Int. J. Environ. Res. Public Health **2013**, *10*, 6106-6130; doi:10.3390/ijerph10116106

OPEN ACCESS

International Journal of
**Environmental Research and
Public Health**
ISSN 1660-4601
www.mdpi.com/journal/ijerph

Article

IAServ: An Intelligent Home Care Web Services Platform in a Cloud for Aging-in-Place

Chuan-Jun Su * and Chang-Yu Chiang

Department of Industrial Engineering & Management, Yuan Ze University, No. 135, Yuan-Tung Rd., Chung-Li City, Taoyuan 320, Taiwan; E-Mail: s978905@gmail.com

* Author to whom correspondence should be addressed; E-Mail: iecjsu@saturn.yzu.edu.tw; Tel.: +886-3-463-8800 (ext. 2529); Fax: +886-3-463-8907.

Received: 3 September 2013; in revised form: 5 November 2013 / Accepted: 5 November 2013 / Published: 12 November 2013

Abstract: As the elderly population has been rapidly expanding and the core tax-paying population has been shrinking, the need for adequate elderly health and housing services continues to grow while the resources to provide such services are becoming increasingly scarce. Thus, increasing the efficiency of the delivery of healthcare services through the use of modern technology is a pressing issue. The seamless integration of such enabling technologies as ontology, intelligent agents, web services, and cloud computing is transforming healthcare from hospital-based treatments to home-based self-care and preventive care. A ubiquitous healthcare platform based on this technological integration, which synergizes service providers with patients' needs to be developed to provide personalized healthcare services at the right time, in the right place, and the right manner. This paper presents the development and overall architecture of IAServ (the Intelligent Aging-in-place Home care Web Services Platform) to provide personalized healthcare service ubiquitously in a cloud computing setting to support the most desirable and cost-efficient method of care for the aged-aging in place. The IAServ is expected to offer intelligent, pervasive, accurate and contextually-aware personal care services. Architecturally the implemented IAServ leverages web services and cloud computing to provide economic, scalable, and robust healthcare services over the Internet.

Keywords: ubiquitous computing; healthcare; aging-in-place; web service; cloud computing; ontology

1. Introduction

Rapid advances in medicine and technology are leading to marked increases in human longevity. Combined with declining birth rates, societies around the world are now facing challenges associated with rapidly aging populations. A recent report from Taiwan's Department of Health shows that nearly 90% of senior citizens in Taiwan have at least one chronic disease, while over half suffer from three or more chronic diseases. Many of these people are incapable of living independently, and their requirements for care will only continue to increase. These issues are raising broader concerns about the overall care and quality of life of the elderly.

Home care is conceived of as the integration of medical, social and familial resources to enable the integral care of elders in their own residences [1]. It emphasizes the concept of "aging-in-place" in which a range of products, services and conveniences are used to enable patients to remain in their own homes, even as their circumstances change [2]. The practice of aging-in-place would have the significant advantages of supporting elders' feelings of dignity, quality of life and independence. Today, however, the current connections between elderly health and housing are tenuous at best. As a result, while aging in place is the most desirable and cost-efficient method of aging, it is difficult to implement, even under the most ideal conditions [3].

The goal of healthcare in home is to improve health outcomes in the population and to improve service personalization, while reducing inequalities in both health outcomes and responsiveness. The healthcare needs of the population should be met with the best possible quantity and quality of services produced at minimum costs. Information technology (IT) can be used to enhance the delivery of healthcare services and to improve quality of life. In this research, we aim to seamlessly integrate such enabling technologies as ontology, intelligent agents, web services, and cloud computing for establishing the Intelligent Aging-in-place Home care Web Services (IAServ) platform to provide ubiquitous, personalized healthcare services.

In the context of healthcare, virtually no cases have identical presentations, and every patient has a unique history and disease manifestation. A robust methodology is therefore needed for the provision of personalized healthcare services, which takes into account the health status and various contexts of each patient [4,5]. To provide personalized health care services, ontology and rules of inference are utilized in IAServ to convert low-level context to high-level context. Ontological design used in IAServ furnishes an accurate formalization of healthcare goals for responding to the needs of health care providers and patients by directing and addressing their current aims. Personalization in IAServ is achieved by compiling individual versions of a personal profile, which derives the action plan from personal agents in a multi-agent environment. The dynamic aspects of health care environments, such as the rapid growth in domain knowledge and constant change in patient conditions, are addressed using the Maintenance Agent.

IAServe is platform- and language-independent, and was built on the Representational State Transfer (REST) architecture [6,7], which is a simple set of principles to connect applications in a style native to the Web. REST web services in the IAServ provide an abstraction for publishing information and giving remote access to the data repository. It regards the information or data as resources, each of which is referenced with a simple Uniform Resource Identifier (URI). Using REST technology, IAServ is able to hide the complexity of its Service Oriented Architecture (SOA) [8,9] and to interoperate other applications via a simple URI.

The development of IAServ aims to improve the quality of aging-in-place, increase the proactivity of healthcare services, and reduce the overall cost of providing such services. An ontology that allows for the formalization of healthcare goals is used to create a personalized care plan for each patient. The Web Services furnish various applications relevant to the provision of care activities. Mobile Agents offer contextually-aware, timely and accurate information for the provision of quality care, with care and data recorded automatically.

Section 2 of this paper reviews related work to discuss the development of Ambient Intelligence (AmI) and personalized services. The section also presents a discussion of IAServ which combines Web services, cloud computing and ontology development. Section 3 presents the system design and architecture of the IAServ. Section 4 covers implementation details and demonstrates usage scenarios. Finally, conclusions and suggestions for future work are made in Section 5.

2. Related Work

2.1. Ambient Intelligence (AmI)

The early developments in AmI took place at Philips, a Dutch electronics conglomerate. The board of directors commissioned a series of internal workshops to investigate different scenarios that would transform the high-volume consumer electronics industry from "fragmented with features" into a post-modern world where user-friendly devices would support ubiquitous information, communication and entertainment [10].

AmI refers to a vision of the future in which people are empowered by an electronic environment that is aware of their presence, and is sensitive and responsive to their needs. It aims at improving quality of life by creating the desired atmosphere and functionality via intelligent, personalized interconnected systems and services [11]. AmI implies a seamless environment of computing, advanced networking technology and specific interfaces. It is aware of the special characteristics of human presence and personality, addresses human needs and is capable of responding intelligently to spoken or gestured commands, and can even engage in intelligent dialogue with the user [12].

A typical context of applying ambient intelligence is smart home environment [13]. Wireless communication technologies can be used to connect home appliances with a specified control unit, which acts as a middle-ware mechanism to enable the remote control and monitoring of appliances through mobile devices. AmI is built on the concept of contextual-awareness, which raises the possibility of a new generation of interactive applications and systems. However, the development and deployment of interactive systems are ultimately determined by context [14].

Schilit and Theimer [15] defined contextual-awareness in the domain of ubiquitous computing as being "the ability of a mobile user's applications to discover and react to changes in the environment they are situated in." Perhaps due to technical limitations, the actual contextual information that Schilit and Theimer used in their ActiveMap Service was limited to location only. They suggested that future contextually-aware applications should include much more than geographic information, a suggestion that numerous researchers have attempted in building upon Barkhuus's research [16]. The Context

Broker Architecture (CoBrA) is an agent-based architecture supporting context-aware systems in smart spaces (e.g., intelligent meeting rooms, smart homes, smart vehicles, *etc.*) [17]. Central to this architecture is an intelligent agent called the context broker that maintains a shared contextual model on the behalf of a community of agents, services, and devices within the space and protects user privacy by enforcing user-defined policy rules.

2.2. AmI Applied in Care Services

Context-aware computing systems and methods are described, along with particular embodiments of location aware systems and methods. In the described embodiments, hierarchical tree structures are used to ascertain a device context or location [18]. Contextual awareness is not a new concept, but technologies (e.g., wireless technologies, mobile tools, wearable instruments, intelligent artifacts, handheld devices, *etc.*) are only now becoming available to support the development of contextually aware applications. Such technologies could help health care professionals manage tasks while increasing the quality of patient care [19].

Su and Chiang proposed an "Ambient Intelligent Community Care Platform" (AICCP) by applying radio-frequency identification (RFID) and Mobile Agent technologies to enable care givers and communities to offer pervasive and contextually-aware care services [20]. RFID allows care-givers to easily locate the patients, while Mobile Agents furnish contextually-aware, timely and accurate information for the provision of quality care, with care and treatment data recorded automatically. Coronato *et al.* presented a semantic location service to locate active mobile objects such as mobile devices and RFID-tagged objects in smart and intelligent environments [21]. The location service is constructed based on the ontologies and rules, which specify a uniform, well-definedand unambiguous model for the location information, but not by use of particular positioning system. Fraile *et al.* presented a hybrid Multi-Agent architecture, named HoCa, for the control and supervision of dependent environments [22]. The HoCa architecture provided the basic idea of incorporating an alert management system based on Short Message Service (SMS) and Multimedia Messaging Service (MMS) technologies and context control system based on Java Card and RFID technologies. Fenza *et al.* developed an integrated environment to provide personalized healthcare services based on agents, sensor networks, and ontologies. The context recognition, service matchmaking and brokerage activities are achieved through the use of agent technology [23,24]. The IAServ proposed in this paper integrates several novel concepts, (e.g., cloud computing, web services, *etc.*), to further increase the robustness of care service provision.

2.3. Profiles and Personalized Services

The user profile is the data instance of a user model that is applied to adaptive interactive systems. It also acts as the key component used to provide personalized services or information in the personalization system [25,26]. Eslami *et al.* proposed an effective service tailoring processes and an architecture to personalize homecare services according to the needs of the individual patient. In the proposed approach, the tailoring process is divided into six steps to configure the personalized services from the user profile [27]. Chang *et al.* developed a personalized service recommendation system (PSRS) in a home-care environment. The PSRS can provide appropriate services based on the user's preferences and habits recorded in the user profile. Through the user profile, the system will be able to automatically launch safety alerts, recommended services and healthcare services in the patient's house [28].

The feasible solution to generating personalized services from the user profile entails two processes: generating goal activities from the user profile and defining the services which can fulfill those activities. In generating the goal activities, a "goal ontology" is created to represent the relationship between the user profile and goal activities. In defining the services, a "task ontology" is created to specify the relationship between the goal activities and the services. This solution has been widely used in many studies [26–28]. Here, this solution is also cited to support the configuration of personalized services configuration.

2.4. Ontology-Based Approach for Knowledge Intensive Applications

An ontology describes the concepts and relationships that are important in a particular domain, providing a vocabulary for that domain as well as a computerized specification of the meaning of terms used in the vocabulary. The aim of an ontology is to generically formalize domain knowledge and provide a common understanding of a domain, which may be used and shared by applications and groups. It has been widely adopted in the business and scientific communities as a way to share, reuse and process domain knowledge [29]. Ontologies are also central to many applications in fields including information management, systems integration and semantic web services [30]. Many studies have also demonstrated that ontology is essential for the development of knowledge-oriented systems.

In the critical medicine and health care fields, ontologies are valued and broadly applied to building knowledge intensive services, applications, and systems. Ontology-based computational models have been proposed to facilitate effective management and acquisition of medical knowledge [31,32], and to serve as the foundation for medical diagnosis and treatment [33–35]. The diagnosis and treatment systems are constructed based on ontologies and rules. The ontologies define the medical knowledge models, such as the physiological symptoms, disease features, and treatments. The rules specify the relationships between symptoms, disease, and treatments to determine the diagnosis and period of treatment. Moreover, Su and Peng addressed the ontological and epistemological issues of information services through the example of OntoRis, an ontology-based rehabilitation service designed to assist patients in acquiring actionable knowledge about his/her prescribed rehabilitation, and to expedite recovery by providing suggestions and advice drawn from evidence-based medicine [36].

2.5. Web Service Approach in Information Integration

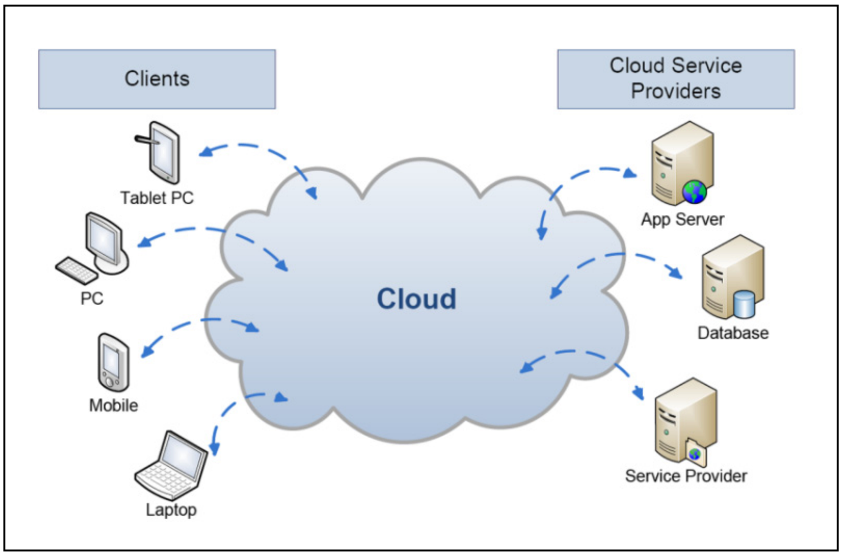
Techniques have been described for facilitating the use of invocable services by configurable applications. In at least some situations, the invocable services are Web services (WSes) or other network-accessible services that are made available by providers of the services for use by others in exchange for fees defined by the service providers [37]. Such techniques may be used, for example, in conjunction with an electronic WS marketplace via which third-party WS providers make their WSes available to third-party WS consumers who locate and purchase access to those WSes, thus allowing a user to dynamically create a new composite WS based on one or more WSes available from other WS providers and that reflects any constraints of the WS marketplace, with the composite WS available for use by other WS consumers [38].

Recent Web 2.0 developments show how easy it can be to dynamically link or compose IT components to achieve the original SOA goals of flexibility, reusability, or reduction by relatively simple means [39,40]. At this point the idea of representational state transfer (REST) [41] is often used in a Resource-oriented Architecture (ROA), which uses only Hypertext Transfer Protocol (HTTP) methods such as POST, GET, PUT and DELETE as actions. REST specifies a collection of architectural principles defining how data resources are represented and addressed. When the REST architectural principles are applied, as a whole, they provide enhanced scalability, interface generality, and deployment independence, while reducing interaction latency and encapsulating legacy systems [41]. Given these advantages and characteristics, REST WSes are broadly applied in information system integration in research fields including smart homes [42], e-Learning [43] and engineering collaboration [44], *etc.*

3. IAServ Design and Architecture

The design of IASERV centers around seamless integration of such enabling technologies as ontologies, Web Services and the Foundation for Intelligent Physical Agents (FIPA)-compliant agent framework Java Agent Development Environment (JADE) [45] to provide personalized and context-aware healthcare services. The services are implemented as web services and deployed in a cloud computing environment for economic, scalable, and ubiquitously accessible service provision as illustrated in Figure 1.

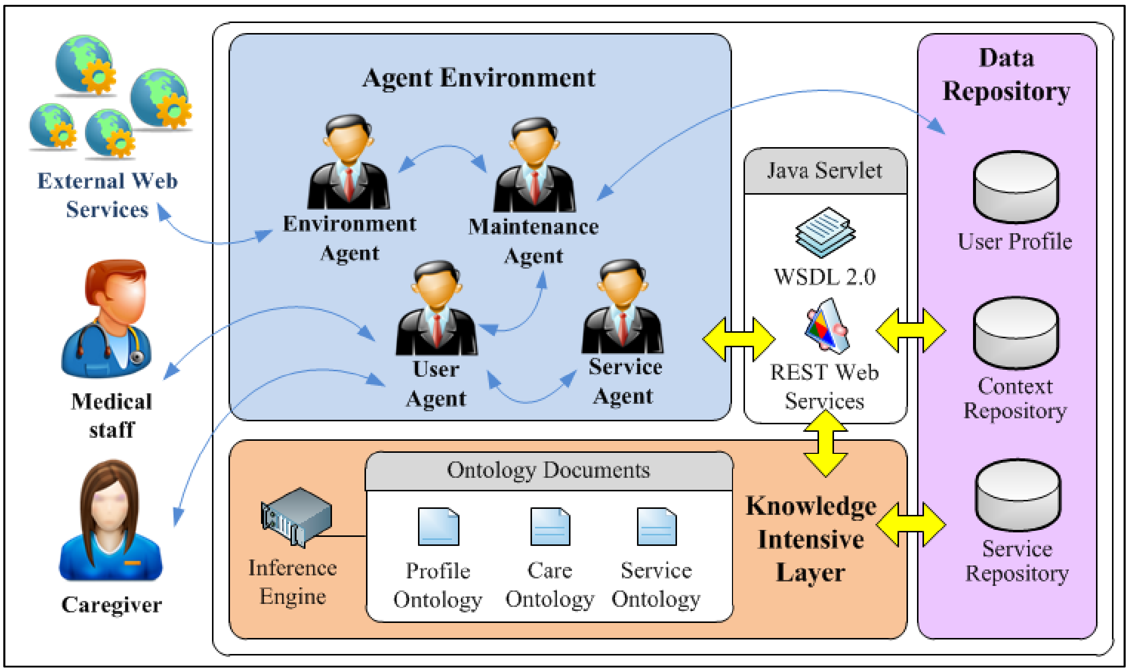
Figure 1. The concept of cloud computing.



Functionally, the IAServ is able to infer the appropriate care plan which defines the needed healthcare activities through the patient's profile and relevant external web services. The care-givers administrate healthcare according to a pre-set medical goal. Care-givers can also access contextually-relevant information (e.g., the impact of recent weather changes for a person suffered from hypertension) through any mobile device. As shown in Figure 2, the IAServ architecture comprises four main components: (1) the Knowledge Intensive Layer; (2) the cloud-based web services; (3) the agent environment; and (4) the Data Repository.

The Data Repository Layer stores the patient's personal information and the web services description. The Web Services provide the web applications related to the tasks encapsulated in the care plan such as reminders, weather information, *etc.* These information sources play vital role in providing personalized and context-aware healthcare services. The agents in the Agent Environment serve as a mechanism for communication between the users and applications. They also execute the healthcare services defined in the care plan by dispatching the Web Services with real-time contextually-aware information through the HTTP protocol. The details of IAServ components are depicted as follows:

Figure 2. The IAServ architecture.



3.1. Knowledge Intensive Layer

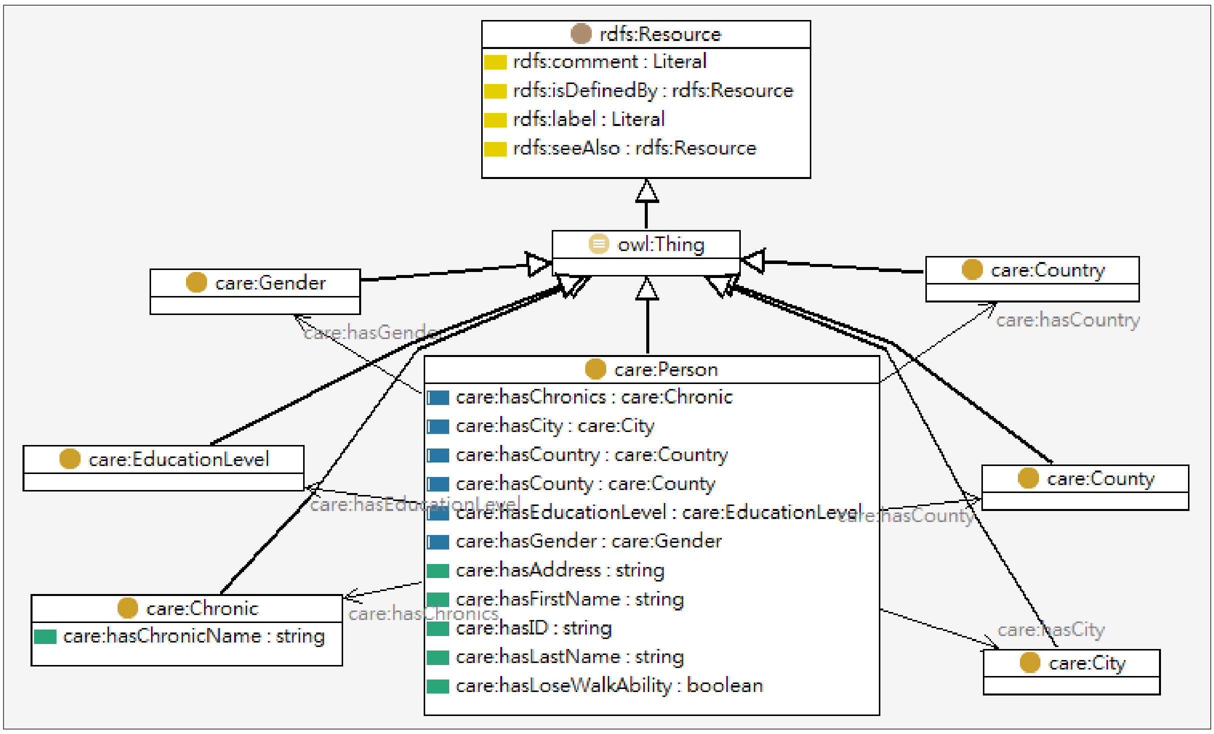
The Inference Engine in the Knowledge Intensive Layer plays a vital role in deriving personalized care plan based on a patient's profile. A health-care provider first compiles a patient's profile into an ontological format through a predefined User Interface (UI). The ontological profile will be subsequently used as input to the Inference Engine for generating a personalized care plan by using the Care ontology and Service ontology.

The personal profile includes a patient's basic information (e.g., name, sex, age, characteristics, preferences, interests, *etc.*) and personal states (e.g., clinical states, disabilities, impairments, *etc.*). Table 1 defines the classes in an ontological profile, and the partial profile ontology model is shown in Figure 3.

Table 1. Ontology classes of a personal profile.

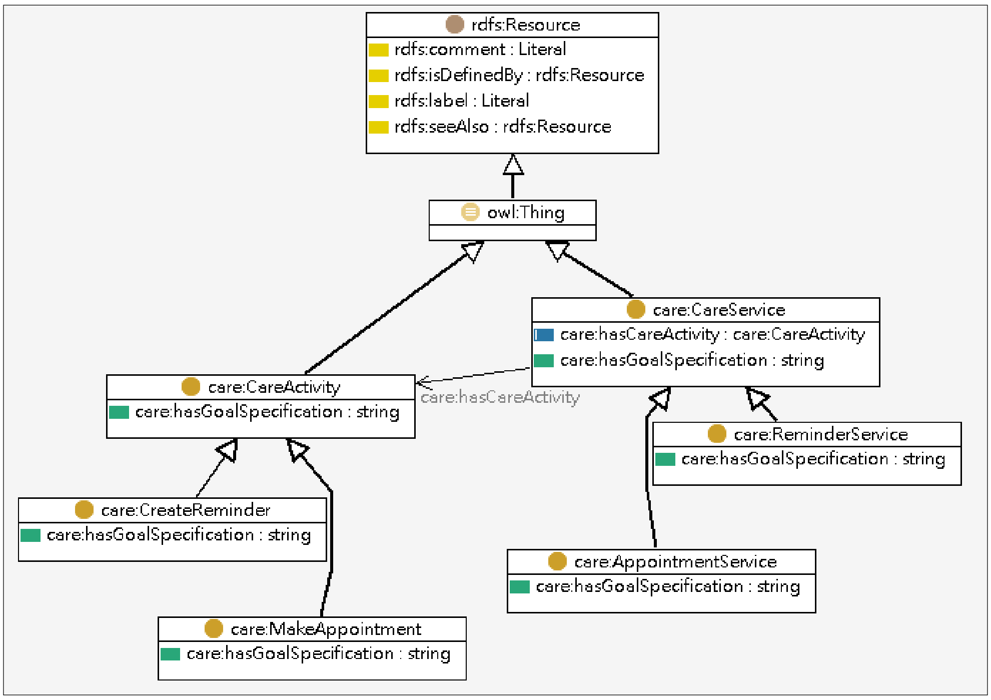
Class	Annotation
Person	Basic user information (e.g., name, sex, age, *etc.*)
Characteristics	General user characteristics (e.g., height, weight, *etc.*)
Ability	User abilities and disabilities, both mental and physical
Chronic Status	User diseases (e.g., hypertension, hemorrhagic stroke)
Living Conditions	Information relevant to the user's residence
Contact	Other persons, with whom the person is related (e.g., including relatives, friends, co-workers)
Preferences	User preferences (e.g., love cats, prefer blue color, dislike classical music)
Interests	User hobby or work-related interests. For example, "interested in sports", "interested in cooking"
Activity	User activities, hobby or work related
Education	User education issues, including for example university diplomas and languages
Profession	The user's profession
Expertise	computer, Internet, mobile devices, *etc.*

Figure 3. Partial model of user profile ontology.


rdfs:Resource
rdfs:comment : Literal
rdfs:isDefinedBy : rdfs:Resource
rdfs:label : Literal
rdfs:seeAlso : rdfs:Resource
owl:Thing
care:Gender
care:Country
care:hasGender
care:hasCountry
care:Person
care:hasChronics : care:Chronic
care:hasCity : care:City
care:hasCountry : care:Country
care:hasCounty : care:County
care:hasEducationLevel : care:EducationLevel
care:hasGender : care:Gender
care:hasAddress : string
care:hasFirstName : string
care:hasID : string
care:hasLastName : string
care:hasLoseWalkAbility : boolean
care:EducationLevel
care:hasEducationLevel
care:County
care:hasCounty
care:Chronic
care:hasChronicName : string
care:hasChronics
care:hasCity
care:City

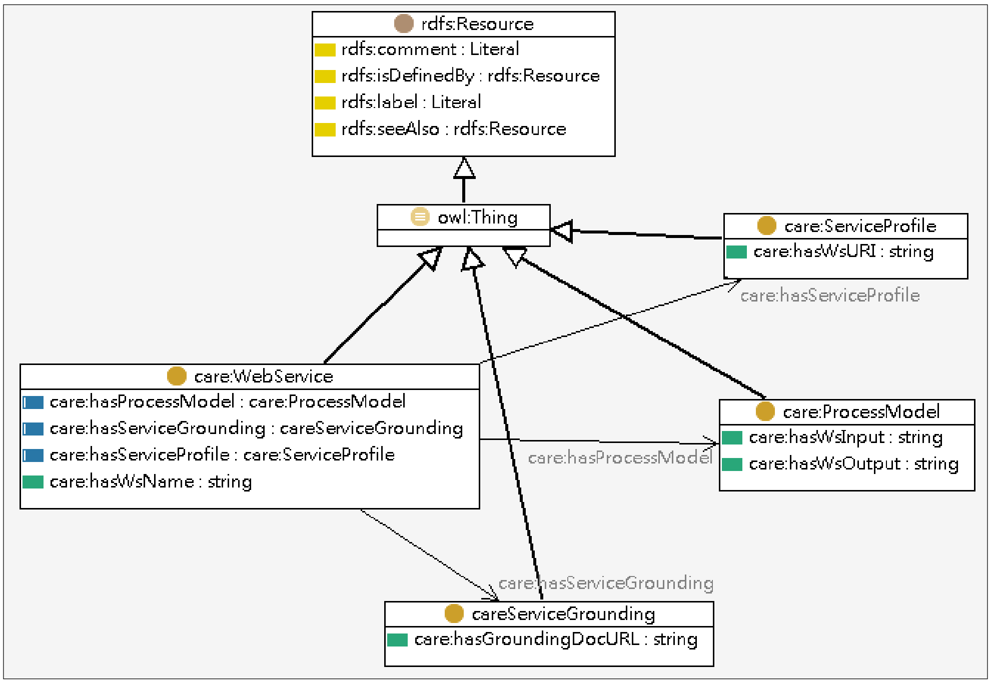
The Care Ontology encapsulates the relationship between the patient's profile and the relevant care services. The classes defined in Care Ontology include User State, Goal-specification, Care Service, and Care Activity as illustrated in Figure 4. The User State class indicates the status of a patient while the Goal-specification and the Care Service classes describe the intended medical goal and define the care services and tasks associated with specific sates respectively.

Figure 4. Partial model of the care ontology.


rdfs:Resource
rdfs:comment : Literal
rdfs:isDefinedBy : rdfs:Resource
rdfs:label : Literal
rdfs:seeAlso : rdfs:Resource
owl:Thing
care:CareService
care:hasCareActivity : care:CareActivity
care:hasGoalSpecification : string
care:CareActivity
care:hasGoalSpecification : string
care:hasCareActivity
care:ReminderService
care:hasGoalSpecification : string
care:CreateReminder
care:hasGoalSpecification : string
care:AppointmentService
care:hasGoalSpecification : string
care:MakeAppointment
care:hasGoalSpecification : string

The Service Ontology encapsulates the relationship between the care activities and the physical web services. The classes covered in the Service Ontology include Care Activity, Web Service, Service Profile, Process Model, and Service Grounding, as shown in Figure 5. The Care Activity specifies the tasks that need to be performed in a care service while the Web Service registers services that are available for executing tasks.

Figure 5. Partial model of the service ontology.

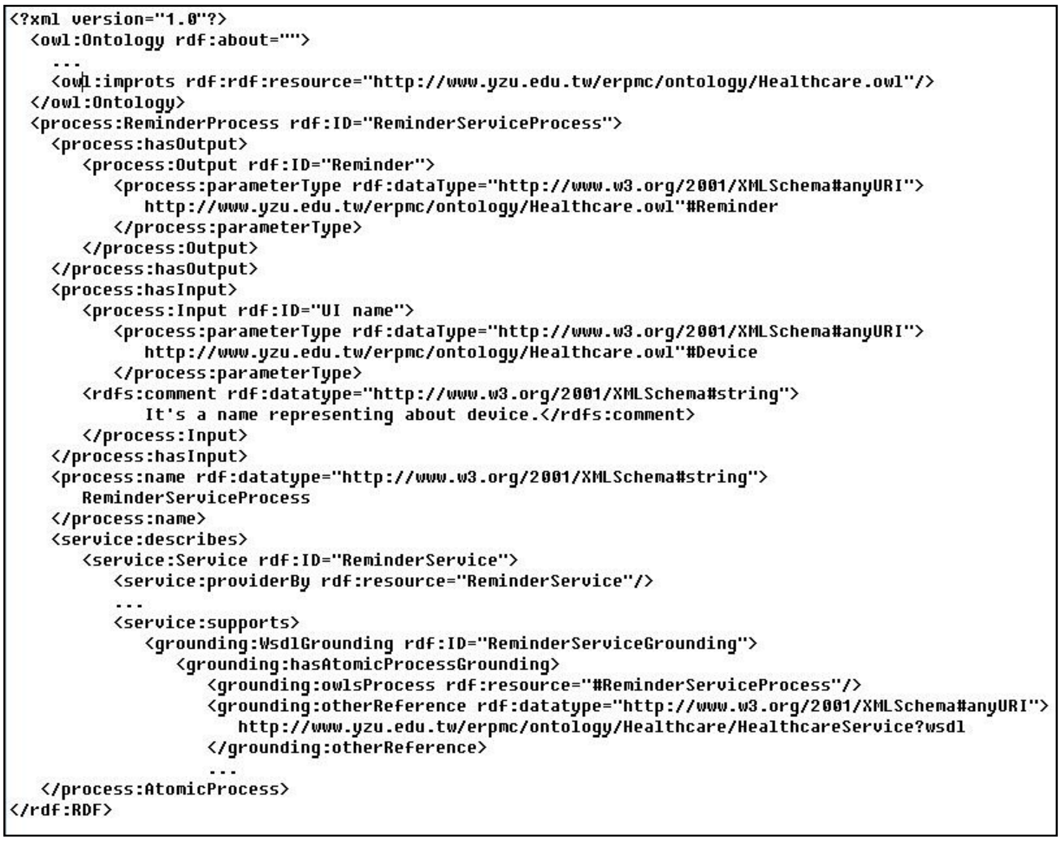


The OWL-S (Semantic Markup for Web Services) is adopted to describe Semantic Web Services and enable agents to automatically discover, invoke, compose, and monitor web services under specified constraints [46]. The OWL-S high level classes used in IAServ are depicted in Table 2. Figure 6 shows an OWL-S description of a reminder service as an example.

Table 2. High level OWL-S classes.

Class	Description
Service Profile	Includes service name and parameter annotations.
Process Model	Includes information, including input (input type), output and local.
Service Grounding	Defined by WSDL (Web Services Definition Language) 2.0 operations.

Figure 6. OWL-S description of the reminder service.



```
<?xml version="1.0"?>
  <owl:Ontology rdf:about="">
    ...
    <owl:improts rdf:rdf:resource="http://www.yzu.edu.tw/erpmc/ontology/Healthcare.owl"/>
  </owl:Ontology>
  <process:ReminderProcess rdf:ID="ReminderServiceProcess">
    <process:hasOutput>
       <process:Output rdf:ID="Reminder">
          <process:parameterType rdf:dataType="http://www.w3.org/2001/XMLSchema#anyURI">
             http://www.yzu.edu.tw/erpmc/ontology/Healthcare.owl"#Reminder
          </process:parameterType>
       </process:Output>
    </process:hasOutput>
    <process:hasInput>
       <process:Input rdf:ID="UI name">
          <process:parameterType rdf:dataType="http://www.w3.org/2001/XMLSchema#anyURI">
             http://www.yzu.edu.tw/erpmc/ontology/Healthcare.owl"#Device
          </process:parameterType>
       <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string">
             It's a name representing about device.</rdfs:comment>
       </process:Input>
    </process:hasInput>
    <process:name rdf:datatype="http://www.w3.org/2001/XMLSchema#string">
       ReminderServiceProcess
    </process:name>
    <service:describes>
       <service:Service rdf:ID="ReminderService">
          <service:providerBy rdf:resource="ReminderService"/>
          ...
          <service:supports>
             <grounding:WsdlGrounding rdf:ID="ReminderServiceGrounding">
                <grounding:hasAtomicProcessGrounding>
                   <grounding:owlsProcess rdf:resource="#ReminderServiceProcess"/>
                   <grounding:otherReference rdf:datatype="http://www.w3.org/2001/XMLSchema#anyURI">
                      http://www.yzu.edu.tw/erpmc/ontology/Healthcare/HealthcareService?wsdl
                   </grounding:otherReference>
                   ...
   </process:AtomicProcess>
</rdf:RDF>
```

The ontologies and reasoning rules are collaboratively carried out with the help of domain experts (e.g., physicians, therapeutists, *etc.*). The Inference Engine is represented by description logic rules. Based on the interaction between the personal information and environmental context, the rules encoded knowledge furnishes inferences for dynamic situation analysis and decision making. The logic rules and constraints are defined using the SPARQL Inferencing Notation (SPIN) framework [47], which allows us to leverage the fast performance and rich expressivity of SPARQL in IAServ. An example of the rule infers a hypertension patient who lives in Tainan and needs a "Cold Wave Advice" service. The patient will be informed when the outdoor temperature drops below 13 degrees Celsius:

```
SELECT ?person
WHERE {
   ?c a care:Context.
  ?c care:hasCity ?city.
  ?c care:hasTemperature ?tmp.

  ?person a care:Person .
  ?person care:hasChronics ?cr .
   ?cr rdfs:label ?crlbl .

   ?cs a care:ReminderService.
   ?cs rdfs:label ?csname .

   FILTER (?tmp<13 &&
                  regex(? city, “Tainan”) &&
                  regex(?cslbl, “Hypertension”) &&
                  regex(?csname, “Cold_Wave_Advice”)
  ).
}
```

3.2. Web Service in the Cloud

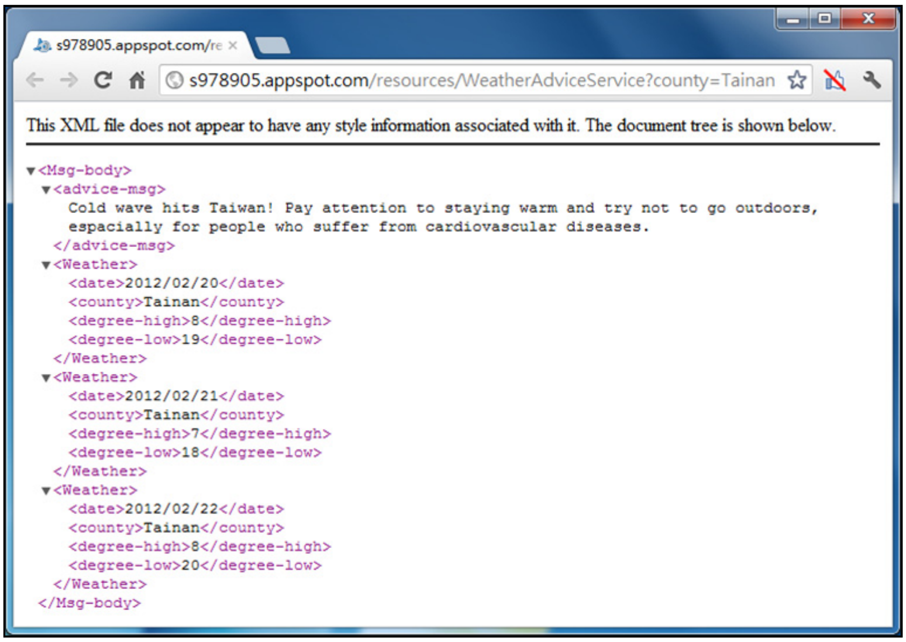
Web services can be described as a technology and platform-independent architecture where loosely coupled components communicate via interfaces over standard web protocols. Software, hardware, and data-centric designs maximize system efficiency, scalability, and network throughput. Such web services are ideal for use in distributed systems such as IAServ where scalability is critical, both in terms of processing power and communication efficiency. The web service can be invoked by a simple HTTP query to a defined URL. The response can be HTML, comma-delimited data, XML, or a more sophisticated document type (such as a spreadsheet).

The Web Services Definition Language (WSDL) 2.0 is an XML format for describing network services as a set of endpoints operating on messages containing either document-oriented or procedure-oriented information. WSDL 2.0 describes Web services in two levels: an XML-based reusable abstract interface and concrete details regarding how and where this interface can be accessed. The Version 2.0 of WSDL features better support for REST-based web services and implementation simplification, which are essential for the development of IAServ.

The web services in IAServ are written in Java and are deployed in the Google App Engine, which provides a free environment (up to a certain level of used resources) for developing and hosting web applications in Google-managed data centers. Future work will also consider other cloud-based platforms for deployment of our web services, including Amazon Web Services [48] and Microsoft’s Azure Services Platform [49].

Figure 7 shows an example of “Weather Advice” Web Service deployed on the Google App Engine [50]. Using simple HTTP request methods, the “Cold Wave Advice” Web Service can be easily accessed via the URL, “http://s978905.appspot.com/resources/WeatherAdviceService”, with a request parameter, “county”, and responds with an XML-formatted document. Figure 8 provides a sample Java code for accessing the Weather Advice Web Service.

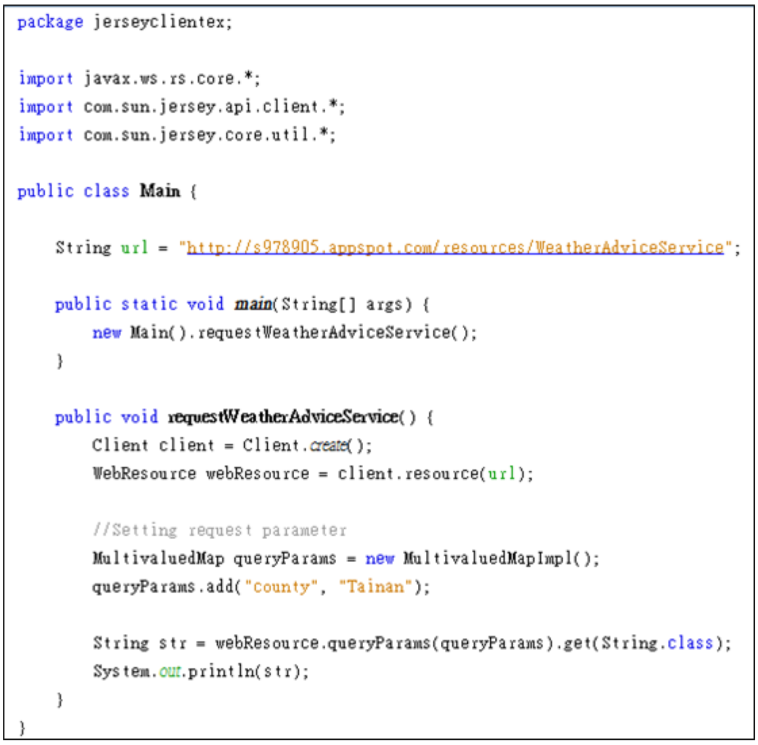
Figure 7. The “Weather Advice” web service in Google App engine.


s978905.appspot.com/re ×
s978905.appspot.com/resources/WeatherAdviceService?county=Tainan

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
▼<Msg-body>
  ▼<advice-msg>
     Cold wave hits Taiwan! Pay attention to staying warm and try not to go outdoors,
     espacially for people who suffer from cardiovascular diseases.
   </advice-msg>
  ▼<Weather>
     <date>2012/02/20</date>
     <county>Tainan</county>
     <degree-high>8</degree-high>
     <degree-low>19</degree-low>
   </Weather>
  ▼<Weather>
     <date>2012/02/21</date>
     <county>Tainan</county>
     <degree-high>7</degree-high>
     <degree-low>18</degree-low>
   </Weather>
  ▼<Weather>
     <date>2012/02/22</date>
     <county>Tainan</county>
     <degree-high>8</degree-high>
     <degree-low>20</degree-low>
   </Weather>
 </Msg-body>
```

Figure 8. The sample Java code of accessing the “Weather Advice” web service.



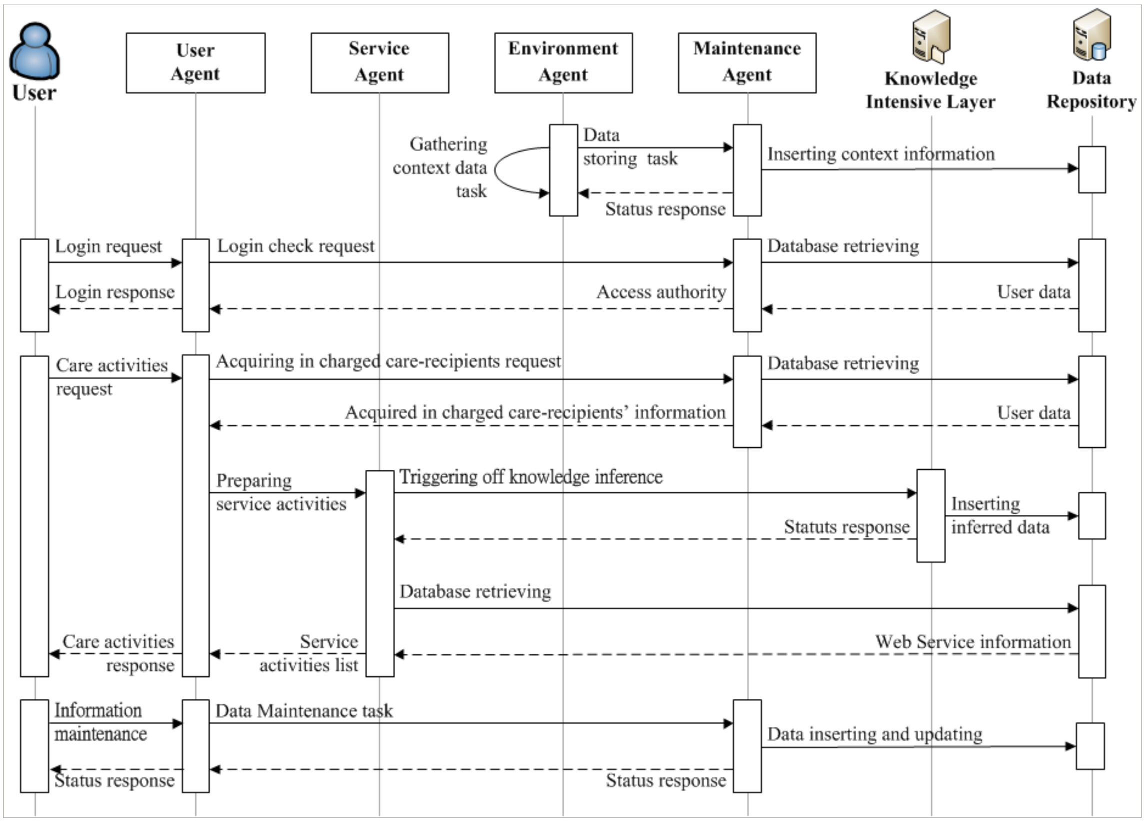
```java
package jerseyclientex;

import javax.ws.rs.core.*;
import com.sun.jersey.api.client.*;
import com.sun.jersey.core.util.*;

public class Main {

    String url = "http://s978905.appspot.com/resources/WeatherAdviceService";

    public static void main(String[] args) {
        new Main().requestWeatherAdviceService();
    }

    public void requestWeatherAdviceService() {
        Client client = Client.create();
        WebResource webResource = client.resource(url);

        //Setting request parameter
        MultivaluedMap queryParams = new MultivaluedMapImpl();
        queryParams.add("county", "Tainan");

        String str = webResource.queryParams(queryParams).get(String.class);
        System.out.println(str);
    }
}
```

3.3. Agent Environment

The Agent Environment is in charge of the management and coordination of all system applications and services. The agents would perform the defined care services in a care plan for a patient by dispatching the Web Services with the contextual data through the HTTP protocol. The agent environment in IAServ comprises four agents and each one of them with specific roles, capabilities and characteristics is described as follows. The interaction between the agents is illustrated in Figure 9.

- *Environment Agent*: The Environment Agent is responsible for managing the environmental data (e.g., temperature, humidity, location, *etc.*). It works with the Context Repository and connects with the External Web Services to collect the environmental data from sources such as Yahoo! Weather [51], and stores the data into the Context Repository. When performing contextually relevant services, the Environment Agent offers the corresponding contextual data to the Service Agent for the provision of contextually-aware services.
- *Service Agent*: The Service Agent is designed to monitor the implementation of the personalized care plan. When it detects the scheduled care plan, it will use contextual data to prepare information from relevant web services and subsequently delegate task execution to the User Agent.
- *User Agent*: The User Agent is responsible for accessing the web service and performing the care services on behalf of patients when it receives tasks from the Service Agent. Following data provided by the Service Agent, the User Agent makes a request to the web service with the needed input data parameters via the HTTP protocol. The resulting information is then published to the user's device by the User Agent to complete the care service.
- *Maintenance Agent*: Expert knowledge and patient conditions and experience may change over time. These changes can be accommodated appropriately in the ontologies through the Maintenance Agent. The Maintenance Agent plays an important role in dealing with the dynamic aspects of the health care environment. It allows experts to update the knowledge, instances, and rules encapsulated in the ontologies, as illustrated in Figure 10.

Figure 9. Interactions among agents, knowledge intensive layer, and data repository.



3.4. Data Repository

The Data Repository serves both as storage for the User Profile, Context, and Services and as a central processing point for the requests of all data. The User Profile encapsulates general information regarding care-givers and patients, which is governed by medical staff. A care-giver's profile comprises personal information, qualifications, track records, and experiences while a patient's profile contains his/her demographics, location, contact information, and services prescribed/subscribed.

The Context Repository stores the contextual ontology, instances (e.g., temperature, humidity, air quality) and rules, and allows care-givers to query or manage contextualized knowledge. The Service Repository stores the description of available web services including service features, inputs, outputs, locations (URI), *etc.*, and allows care-givers to manage the services.

4. Implementation

4.1. Hardware and Software Configuration

The IAServ is implemented on the Windows™ platform. To derive personalized a care plan based on a user profile, an inference engine is needed to take an "asserted" model as input and create an "inferred" model as output in the context of ontology. The past decade has seen the development of several inference engines including OWL DL tableaux reasoners like Pellet, Flora-2 implemented F-OWL, RacerPro, FacT++, Minerva, and Rule engines such as Jena [52]. Most implementation require the integration of various inference engines. For instance, the system requires running SPARQL or Jena which have no built-in knowledge of OWL DL on top of an OWL-compliant reasoning service such as Pellet. Considering potential future expansion of IAServ, we adopted TopBraid™ Composer (TBC) [53] as our development platform. The Eclipse-based TBC is not only a visual ontology editor but also a knowledge-based framework capable of integrating such built-in inference engines, as TopSPIN [54], Pellet, Jenna, *etc.* through hybrid inference-chaining. TopSPIN, the embedded rule-based reasoner in TBC, which supports SPARQL query and SPIN rules, is used as the main inference engine in IAServ. Its built-in ability to run multiple inference engines shortens the IAServ development cycle.

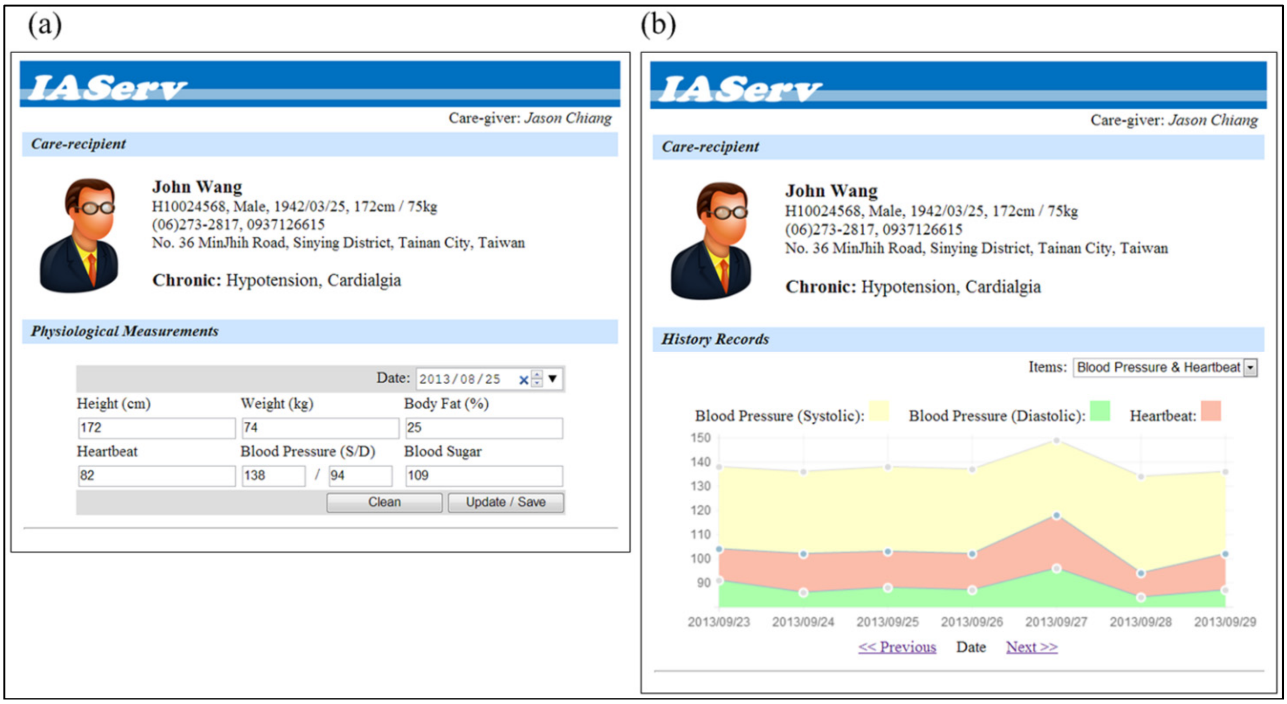
The Agent Environment in IAServ is implemented in the FIPA-compliant JADE framework. To run JADE, we need to install the Java 2 Run-time Environment (JRE). However, we recommended installing the full Java 2 Source Development Kit (J2SDK) because it is used in the development and compilation of our own agents in JADE.

4.2. Usage Scenario and Demonstration

The IAServ allows healthcare professionals to administer a personalized service plan and subscribe to needed services in the healthcare context by using practically any Internet-enabled device. In this section, we describe an implementation scenario to illustrate IAServ usage. "John Wang is 72 years old. Despite his age and having various medical conditions, he prefers to remain in his own home. John suffered a hemorrhagic stroke with mild claudication and hypertension, and needs to go to the hospital three times a week (Monday, Wednesday, and Thursday every week at 9:30 A.M.) for rehabilitation and to take blood pressure medicine twice daily. However, John was difficulty walking to the hospital because of claudication, and frequently forgets to take his medicine. In John's case, the care-givers may use IAServ to provide personalized healthcare plan to address these problems." [55].

To produce a personalized healthcare plan which fulfills John's desirable goal "remain living in his own home", a healthcare professional first submits John's profile to IAServ via the Maintenance Agent's GUI, as shown in Figure 10. Figure 10a shows the GUI for care-giver to record basic physiological signals, and the history records are also provided for observation and monitoring of the patient's performance, as shown in Figure 10b. The profile is subsequently converted into an ontology specification as depicted in Figure 11 and stored in the personal information repository. The inference engine then generates a personalized care plan for John based on the CARE ontology and Service ontology.

Figure 10. (**a**) User agent GUI for user profile submission and (**b**) for performance monitoring.



(a)

Care-giver: *Jason Chiang*

Care-recipient

John Wang
H10024568, Male, 1942/03/25, 172cm / 75kg
(06)273-2817, 0937126615
No. 36 MinJhih Road, Sinying District, Tainan City, Taiwan

Chronic: Hypotension, Cardialgia

Physiological Measurements

Date: 2013/08/25

Height (cm)	Weight (kg)	Body Fat (%)
172	74	25
Heartbeat	Blood Pressure (S/D)	Blood Sugar
82	138 / 94	109

(b)

Care-giver: *Jason Chiang*

Care-recipient

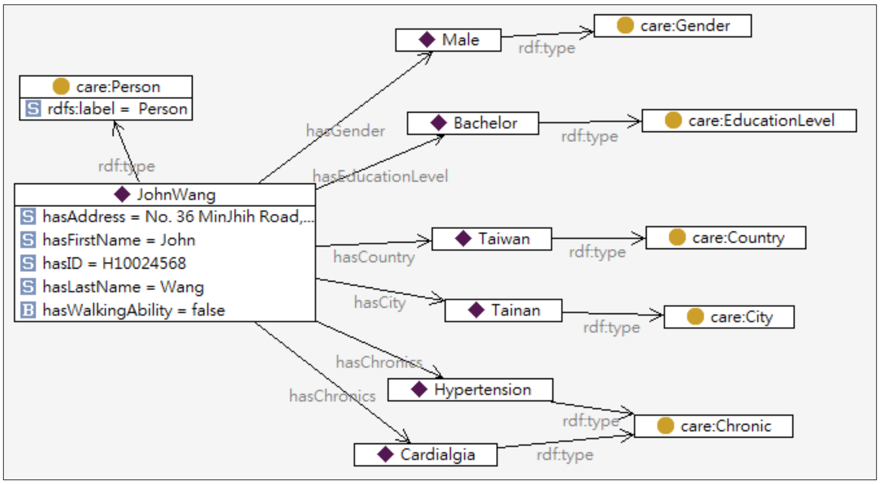
John Wang
H10024568, Male, 1942/03/25, 172cm / 75kg
(06)273-2817, 0937126615
No. 36 MinJhih Road, Sinying District, Tainan City, Taiwan

Chronic: Hypotension, Cardialgia

History Records

Items: Blood Pressure & Heartbeat

Figure 11. Partial ontology model of John's profile.



Given John's clinical state of "Hemorrhagic Stroke", the inference engine generates a relevant care service "Medical Appointment Service" based on the Care Ontology, which will be included in John's care plan. Similarly, the inference engine infers the clinical state "Hypertension". The generated care services "Reminder Service" and "Weather Service" will also be included in John's care plan. The partial Care Ontology is shown in Figure 12.

Figure 12. The partial ontology model of "Medical Appointment Service".

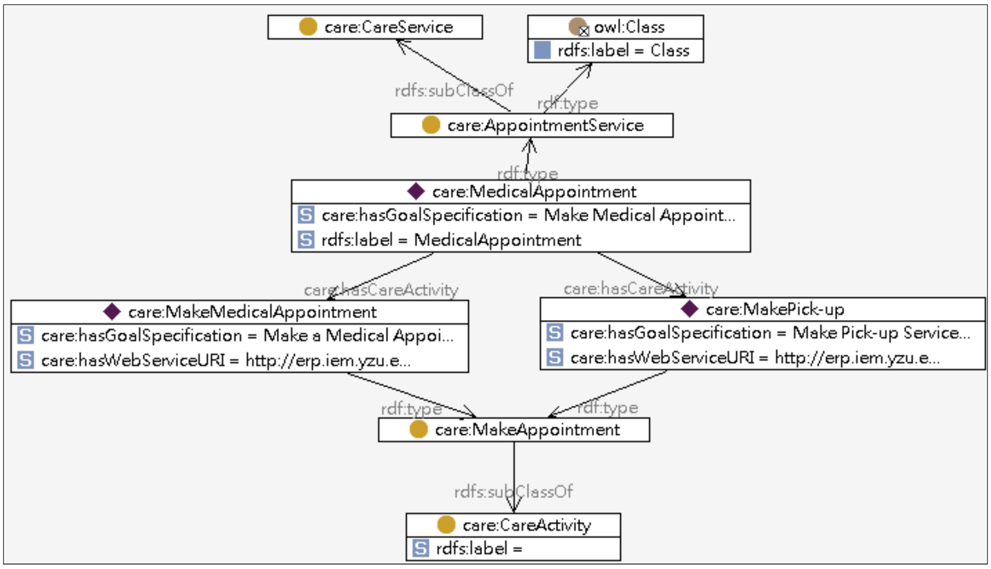
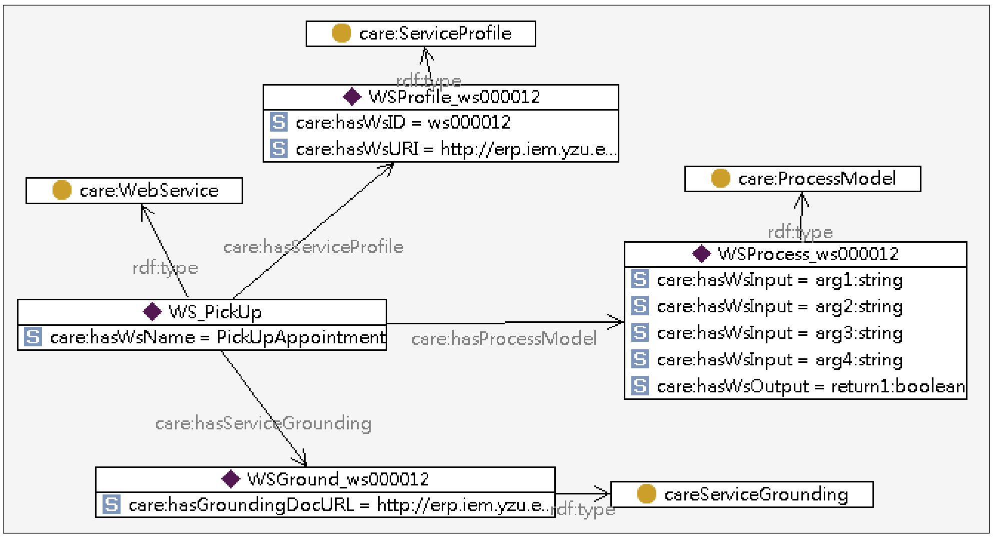

care:CareService
owl:Class
rdfs:label = Class
rdfs:subClassOf
rdf:type
care:AppointmentService
rdf:type
care:MedicalAppointment
care:hasGoalSpecification = Make Medical Appoint...
rdfs:label = MedicalAppointment
care:hasCareActivity
care:hasCareActivity
care:MakeMedicalAppointment
care:hasGoalSpecification = Make a Medical Appoi...
care:hasWebServiceURI = http://erp.iem.yzu.e...
care:MakePick-up
care:hasGoalSpecification = Make Pick-up Service...
care:hasWebServiceURI = http://erp.iem.yzu.e...
rdf:type
rdf:type
care:MakeAppointment
rdfs:subClassOf
care:CareActivity
rdfs:label =

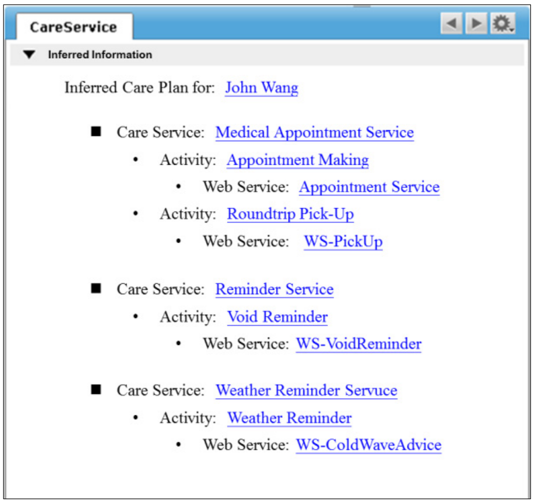
Figure 13. Partial ontology model of "Pick-up Web Service".


care:ServiceProfile
rdf:type
WSProfile_ws000012
care:hasWsID = ws000012
care:hasWsURI = http://erp.iem.yzu.e...
care:WebService
care:ProcessModel
rdf:type
care:hasServiceProfile
rdf:type
WSProcess_ws000012
care:hasWsInput = arg1:string
care:hasWsInput = arg2:string
care:hasWsInput = arg3:string
care:hasWsInput = arg4:string
care:hasWsOutput = return1:boolean
WS_PickUp
care:hasWsName = PickUpAppointment
care:hasProcessModel
care:hasServiceGrounding
WSGround_ws000012
care:hasGroundingDocURL = http://erp.iem.yzu.e...
rdf:type
careServiceGrounding

If the generated care plan is approved by John's care-giver, the tasks required to complete the services in the plan will be further inferred based on the Service ontology as shown in Figure 13. In this example, the "Medical Appointment Service" may consist of "Appointment Making" and "Round-trip Pickup" (as John suffers from claudication) to aid John in making the appointment. A "Voice Reminder" activity using an event-driven calendar may be generated to provide a "Reminder Service" for John to take his medicine. If John's profile indicates he has hearing problems, the reminders can automatically be produced as text-messages or video-messages. Weather has a significant impact on blood pressure. Because John suffers from hypertension, changes in weather will automatically trigger appropriate reminders for John to closely monitor his blood pressure.

Once the services and related tasks are inferred, a care plan is generated and stored in the personal information repository as shown in Figure 14. The web services associated with the tasks can be subscribed for plan execution.

Figure 14. Inferred information page of care services.

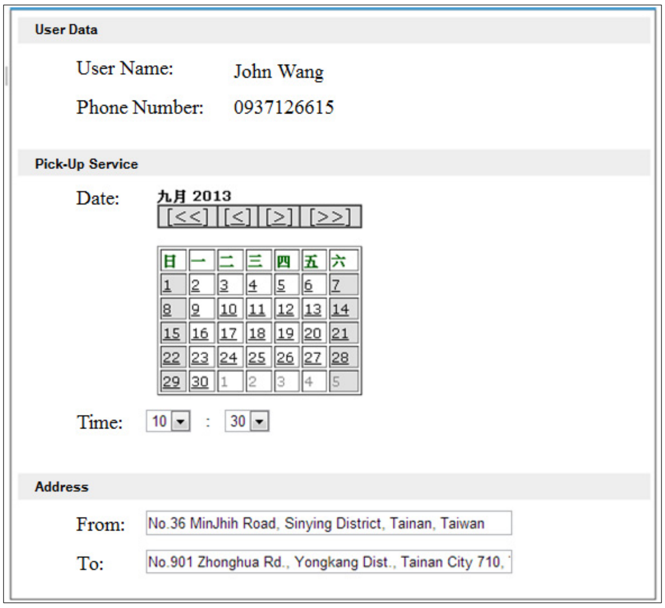


Inferred Care Plan for: John Wang

- Care Service: Medical Appointment Service
  - Activity: Appointment Making
    - Web Service: Appointment Service
  - Activity: Roundtrip Pick-Up
    - Web Service: WS-PickUp
- Care Service: Reminder Service
  - Activity: Void Reminder
    - Web Service: WS-VoidReminder
- Care Service: Weather Reminder Servuce
  - Activity: Weather Reminder
    - Web Service: WS-ColdWaveAdvice

John's care plan may be adjusted periodically by his physician, depending on his health conditions or the outcome of plan execution. For example, if arranging pick-up service for John on Monday, Wednesday, and Thursday at 9:30 A.M. every week can be accomplished using a built-in GUI, as illustrated in Figure 15.

Figure 15. Pick-up web service settings form.



User Data

User Name: John Wang

Phone Number: 0937126615

Pick-Up Service

Date: 九月 2013

日	一	二	三	四	五	六
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	1	2	3	4	5

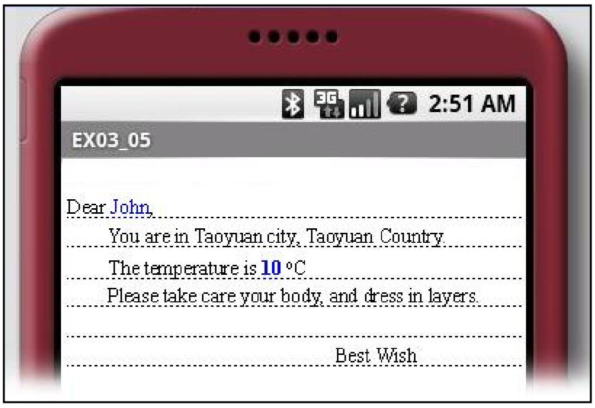
Time: 10 : 30

Address

From: No.36 MinJhih Road, Sinying District, Tainan, Taiwan

To: No.901 Zhonghua Rd., Yongkang Dist., Tainan City 710, The web services are implemented in Java using a REST-style architecture and deployed in a cloud environment. The services can therefore be reliably accessed from anywhere on virtually any Internet-enabled device, as shown in Figure 16.

Figure 16. Example of cold wave adviser service message.



Dear John,

You are in Taoyuan city, Taoyuan Country.

The temperature is 10 °C

Please take care your body, and dress in layers.

Best Wish

4.3. System Usability Evaluation

System usability/readability was evaluated through a questionnaire distributed to twenty care-givers and twenty patient users. Their respective profiles are illustrated in Tables 3 and 4.

Table 3. User profiles (care-givers).

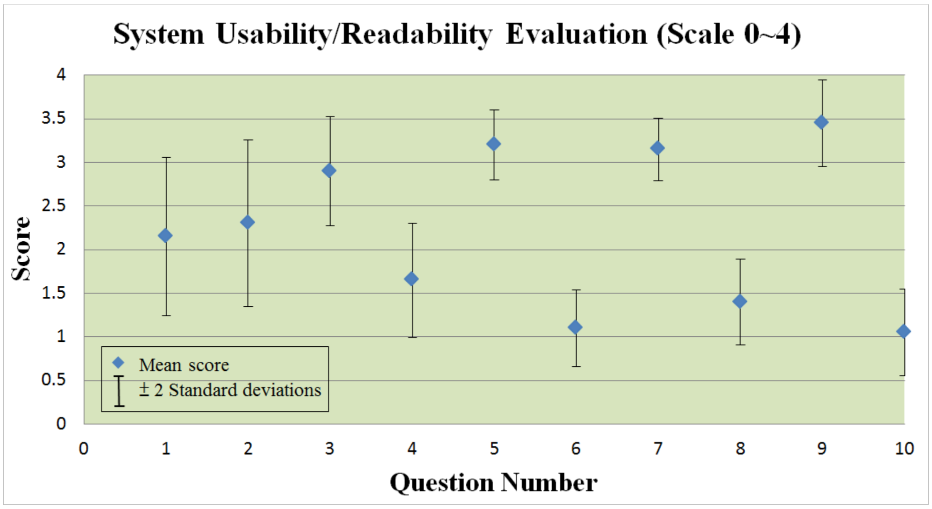
Personal information	Number	%
Age		
20–30	4	20.0
30–40	13	65.0
40 or above	3	15.0
Total	20	100.0
Gender		
Male	8	40.0
Female	12	60.0
Total	20	100.0
Service year		
1 or below	6	30.0
1–3	7	35.0
3–5	6	30.0
5 or above	1	5.0
Total	20	100.0
Experience of using Internet		
Yes	15	75.0
No	5	25.0
Total	20	100.0
Experience of using Smart Mobile Device		
Yes	11	55.0
No	9	45.0
Total	20	100.0

Table 4. User profiles (patients).

Personal information	Number	%
Age		
45–55	10	50.0
55–65	8	40.0
65 or above	2	10.0
Total	20	100.0
Gender		
Male	8	40.0
Female	12	60.0
Total	20	100.0
Patient with numbers of chronic disease		
0	4	20.0
1–2	11	55.0
3 or above	5	25.0
Total	20	100.0

Figures 17 and 18 present a statistical summary of usability/readability respectively obtained from the questionnaires for professional and general users. For each section, the mean score and a range representing +/−2 standard deviations is displayed.

Figure 17. System usability/readability evaluation for health-care professional users.

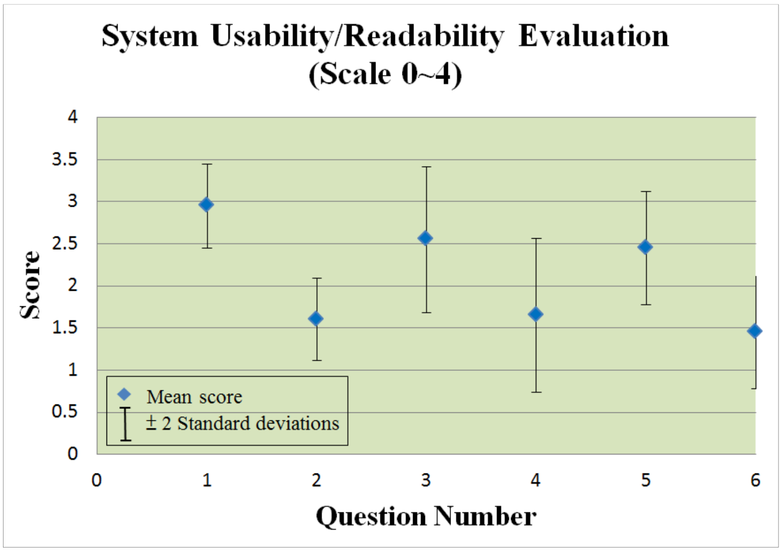


In our questionnaire, question validity is reinforced by posing each question twice: once as a positive statement and followed by a negative statement The figure indicates that the usability/readability of the IAServ is reasonably high. The question numbers below correspond to the x-axis in Figure 17:

1. IAServ was easy to use.
2. I would need a technical person to assist me.
3. IAServ is useful in providing better services to patients.
4. IAServ wasn't useful in improving service quality.
5. IAServ significantly reducing the workload of care-givers.

6. The system did not significantly assist care-givers.
7. The features offered by IAServ are sufficient.
8. The features offered by IAServ are insufficient.
9. IAServ significantly contributes to improving the delivery of health care services.
10. Care-givers won’t find IAServ isn’ helpful in serving patients.

Figure 18. System usability/readability evaluation for patients.



The question numbers below correspond to the x-axis in Figure 18:

1. I would like to offer health care services using IAServ.
2. I’m not interested in using the IAServ platform.
3. The features offered by IAServ are sufficient.
4. The features offered by IAServ are insufficient.
5. IAServ was useful in helping to maintain my health.
6. IAServ was useless for managing my health.

5. Concluding Remarks and Future Work

Personalized and context-aware healthcare services can be provided through proposed IAServ system, based on seamless integration of such enabling technologies as Ontology, Web Services and the Foundation for Intelligent Physical Agents (FIPA)-compliant agent framework Java Agent Development Environment (JADE). The IAServ employs ontologies to improve the integration and orchestration of all the technologies involved such as web services, context-aware healthcare services, and software agents. Rules of inference are formulated, using structural information in ontologies to derive personalized, context-aware healthcare plan for elderly patients. The proposed system design can serve as the foundation for an economic, scalable, and robust Web-based healthcare platform. Table 5 presents a comparison with similar studies discussed in Section 2 to highlight the advantages and contributions of the proposed IAServ. While the comparison might not be comprehensive, it provides an overview of the differences between the IAServ and our understanding of other similar systems.

Table 5. Comparison of IAServ and other proposed systems.

Reference	**Service domain**	**Ambient intelligence**	**Multi-agent system**	**Context-aware reasoning**	**Web services based architecture**
Riva, 2003 [12]	Smart home environment	Yes	No	No	No
Su and Chiang, 2013 [20]	Home health care	Yes	Yes	Yes	No
Coronato *et al.*, 2009 [21]	Smart hospital environment	No	No	Yes	No
Fraile *et al.*, 2009 [22]	In-home Monitoring	Yes	Yes	Yes	No
Eslami *et al.*, 2010 [27]	Care service tailoring	Yes	No	Yes	No
Chang *et al.*, 2009 [28]	In-home Monitoring	Yes	No	No	No
Fenza *et al.*, 2012 [23]	In-home Monitoring	Yes	Yes	Yes	No
IAServ	Home health care	Yes	Yes	Yes	Yes

One critical issue yet to be addressed is the protection of user privacy and personal information. While the advantage of mobility in the agent paradigm helps us in developing distributed health care systems and reaching a larger mobile clientele, it also leaves the system vulnerable to attacks by malicious entities. A JADE plug-in (JADE-S [56]) was to provide security through access controls and secure communications, making the platform suitable for use in real environments. While JADE-S provides fundamental support for application-level security, it suffers from several shortcomings. For example, as SSL must be applied globally to all messages, and agent identity information cannot be accessed from the program. User data can be better protected by incorporating more sophisticated and robust security models in the IAServ server [57–59].

Conflicts of Interest

The authors declare no conflict of interest.

References

1. Valls, A.; Gibert, K.; Sánchez, D.; Batet, M. Using ontologies for structuring organizational knowledge in Home Care assistance. *Int. J. Med. Inform.* **2010**, *79*, 370–387.
2. Fergenson, M. TigerPlace: An innovative ‘Aging in Place’ community. *Am. J. Nurs.* **2013**, *113*, 68–69.
3. Emily, A.G.; Andrew, S.; Amanda, J.L.; Joan, K.D. A conceptual framework for examining the promise of the NORC program and Village models to promote aging in place. *J. Aging Stud.* **2012**, *26*, 273–284.
4. Swan, M. Emerging patient-driven health care models: An examination of health social networks, consumer personalized medicine and quantified self-tracking. *Int. J. Environ. Res. Public Health* **2009**, *6*, 492–525.

5. Riaño, D.; Real, F.; Campana, F.; Ercolani, S.; Annicchiarico, R. An Ontology for the Care of the Elder at Home. In *Artificial Intelligence in Medicine*; Combi, C., Shahar, Y., Abu-Hanna, A., Eds.; Springer: Berlin, Germany, 2009; Volume 5651, pp. 235–239.
6. Rosenberg, F.; Curbera, F.; Duftler, M.J.; Khalaf, R. Composing RESTful services and collaborative workflows: A lightweight approach. *IEEE Internet Comput.* **2008**, *12*, 24–31.
7. Maximilien, E.M.; Ranabahu, A.; Gomadam, K. An online platform for web APIs and service mashups. *IEEE Internet Comput.* **2008**, *12*, 32–43.
8. Konieczny, E.; Ashcraft, R.; Cunningham, D.; Maripuri, S. Establishing Presence within the Service-Oriented Environment. In Proceedings of the IEEE Aerospace Conference, McLean, VA, USA, 7–14 March 2009.
9. Wang, Y.H.; Liao, J.C. Why or Why Not Service Oriented Architecture. In Proceedings of the 2009 IITA International Conference on Services Science, Management and Engineering, Washington, DC, USA, 11–12 July 2009.
10. Kikhia, B. Acceptance of Ambient Intelligence (AmI) in Supporting Elderly People and People with Dementia. Ms.C. Thesis, Lulea University of Technology, Lulea, Sweden, 2008.
11. Loenen, E.J. On the Role of Graspable Objects in the Ambient Intelligence Paradigm. In Proceedings of the Smart Objects Conference, Grenoble, France, 15–17 May 2003.
12. Riva, G. Ambient intelligence in health care. *Cyberpsychol. Behav.* **2003**, *6*, 295–300.
13. Bieliková, M.; Krajcovic, T. Ambient Intelligence within a Home Environment. Available online: http://www.ercim.org/publication/Ercim_News/enw47/bielikova.html (accessed on 28 August 2013).
14. Schmidt, A. Interactive Context-Aware Systems Interacting with Ambient Intelligence. In *Ambient Intelligence*; Riva, G., Vatalaro, F., Davide, F., Alcañiz, M., Eds.; IOS Press: Amsterdam, Nederland, 2005; pp. 159–178.
15. Schilit, B.N.; Theimer, M.M. Disseminating active map information to mobile hosts. *IEEE Netw.* **1994**, *8*, 22–32.
16. Barkhuus, L. Context Information *vs.* Sensor Information: A Model for Categorizing Context in Context-Aware Mobile Computing. In Proceedings of the International Symposium on Collaborative Technologies and Systems, San Diego, CA, USA, 19–23 January 2003.
17. Chen, H. An Intelligent Broker Architecture for Pervasive Context-Aware Systems. Ph.D. Thesis, University of Maryland, Baltimore, MD, USA, December 2004.
18. Parupudi, G.; Evans, S.S.; Holtgrewe, B.J.; Reus, E.F. Context Aware Systems and Methods Utilizing Hierarchical Tree Structures. U.S. Patent 7743074, 29 June 2010.
19. Bricon-Souf, N.; Newman, C.R. Context awareness in health care: A review. *Int. J. Med. Inform.* **2007**, *76*, 2–12.
20. Su, C.J.; Chiang, C.Y. Pervasive community care platform: Ambient Intelligence leveraging sensor networks and mobile agents. *Int. J. Syst. Sci.* **2013**, doi:10.1080/00207721.2013.807384.
21. Coronato, A.; Esposito, M.; Pietro, G.D. A multimodal semantic location service for intelligent environments: An application for Smart Hospitals. *Pers. Ubiquitous Comput.* **2009**, *13*, 527–538.
22. Fraile, J.A.; Bajo, J.; Lancho, B.P.; Sanz, E. HoCa Home Care Multi-agent Architecture. In *International Symposium on Distributed Computing and Artificial Intelligence*; Corchado Rodríguez, J.M., Rodriguez, S., Llinas, J., Molina, J.M., Eds.; Springer-Verlag: Berlin/Heidelberg, Germany, 2009; Volume 50, pp. 52–61.

23. Fenza, G.; Furno, D.; Loia, V. Hybrid approach for context-aware service discovery in healthcare domain. *J. Comput. Syst. Sci.* **2012**, *78*, 1232–1247.
24. Meulendijk, M.; van de Wijngaert, L.; Brinkkemper, S.; Leenstra, H. AmI in good care? Developing design principles for ambient intelligent domotics for elderly. *Inform. Health Soc. Care* **2011**, *36*, 75–88.
25. Gauch, S.; Speretta, M.; Chandramouli, A.; Micarelli, A. User Profiles for Personalized Information Access. In *The Adaptive Web*; Brusilovsky, P., Kobsa, A., Nejdl, W., Eds.; Springer-Verlag: Berlin/Heidelberg, Germany, 2007; pp. 54–89.
26. Golemati, M.; Katifori, A.; Vassilakis, C.; Lepouras, G.; Halatsis, C. Creating an Ontology for the User Profile: Method and Applications. In Proceedings of the First International Conference on Research Challenges in Information Science, Ouarzazate, Morocco, 23–26 April 2007.
27. Eslami, M.Z.; Zarghami, A.; Sapkota, B.; Sinderen, M. Service Tailoring: Towards Personalized Homecare Services. In Proceedings of the 4th International Workshop on Architectures, Concepts and Technologies for Service Oriented Computing (ACT4SOC), Athens, Greece, 23 July 2010.
28. Chang, Y.K.; Yang, C.L.; Chang, C.P.; Chu, C.P. A Personalized Service Recommendation System in a Home-Care Environment. In Proceedings of the 15th International Conference on Distributed Multimedia Systems, San Francisco, CA, USA, 10–12 September 2009.
29. Chang, X.; Sahin, A.; Terpenny, J. An ontology-based support for product conceptual design. *Robot. Comput. Integr. Manuf.* **2008**, *24*, 755–762.
30. Horridge, M.; Jupp, S.; Moulton, G.; Rector, A.; Stevens, R.; Wroe, C. A Practical Guide to Building OWL Ontologies Using the Protege 4 and CO-ODE Tools. Available online: http://owl.cs.manchester.ac.uk/tutorials/protegeowltutorial/resources/ProtegeOWLTutorialP4_v1_1.pdf (accessed on 28 August 2013).
31. Juarez, J.M.; Riestra, T.; Campos, M.; Morales, A.; Palma, J.; Marin, R. Medical knowledge management for specific hospital departments. *Expert Syst. Appl.* **2009**, *36*, 12214–12224.
32. Kola, J.; Wheeldin, B.; Rector, A. Lessons in Building OWL Ontology Driven Applications: OCHWIZ—An Occupational Health Application. In Proceedings of the UK e-Science All Hands Conference, Nottingham, UK, 10–13 September 2007.
33. Huang, M.J.; Chen, M.Y. Integrated design of the intelligent web-based Chinese Medical Diagnostic System (CMDS)—Systematic development for digestive health. *Expert Syst. Appl.* **2007**, *32*, 658–673.
34. Ángel, G.C.; Alejandro, R.; Mencke, M.; Miguel, G.B.J.; Colomo-Palacios, R. ODDIN: Ontology-driven differential diagnosis based on logical inference and probabilistic refinements. *Expert Syst. Appl.* **2010**, *37*, 2621–2628.
35. Alexandrou, D.A.; Skitsas, I.E.; Mentzas, G.N. A holistic environment for the design and execution of self-adaptive clinical pathways. *IEEE Trans. Inf. Technol. Biomed.* **2010**, *15*, 108–118.
36. Su, C.J.; Peng, C.W. Multi-agent ontology-based Web 2.0 platform for medical rehabilitation. *Expert Syst. Appl.* **2012**, *39*, 10311–10323.
37. Sirota, P.; Johnson, D.; Ghare, G.D.; Jain, T.; Geller, A.S. Using Configured Application Pricing to Determine End User Fees for Use of Invocable Services. U.S. Patent 7925554, 12 April 2011.

38. Frederick, R.; Kakulapati, G.; Agrawal, A.; McCann, J.R., III. Providing an Invocable Composite Network Service Based on Multiple Other Invocable Constituent Network Services. U.S. Patent 7797198, 2010.
39. Thies, G.; Vossen, G. Web-Oriented Architectures: On the Impact of Web 2.0 on Service-Oriented Architectures. In Proceedings of the 3rd IEEE Asia-Pacific Services Computing Conference, Yilan, Taiwan, 9–12 December 2008.
40. Szepielak, D.; Tumidajewicz, P.; Hagge, L. Integrating Information Systems Using Web Oriented Integration Architecture and RESTful Web Services. In Proceedings of the 6th IEEE World Congress on Services, Miami, FL, USA, 5–10 July 2010.
41. Fielding, R.T.; Taylor, R.N. Principled design of the modern Web architecture. *ACM Trans. Internet Technol.* **2002**, *2*, 115–150.
42. Belimpasakis, P.; Moloney, S. A Platform for proving family oriented RESTful services hosted at home. *IEEE Trans. Consum. Electr.* **2009**, *55*, 690–698.
43. Dodero, J.M.; Ghiglione, E. ReST-based web access to learning design services. *IEEE Trans. Learn. Technol.* **2008**, *1*, 190–195.
44. Su, C.J.; Chiang, C.Y. Enabling successful Collaboration 2.0: A REST-based Web Service and Web 2.0 technology oriented information platform for collaborative product development. *Comput. Ind.* **2012**, *63*, 948–959.
45. JADE. Available online: http://jade.tilab.com/ (accessed on 28 August 2013).
46. OWL-S: Semantic Markup for Web Services. Available online: http://www.w3.org/Submission/OWL-S (accessed on 28 August 2013)
47. W3C, SPIN—Modeling Vocabulary. Available online: http://spinrdf.org/spin.html (accessed on 3 October 2013).
48. Murty, J. *Programming Amazone Web Service*; O'Reilly Media: Sebastopol, CA, USA, 2008.
49. Krishnan, S. *Programming Windows Azure*; O'Reilly Media: Sebastopol, CA, USA, 2010.
50. Uploading Your Application. Google App Engine. Available online: https://developers.google.com/appengine/docs/java/gettingstarted/uploading (accessed on 28 August 2013).
51. Yahoo Weather RSS Feed. Available online: http://developer.yahoo.com/weather (accessed on 28 August 2013).
52. Jena—A Semantic Web Framework for Java. Available online: http://jena.sourceforge.net/index.html (accessed on 28 August 2013).
53. TopBraid Composer. Available online: http://www.topquadrant.com/products/TB_Composer.html (accessed on 28 August 2013).
54. SPIN Support in TopBraid. Available online: http://www.topquadrant.com/products/SPIN.html (accessed on 3 October 2013).
55. Jih, W.; Hsu, J.Y.; Tsai, T. Context-Aware Service Integration for Elderly Care in a Smart Environment. In Proceedings of the AAAI Workshop on Modeling and Retrieval Context, Boston, MA, USA, 16–17 July 2006.
56. Moreno, A.; Sanchez, D.; Isern, D. Security Measures in a Medical Multi-Agent System. In *Artificial Intelligence Research and Development*; Aguiló, I., Valverde, L., Escrig, M.T., Eds.; IOS Press: Amsterdam, Nederland, 2003; pp. 244–255.

57. Burkle, A.; Essendorfer, B.; Hertel, A.; Muller, W.; Wieser, M. A Test Suite for the Evaluation of Mobile Agent Platform Security. In Proceedings of the IEEE/WIC/ACM International Conference on Intelligent Agent Technology, Hong Kong, China, 18–22 December 2006.
58. Shakshuki, E.; Luo, Z.; Gong, J.; Chen, Q. Multi-Agent System for Security Service. In Proceedings of International Conference on Advanced Information Networking and Applications, Fukuoka, Japan, 29–31 March 2004.
59. Wong, S.W.; Ng, K.W. Security Support for Mobile Grid Services Framework. In Proceedings of International Conference on Next Generation Web Services Practices, Seoul, Korea, 25–28 September 2006.