. Infrastructure of the ZigBee network.

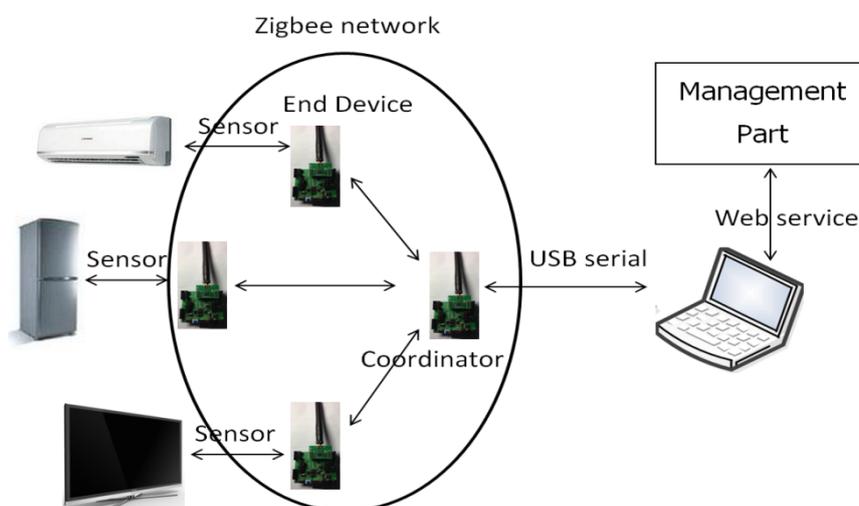
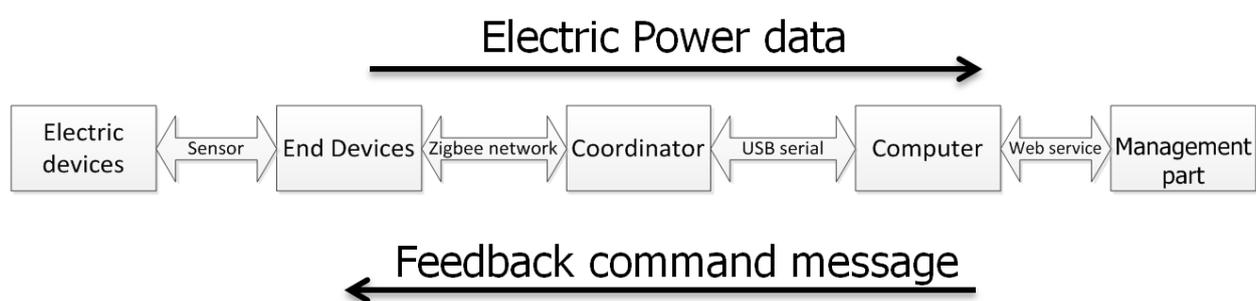


Figure 9. Data flow of the ZigBee network.



The functionalities of end devices, the coordinator and the electricity control unit used in the design are as follows.

(1) End devices.

The main job for end devices is to simulate electric power consumption data, which is defined as actual power. The actual power is defined as “factor × rated power”, where factor represents the percentage of the rated power and the rated power is set according to different simulations of electric devices. The factor will be adjusted according to the feedback command message from the electric power distribution system. Figure 10 illustrates the activity diagram for end devices.

(2) The coordinator.

The main job for the coordinator is to collect actual power data from end devices and send the data to the electricity power distribution system through USB. Furthermore, the coordinator needs to get a feedback message from the electricity power distribution system and send the message to the end devices. Figure 11 illustrates the activity diagram for the coordinator.

(3) The electricity control unit.

This unit collects actual power data from the coordinator through USB, sends the data to the DRSM and forwards feedback information to the coordinator. Figure 12 illustrates the activity diagram for the electricity control unit.

Based upon the detailed design described above, we have completed an implementation [34].

**Figure 10.** The activity diagram for end devices.

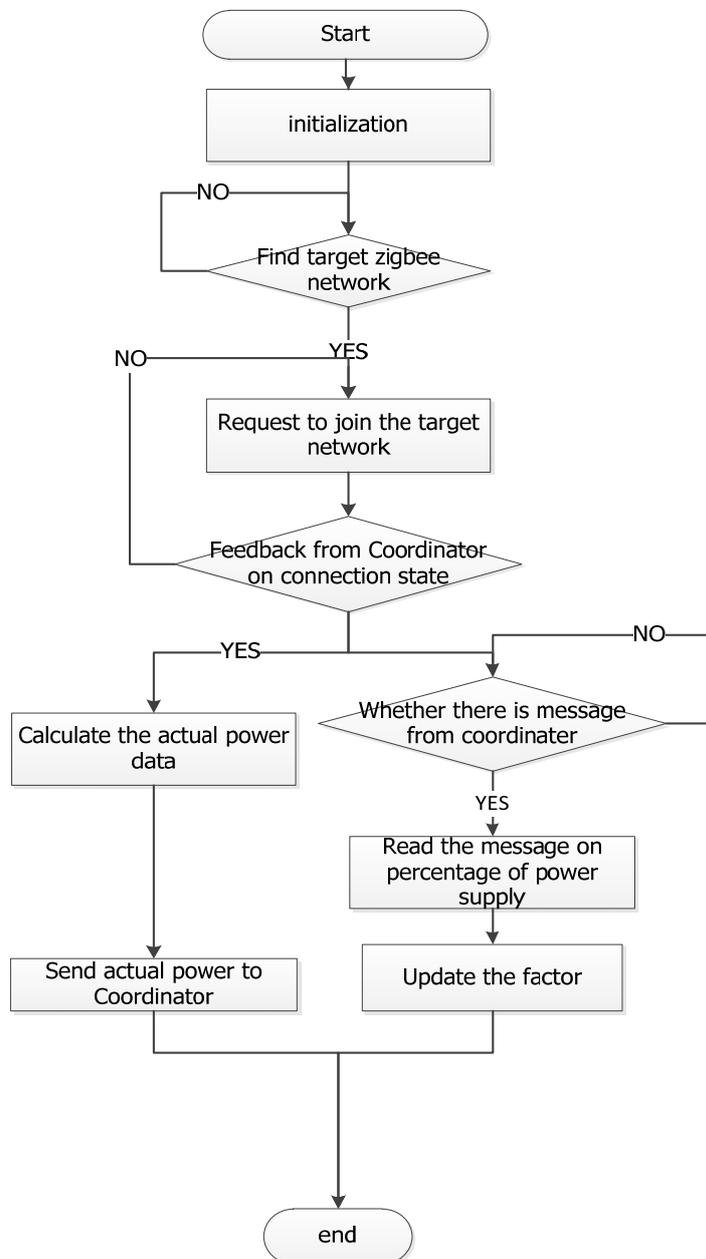


Figure 11. The activity diagram for the coordinator.

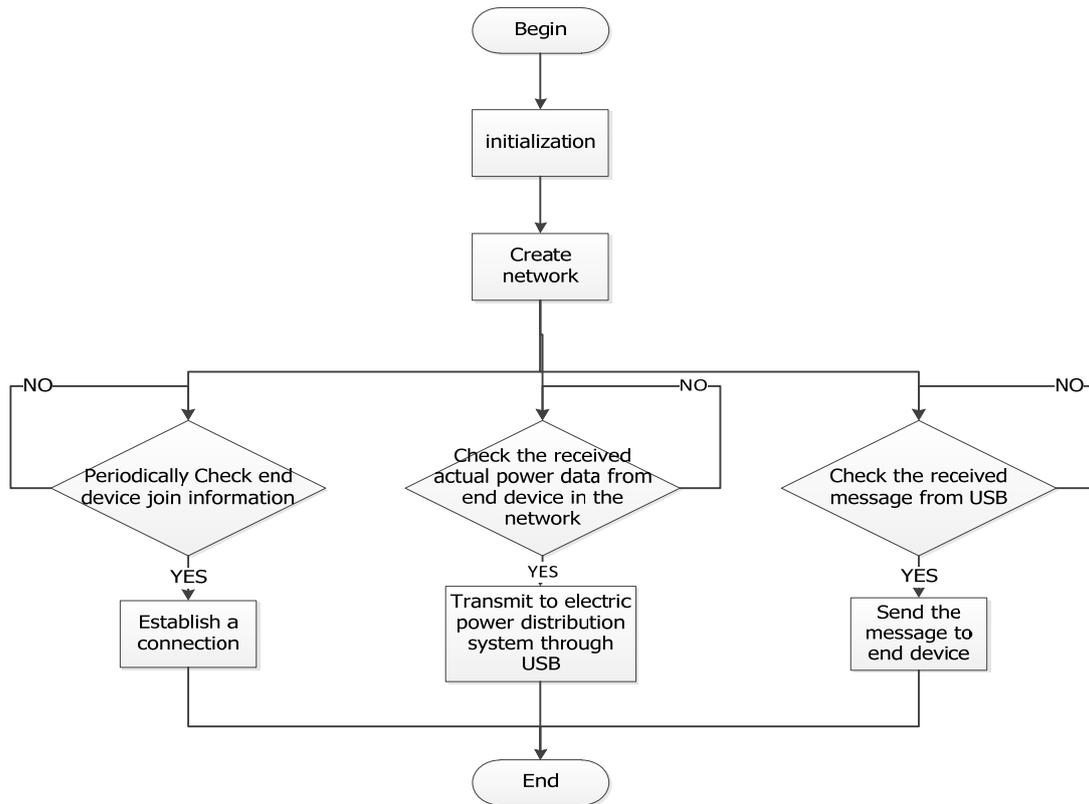
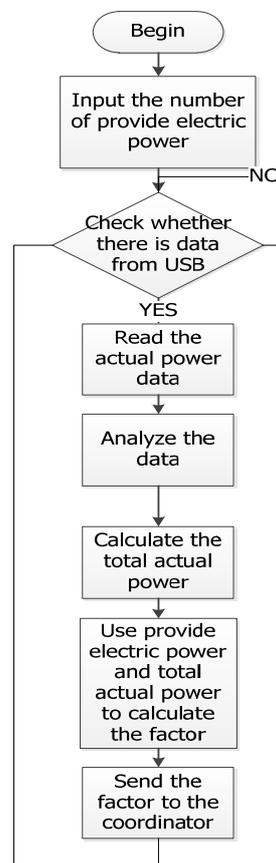


Figure 12. The activity diagram for the electricity control unit.



## 5. Conclusions

In this paper, we have proposed a dynamic resource supply model (DRSM) containing a Resource Supply Description Template, Resource Instance Notation and a resource operating decision machine (RODM). The whole model is designed, explained and integrated into a three-layer service-oriented architecture of a cyber-physical system. The advantages of this model include extending the resource concept to a more generic model, introducing both a static and dynamic resource description and separating context management by the RODM. In order to validate the advantages of the separation of the concerns achieved in our design and the significant role of the resource model, we applied the DRSM in the design and implementation of a smart grid, which is a typical example of a complex CPS. We have described the mechanisms and detailed design of communication between plug-in hybrid electric vehicles (PHEV) and DRSM through the Android system. We explained the advantages of using the ZigBee network for collecting electricity consumption data and re-allocation of power through the network.

We are working to improve the algorithm of resource collection and re-allocation, so that the real-time decision can be made more intelligent, electricity power can be distributed more evenly and the price of power can be charged more dynamically according to the different time durations and available power. In our current work, we have used several end devices and one coordinator to form a ZigBee network, which is sufficient for power allocation to a single house or one room in a house. In the near future, we are planning to put together several ZigBee networks to form a hyper ZigBee network, so as to simulate electric power consumption in a region.

## Acknowledgments

This research is supported by Grants from the National Natural Science Foundation of China (Project Number 61103029), the XJTLU PGR Scholarship PGRS-12-03-05 and XJTLU Research Development Funding, RDF-13-02-06.

## Author Contributions

Kaiyu Wan, Yuji Dong, Qian Chang and Tengfei Qian conceived and designed the experiments; Yuji, Qian and Tengfei performed the experiments and analyzed the data; Kaiyu, Yuji, Qian and Tengfei wrote the paper.

## Conflicts of Interest

The authors declare no conflict of interest.

## References

1. National institute of standards and technology. Foundations for Innovation in Cyber-Physical System. Available online: <http://www.nist.gov/el/upload/CPS-WorkshopReport-1-30-13-Final.pdf> (accessed on 16 September 2014).

2. Industry-Academy Collaboration in Cyber Physical System (CPS) Research. Available online: <http://www.cra.org/ccc/files/docs/CPSWorkshopCFP/CPS-White%20Paper-May-19-2009-GMU-v1.pdf> (accessed on 16 September 2014).
3. Sunder, S.S. Foundations for Innovation in Cyber-Physical Systems. In proceedings of the NIST CPS Workshop, Chicago, IL, USA, 13 March 2012.
4. Stallings, W. *Operating Systems: Internals and Design Principles*, 6/e; Pearson Education: Bangalore, India, 2009.
5. Klyne, G.; Carroll, J.J.; McBride, B. Resource description framework (RDF): Concepts and abstract syntax. Available online: <http://www.w3.org/TR/2014/REC-rdf11-concepts-20140225/> (accessed on 16 September 2014).
6. Fielding, R.T. *Architectural Styles and the Design of Network-Based Software Architectures*. Ph.D. Thesis, University of California, Irvine, CA, USA, 2000.
7. Zhuge, H.; Xing, Y.; Shi, P. Resource space model, OWL and database: Mapping and integration. *ACM TOIT* **2008**, *8*, 20.
8. Georgakopoulos, D.; Papazoglou, M.P. *Service-oriented computing*; The MIT Press: Cambridge, MA, USA, 2008.
9. Sztipanovits, J.; Karsai, G.; Antsaklis, P.; Baras, J. Toward a science of cyber-physical system Integration. *Proc. IEEE* **2012**, *100*, 29–44.
10. Wan, K.; Alagar, V. Modeling Resource-centric Services in Cyber Physical Systems. In Proceedings of the International Multi Conference of Engineers and Computer Scientists, Hong Kong, China, 13–15 March, 2013.
11. Wan, K.; Alagar, V. A resource-centric architecture for service-oriented cyber physical system. In *Grid and Pervasive Computing*; Gil, S.-M., Shi, W., Eds.; Springer: Berlin, Germany, 2013; pp. 686–693.
12. Uppuluri, P.; Gupta, A. Resource usage policy specification for managing application resources. In Proceedings of the Fifth International Conference on Information Technology: New Generations, ITNG 2008, Las Vegas, NV, USA, 7–9 April 2008.
13. Damianou, N.; Dulay, N.; Lupu, E.; Sloman, M. The ponder policy specification language. In *Policies for Distributed Systems and Networks*; Springer: Berlin, Germany, 2001; pp. 18–38.
14. Foster, I.; Kesselman, C. Globus: A metacomputing infrastructure toolkit. *Int. J. High Perform. Comput. Appl.* **1997**, *11*, 115–128.
15. Litzkow, M.L.; Livny, M.; Mutka, M.W. Conder—A hunter of idel workstations. In Proceedings of the 8th International Conference on Distributed Computing Systems, San Jose, CA, USA, 13–17 June 1988.
16. Czajkowski, K.; Foster, I.; Karonis, N.; Kesselman, C.; Martin, S.; Smith, W.; Tuecke, S. *A Resource Management Architecture for Metacomputing Systems*; Springer: Berlin, Germany, 1998; pp. 62–82.
17. Huang, J.; Bastani, F.; Yen, I.-L.; Jeng, J.-J. Toward a Smart Cyber-Physical Space: A Context-Sensitive Resource-Explicit Service Model. In Proceedings of the 33rd Annual IEEE International Computer Software and Applications Conference, Seattle, WA, USA, 20–24 July 2009; IEEE: New York, NY, USA, 2009; pp. 122–127.

18. Endrei, M.; Ang, J.; Arsanjani, A.; Chua, S.; Comte, P.; Krogdahl, P.; Luo, M.; Newling, T. *Patterns: Service-Oriented Architecture and Web Services*; IBM Corporation, International Technical Support Organization: New York, NY, United States, 2004; p. 25.
19. Hoang, D.D.; Paik, H.-Y.; Kim, C.-K. Service-oriented middleware architectures for cyber-physical systems. *Int. J. Comput. Sci. Netw. Secur.* **2012**, *12*, 79–87.
20. Smart Grid. Available online: [http://en.wikipedia.org/wiki/Smart\\_grid](http://en.wikipedia.org/wiki/Smart_grid) (accessed on 29 November 2013).
21. The History of Electrification: The Birth of our Power Grid. Available online: <http://firstlogicaautomation.com/blog/history-electrification-birth-power-grid-plc-scada-training-centre/> (accessed on 29 November 2013).
22. Mo, Y.; Kim, T.H.-H.; Brancik, K.; Dickinson, D.; Lee, H.; Perrig, A.; Sinopoli, B. Cyber-physical security of a smart grid infrastructure. *Proc. IEEE* **2011**, *100*, 195–209.
23. Suo, H.; Wan, J.; Li, D.; Zou, C. Energy Management Framework Designed for Autonomous Electric Vehicle with Sensor Networks Navigation. In Proceedings of the IEEE 12th International Conference on Computer and Information Technology, Chengdu, China, 27–29 October 2012.
24. Warmer, C.; Kok, K.; Karnouskos, S.; Weidlich, A.; Nestle, D.; Selzam, P.; Ringelstein, J.; Dimeas, A.; Drenkard, S. Web Services for Integration of Smart Houses in the Smart Grid. Available online: <http://smarthouse-smartgrid.eu/fileadmin/templateSHSG/docs/publications/GridInterop.pdf> (accessed on 16 September 2014).
25. Asad, O.; Erol-Kantarci, M.; Mouftah, H.T. Management of PHEV Charging from the Smart Grid Using Sensor Web Services. In Proceedings of 2011 24th Canadian Conference on Electrical and Computer Engineering (CCECE), Niagara Falls, ON, Canada, 8–11 May 2011.
26. Erol-Kantarci, M.; Mouftah, H.T. Prediction-Based Charging of PHEVS from the Smart Grid with Dynamic Pricing. In Proceedings of the First Workshop on Smart Grid Networking Infrastructure in LCN 2010, Denver, CO, USA, 10–14 October 2010.
27. Dogan, Y.; Adam, D. Efficient application integration in ip-based sensor networks. In Proceedings of the First ACM Workshop on Embedded Sensing System for Energy-Efficiency in Buildings, Berkeley, CA, USA, 04–06 November 2009.
28. Kamlaris, A.; Trifa, V.; Pitsillides, A. HomeWeb: An Application Framework for Web-based Smart Homes. In Proceedings of the 18th International Conference on Telecommunications (ICT 2011), Ayia Napa, Cyprus, 8–11 May 2011.
29. Weerawarana, S.; Curbera, F.; Leymann, F.; Storey, T.; Ferguson, D.F. *Web services platform architecture: SOAP, WSDL, WS-Policy, WSAddressing, WS-BPEL, WS-Reliable Messaging, and more*, 1st ed.; Prentice Hall: Englewood Cliff, NJ, USA, 2005.
30. Chang, Q.; Wan, K.; Dong, Y. Design and Implementation of Resource-Centric Web Services in Smart Grid Cyber-Physical Systems. In Proceedings of IMECS 2014, Hong Kong, China, 12–14 March 2014.
31. Chang, Q. Design and implementation of resource-centric web services in cyber Physical Systems. Presented at 2014 Final Year Project, Xi'an Jiaotong-Liverpool University, Suzhou, China, 23 May 2014.

32. Bennet, B.; Boddy, M.; Doyle, F.; Jamshidi, M.; Ogunnaike, I. Assessment study on sensors and automation in the industries of the future. Available online: [http://www.adventiumlabs.com/sites/adventiumlabs.com/files/doyle\\_03\\_2\\_0.pdf](http://www.adventiumlabs.com/sites/adventiumlabs.com/files/doyle_03_2_0.pdf) (accessed on 16 September 2014).
33. Patrick, K. ZigBee Technology: Wireless Control that Simply Works. Available online: [http://www.zigbee.org/imwp/idms/popups/pop\\_download.asp?contentID=5162](http://www.zigbee.org/imwp/idms/popups/pop_download.asp?contentID=5162) (accessed on 16 September 2014).
34. Qian, T. Smart Grid Simulation Based on Zigbee Network. Presented at 2014 Final Year Project, Xi'an Jiaotong-Liverpool University, Suzhou, China, 23 May 2014.

© 2014 by the authors; licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution license (<http://creativecommons.org/licenses/by/3.0/>).