LABORATOIRE INFORMATIQUE DE L'UNIVERSITE DE PAU ET DES PAYS DE L'ADOUR
UNIVERSITE PIERRE ET MARIE CURIE

Towerd a Context-Aware and Automatic Evaluation of Elderly Dependency in Smart Homes and Cities

Kenza Abdelaziz
Pierre et Marie Curie University
abdelaziz.kenza@gmail.com
Philippe Roose , Sebastien Laborie
LIUPPA-T2i, Lab of University of Pau
Anglet,France
(sebastien.laborie,philippe.roose)@univ-pau.fr
Bernd Amann
Lip 6 Lab University Pierre et Marie Curie
Paris,France
bernd.amann@lip6.fr
Tayeb Lemlouma
IRISA LAb University of Rennes 1
Lannion, France
tayeb.lemlouma@irisa.fr

Abstract. In the healthcare domain, the evaluation of elders' daily activities is important to define their dependencies; professionals define manually this information by using the AGGIR grid. With the aging of the word population, it is interesting to automate the dependency's evaluation. In our work, we use the data provided by the different sensors in iCASA simulator and give them semantic. We use the RDF to add semantic to our data. We extend our system by using the ontology and use the SWRL to get more useful information. Our system is based on 3 levels, each level provides information to help us to evaluate the dependency. The first level collects the raw data and represents them in RDF, the second level represents the ambient services, those services use the aggregation of data to get information that help to evaluate the AGGIR services that represents the daily activity of the elder person, those services are represented in the third layer. Having the dependency's value, we can specify the services according to the person's needs.

1 Introduction

In the healthcare domain, the evaluation of elders' daily activities is important to define their dependencies, professionals need this kind of statistics to identify person's needs of assistance, services and allowance, and to take the right decision to host the patient in the health institution, nursing home or let him at his own home with or without healthcare monitoring.

Nowdays , the evaluation of the elder person's dependency is defined manually by professionals. In France for example, experts use the AGGIR's grid (Autonomy Gerontology Iso-Resources Group) [5] based on the activities daily living(ADL) in order to identify dependency of the elder person. This grid is composed of 10 discriminated variables used to classify the dependency of the person in one of the 6 different groups (called GIR), from totally dependent (GIR 1) to totally autonomous(GIR 6).

Recently, the elder persons left alone in home without supervision which cannot help professionals to detect the state changes of patients' abilities to achieve elementary daily activity. The development of smart homes [8] becomes an important way to identify their ability to accomplish daily activities. Indeed, these information can be gathered from the different sensors installed in their houses and then help the system to evaluate the person's dependancy automatically. Having the dependancy's value, the system will be able to specify the services according to the person's needs.

With the aging of the word population, it is interesting to automate the dependency's evaluation. In this work we will use the data given by the different sensors installed inside the smart home, to identify the level of person's dependency. However, it's hard to use this data directly in our system, because of the heterogeneity of sensors and the raw data. To overcome these problems we use the ontology. It gives us semantic and help us to share the information between different services [15] [16].

Many works used ontologies to define the activities of person's, Kim and Choi [9]use ontology to describe the getting up scenario to show whether the model is valuable for description of context information. Chahuara , Portet et al[3] use the SWRL(Semantic Web Rule Language) to recognize the different situation, they use also the Markov Logic Network to take decision, Latfi,Lefebve and Descheneaux [11] present an ontology-based model of the Telehealth Smart Home (TSH) to initialize the Bayesian network to recognize the activity of the patient. All this work aims to recognize the activity of person but none of them try to evaluate the dependency of the person. In this paper we will present our approach to define the dependency of the elder person by using a high level ontology and the SWRL to define the value of person's dependency. The SWRL [7]is a rule language for semantic web, combining OWL language and RuleML (Rule Markup Language (Unary/Binary Datalog). In our work, we use the iCASA simulator. It gives us the ability to create sensors and to simulate a scenario, in order to collect different data that will be used to identify the autonomy of the person along with the rules SWRL.

The next section introduces ontology, section 3, attend to give introduction of AGGIR, the section 4 describes the modeling of the context and the architecture proposed to arrive at the desired results, it contains a scenario aims to see how our system will react to define the dependency of the elder person. In section 5 we will present the implementation and some result for our work, the conclusion and the perspective will be in the section 6.

2 ONTOLOGY

The origin of ontology came from philosophy to express "what exist" [4]. In computer science, the use of the ontology notion is essentially spread in the design and development of knowledge base. According to Gruber[15] [16], ontology is seen as an explicit specification of a conceptualization which contains representation of the knowledge in a domain as a set of concepts and the relationships between them. Ontology can also support reasoning about concepts. W3C recommends OWL as a language to represent ontology, it is based on the RDF[6] that describes information as triples: subject, predicate and object. OWL extends the representation in RDF to define classes, hierarchical relationship between classes or properties, to describe some constraint, binary relation of classes or properties. With this capacity of logical expressiveness leads to deduce new information from the existent ones and the axiom defined implicitly.

An OWL ontology in the abstract syntax contains a sequence of axioms and facts. Axioms may be of various kinds, e.g., subClass axioms and equivalentClass axioms. It is proposed to extend this with rule axioms. that consists of an antecedent (body) and a consequent (head), each of them consists of a set of atoms that have the possibility to be empty. The intended meaning can be read as: whenever the conditions specified in the antecedent hold, then the conditions specified in the consequent must also hold. Whenever the consequent is empty then it must be not hold. We can use SWRL to write rules to deduce other information that will be usufull to our study. For exemple, the following rule

$hasChild(?x,?y), hasChild(?y,?z) -> hasGrandChild(?x,?z)$

means that if x has a child y and y has a child z then the x has a grandchild z. This rule will be applied to all the instance that satisfy the condition to deducing another information implicitly without user interference.

The Semantic Web Rule Language (SWRL) is a proposed language for the Semantic Web that can be used to express rules as well as logic, combining OWL with a subset of the Rule Markup Language. The specification was submitted in May 2004 to the W3C by the National Research Council of Canada, Network Inference, and Stanford University in association with the Joint US/EU ad hoc Agent Markup Language Committee.

In our project, we use the ontology to represent the AGGIR variables as classes, those classes have values. Next section, introduce the AGGIR variables and how they are evaluated.

3 AGGIR GRID AND AGGIR ALGORITHM

AGGIR grid tends to define the dependency profile of an elder person. The dependency is evaluated by 10 different discernment variables. 8 of them are used in the classification of the dependency of the elder person. There are 6 groups of dependancy called iso-resource group (GIR). The algorithm of AGGIR is used to compute the GIR number (1 to 6). In [12], the authors explain how the AGGIR's algorithm computes the value of dependency to identify the appropriate class of the elder person. The algorithm uses 8 classification functions, those functions compute the classification scores. they are defined in the equantion (3.1).The score is calculated in a sequential order of classification from S_1 to S_8.

$$S_i = \sum \underset{k=1}{\overset{8}{w}} \tag{3.1}$$

In table 3.1, we represent the 6 classes (GIR) and the score condition. The function computes the value and compare it with the values in the table. One of the six classes will be assigned in depend of the result obtained by the algorithm.

Each activity of the the elder person is caracterized by four possible adverbs: Spontaneously (S), Completely/Totally (T), Usually (U) and Correctly (C). Spontaneously means that the person exerce the activity without any stimulation, the T has for objective to identify if the person can do the activity completely, which mean all the sub task will be done, the U mean that the activity is done regularly and the C involves the quality of elder person's realization of the activity. Each activity has one of the three modalities: A, B or C. The A evaluation means that

GIR	Score Condition
1	$S_1 \geq 4380$
2	$4140 \leq S_1 < 4380$
	$3390 \leq S_1 < 4140$
	$S_2 \geq 2016$
	$S_3 \geq 1700$
	$1432 \leq S_3 < 1700$
	$S_4 \geq 2400$
3	$S_5 \geq 1200$
	$S_6 \geq 800$
4	$S_7 \geq 650$
	$S_8 \geq 4000$
5	$2000 \leq S_8 < 4000$
6	$2000 \geq S_8$

Table 3.1: Table represents the GIR and the value of S_i.

the person is completely autonomous, B means that the person is partially dependent and C means that the person is dependent and cannot achieve the activity alone. This modularity is evaluated following condition: $S \wedge T \wedge C \wedge U$, if all the adverbs are evaluated by A, the activity will be an A, if the adverbs have value C, the Activity will be evaluated by the modality C, otherwise the activity will have the modality B. After the evaluation of all the variables (or activities), the AGGIR algorithm computes and evaluate the class of the person dependency. The 10 variables are Coherence (how the person behave in the day), Orientation (how the person locate in time and place), Hygiene, Dressing, Eating, Urinary and Fecal Elimination, Transfer (to lie down, to get up and to sit down transfer), Interior displacement, Exterior displacement and Communication (how the individual communicate with the other to alert them).

To evaluate those activities, we must collect data and represent them in a way that supports ontology. Next section tends to explaine how we get our variables from raw data collected from iCASA simulator.

4 The context Model

4.1 Hygiene's scenario

Hygiene is an important point to be considered for the healthcare. For the elder, to maintain the hygiene without intervention of another person is considering as one of the key of the evaluation of the dependency. In AGGIR, hygiene has two other variables, top and down body hygiene. The person identifies as dependent if he can do top and down body hygiene without introducing other person to help him or to remind him to do this activity (S), he does this activity completely (T),

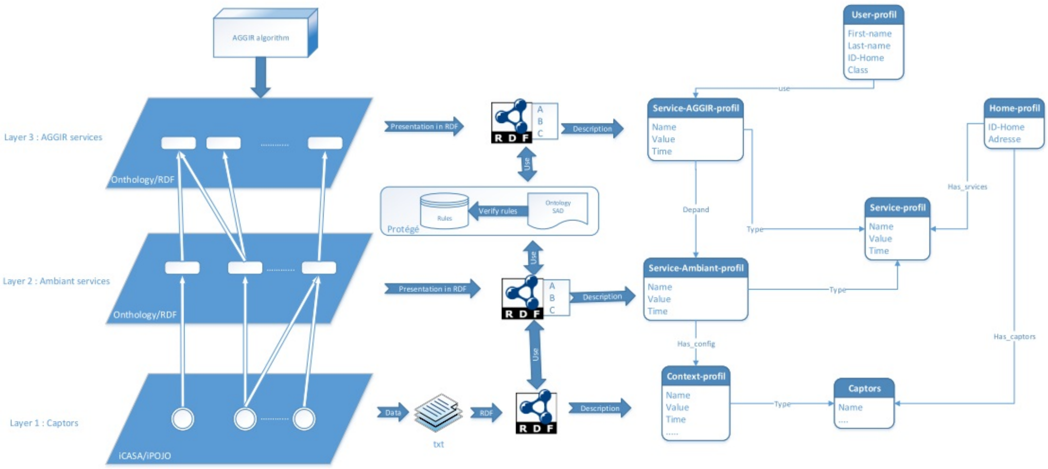

Figure 4.1: Architecture proposed.

correctly (C) and usually (U). If this person does this activity properly he will be evaluated with an A, if all of the criteria are not respected, the evaluation will be a C, else it will take the modality B.

This section presents one scenario to see how our system reacts to evaluate the dependency of the elder person.

SCENARIO : Let "X" be the individual we investigate.In a house, "X" takes bath 28 times in one month without the help of any person. He washes his hands more or less than three times by day.

To evaluate the hygiene dependency of this elder, we have various devices (sensors) that can help us to check if he is taking a bath or not. Among these devices, there are "presence sensor" and "faucet sensor". If we can detect the presence and the use of the faucet inside the zone of the bath, we can deduce that the elder person is taking a bath. It remains to define how many times he does this activity in one month.

4.2 ARCHITECTURE

For this research, we used iCASA [10].The iCASA framework provides required support to build digital home applications. The main goal of digital home applications is to improve the living experience of people in its habitat, reducing the need for interaction with computers. It has been developed in the context of the Medical project by the LIG/ADELE team (Grenoble, France) and the Orange Labs. Although, the group has implemented many sensors in iCASA, but in our research we needed other sensors to continue our study. The ADELE team has worked with us in order to provide us devices according to our request.

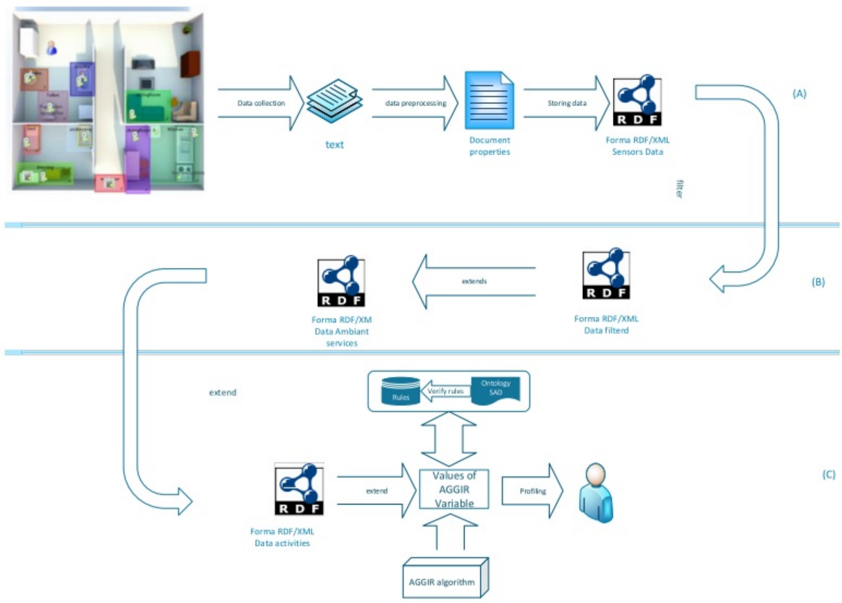

Figure 4.2: This is a figure caption.

The iCASA simulator gives us the possibility to create devices, to simulate a scenario, to develop different services and also to get the information in real time. It is based on the iPOJO model [2], which was developed using the OSGi framework [1] [10]. OSGi is used to develop a dynamic application that reacts in real time.

Figure 4.1 represents our architecture to explain our approach for the evaluation of the elder person's dependency. Our architecture has three layers, each one has its own particularity semantic for the data that it represents. The first layes collectes the data from the iCASA simulator and to represent them with RDF model. The data are represented with triples : subject, property and object, which will give them a meaning. the second layer extends the RDF model from the first level by adding this model more usufull information that will be used to our evaluation.This new information is a conbination of more than one data. the third layer represents our AGGIR variables. In this level we extend the RDF model by using onthology and also by applying our rules and owl axioms.

4.2.1 Layer 1

This level aims at collecting the different data from devices and services developed in iCASA. However, this data are heterogeneous and raw. Figure4.2(A) represents the process from iCASA simulator to the RDF representation of the data. The iCASA simulator provides data that represents the change of the device's state. It will be collected in a text file. However, this data can be duplicated. In iCASA each device has properties. When there are changes in the state of the device,

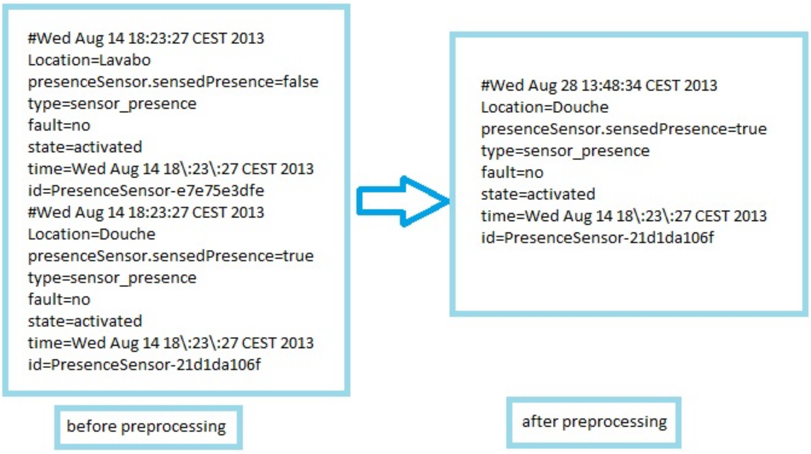

Figure 4.3: This is a figure caption.

each properties change will have a data raw even if they changed at the same time. The "sensor presence" has different properties like "$sensedpresence$" and $Location$, if those two properties had changed in the same time, we will have two raws of data that represents the change for each property; data raw for the change of "$sensedpresence$" and another data raw this one represents the change of$Location$. Therefore, this duplication will have an impact to evaluate the dependency. For example, if we want to evaluate the frequency of presence in a location, the system will search in the file to calculate the number of presence in this location. But if our data are duplicated, the frequency is probably $false$; consequently, our evaluation will be false. To override this problem, we will pre-process the data to get only the useful. Figure4.3,represents raw of data from the file text before the preprocessing, and the file properties after the preprocessing.

Besides, raw data collected from the simulator make it difficult to extend our RDF model and deduce other information that can be used in our evaluation. One possible solution is to get the semantic from this data. Each data represent a value of a property or the name of class. By given each data what it represents, we can get the semantic. The goal of RDF (Resource Description Framework) is to make possible to specify semantics for data and describe resources, using the triple; source, property and value. With this ability, we can describe our sensors as resources, properties as the RDF properties and the value as the value for our RDF properties. In this project, we represent the data collected by RDF model. Figure 4.4 has two parts; the left and the right. the rignt part represents an RDF/XLM including data collected from the simulator. The left part is the representation of the RDF/XML in graph form. the data here describe one presence sensor

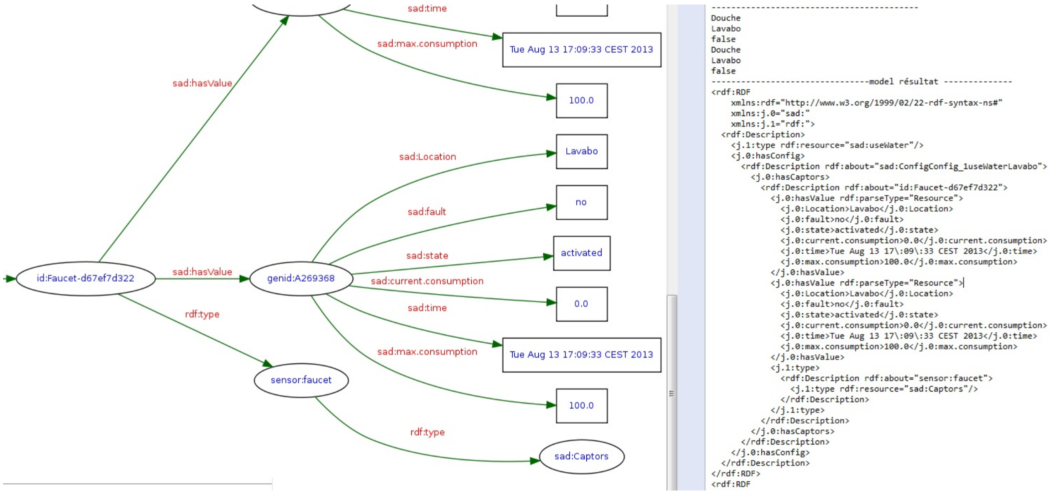

Figure 4.4: Screen shot representing a RDF graph and the XML/RDF for the the data of the faucet sensor.

in one location. Another possible solution is to represent the semantic of data with ontology. Lot of works in research seek to model sensors using a low level ontology[14]. However in this work we will not do this step because we work with a simulator. The sensors in the simulator do not represent the real sensor in the real word. If we need to represent ontology for sensors, we must have all the information about it and some information is missing like the energy level. At this level, we get RDF model of the data gathering from the sensors.

4.2.2 Layer 2

In the simulator, we have "water mater sensor" and "faucet sensor", those two sensors provide the same information: the consumption of water. To evaluate the value of the dependency for hygiene, the system needs this information. If our system uses this information directly from the first layer, it must deal with the complexity of the system and consider the change of the configuration of the sensors. That will not ease the task for a good evaluation of dependency. Layer 2 has two objectives, first, the complexity of the sensors has to be hidden and only the information that can be used will be considered. Second, different information provided by different sensors will be aggregated. These information will be used for the evaluation. This layer will focus beyond the complexity of the system to get information that satisfies the evaluation of the activity, hence the evaluation of the dependency. Those services are called the ambient services. Mark Weiser [17] explains the ambient by when the complexity is disepearing, we are freed to use the iformation without thinking and so focus beyond then on new goals. With the addition of semantic for the data collected, we will improve the data coming from these services, because the system will focus on the meaning of the data to improve the data that will be provided. Figure 4.2 B represents the workflow of data from the data provided from the first layer two the aggregated data from this layer. Layer 2 can be seen as an intermediate between the first layer, which uses to get the raw data, and the third layer, whose goal is to evaluate the daily activities. When the third layer use the information given by this level, it will not consider the complexity of the sensors or configuration, but only the data provided by this layer. For example to evaluate the hygiene of an elder, we must know his presence in the shower's location and the Consumption of the water in this location. In this layer we will have two services "$Presence$" and "$UseWater$" to help us for the evaluation of the hygiene. The first service provides the frequency of the presence in a location. The second service has for objective to get the sum of the water consumption. Each of these services will provide information by using the presence sensor and faucet sensor that satisfy the objective.

The "$UseWater$" service has different configurations. As we said before, we can calculate the consumption of water from different type of sensors,if our system is built only with two layers, the AGGIR services must deal with this problem. Indeed, each house has their type of sensors, and therefore their configuration, built an architecture based only whith two levels will increase the risk of errors and a bad evaluation. Any change in sensors or configuration will cause a new implementation of our system, or at least the change of AGGIR services using those sensors. That is not desirable for the stability of the system. Having this layer will almost stabilize the system, any change in the sensors type or configuration provokes only the change of the ambiant services responsable for those sensors and not for the AGGIR services. This layer reduces the problems attached with the nature and differences of sensors, to help the system to evaluate eache activity without considering the complexity of sensors or their configuration. Another advantage to have this level is the reuse of the same ambient service by different AGGIR service. The "$Presence$" service, For example, is used by "$Hygiene$" service and "$InteriorDisplacement$".

4.2.3 Layer 3

The objective of this project is to get the dependency of an elder person to specify the appropriate services, that means, we must share information between services. The ontology is represented by classes, properties and values, by linking two classes in different domains for exemple, the services like nursing and allowance represent two different domains , but they can have the same elder persons then we can share information between those services only by linking the person's class in nursing domain with the class of the same person in allowance damain.Figure 4.2 C represent the process from the data provided by the ambient services to the class of the dependency of the elder person.

This level will evaluate implicitly the dependency of the individual with the use of the ontology to get the semantic for our data, axioms ; that can deduce other information and help to get the dependency's value of each activity. That

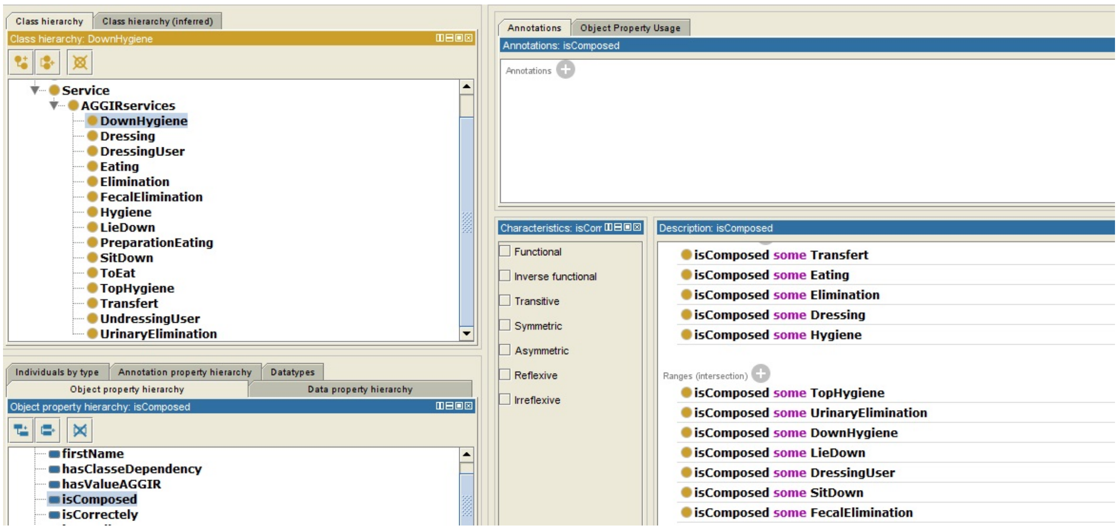

Figure 4.5: The different level and data.

can be done by applying rules on the data from the level 2. At this level, we get the AGGIR variable's value and By applying the AGGIR algorithm, we will compute the dependency value and get the elder person dependency's class. by having this information we will get the dependency profile and therefore to specify services for him. That can be done by the use of the ontology. this one will share knowledge between different services. In fact, ontology describes any domain by classes, properties and values. if we want to share knowledge between diffrent domains like nursing and allowance, we link two classes wich represent the same class person by specifying the equivalence between those two.

The objective of this project is to get the dependency of an elder person to specify the appropriate services, that means, we must share information between services. Figure 4.2 C represent the process from the data provided by the ambient services to the class of the dependency of the elder person.

4.3 Modeling of context information

The context information describes our services in classes, properties and values. We distingue two classes: "*AGGIRServices*" and "*AmbientServices*". The first class represents the AGGIR variables and has sub classes such us "*Coherence*", "*Hygiene*". The second class represents the different ambient services used to evaluate the AGGIR variable, it has sub classes like "*UseWater*", "*Presence*". This distinction between those two classes "*AGGIRServices*" and "*AmbientServices*" is due to the information given by the different sub classes and the use of data provide by those services. Take for example "*Hygiene*" and "*UseWater*", those two services represent two different classes the first one aims to evaluate the dependancy for the hygiene activity. The properties for this class have values : A, B or

C (The AGGIR modalities). Contrarily, The "$UseWater$" tends to give information that is used for the evaluation of the hygiene activity. Properties in this class have digital values. The information given by the ambient services do not represent the value of the dependency but the aggregation of data sensors. For example, "$UseWater$" tends to give us the total of the water consumption but not to give us the dependency's value of any activity.

Figure4.5 is screenshot represents our ontology with Protégé. we can notice that screen is divided in four parts; two in the left side, and two in the rignt side. The upper left part represents the different classes, down left, the representation of different properties. the down rignt part represents the domain and the rang of the property. Figur 4.6 represets a part of graph that describes the activity hygiene. We can see that hygiene is composed by "$TopHygiene$" and "$DownHygiene$". in fact, some AGGIR services are composed in more than AGGIR services.

To evaluate any activity of the elder person, we need to evaluate the four adverbs: Totally T, Usually U, Correctly C and Spontaneously S. For example, if we need to evaluate usually, we calculate the frequency of the activity and compare it with a threshold (this one is given by us). In our ontology, the activities are represented by classes in "$AGGIRService$", those classes have properties such us "$isUsually$" which represents the adverb Usually. If we need to evaluate any "$AGGIRService$" we must give the properties such us "$isUsually$" a value. To evaluate "$TopHygiene$" for exemple, we need to give each property that represents the AGGIR adverb a value. In our system, the property "$isUsually$" is evaluate by using the frequency of presence in location sink. This frequency is given by the "$AmbientServices$" ; "$Presence$". We compare the frequency given (fg) and the threshold (tf). If the $tf \leq fg$, then the Property "$isUsually$" will have the value C, Otherwise, it will have the value A. After the evaluation of all the properties representing the different adverbs, we use SWRL rules to get the value of "$TopHygiene$". This process will be applied to evaluate the other "$AGGIRService$".

Some "$AGGIRService$" are not evaluated directly like "$Hygiene$" because they are composed by more than one "$AGGIRService$". For example,In figure 4.6 we can see that "$Hygiene$" is composed by "$TopHygiene$" and "$DownHygiene$". To evaluate "$Hygiene$", we must evaluate "$TopHygiene$" and "$DownHygiene$", if the two last services have the same modality mod ($mod \in \{A, B, C\}$), the hygiene will have the modality mod, otherwise it will be the modality B.

4.4 Context reasoning

The ability of logical expressiveness given by the ontology will help us to define the dependency of the person implicitly by using rules. The rule ($S \wedge T \wedge C \wedge U$) which can be easily expressed by SWRL, can help the system to implicitly deduce the dependency of any activity, only by the evaluation of all the adverbs. The system implies the rule automatically and gives us a value (A, B or C). The ability

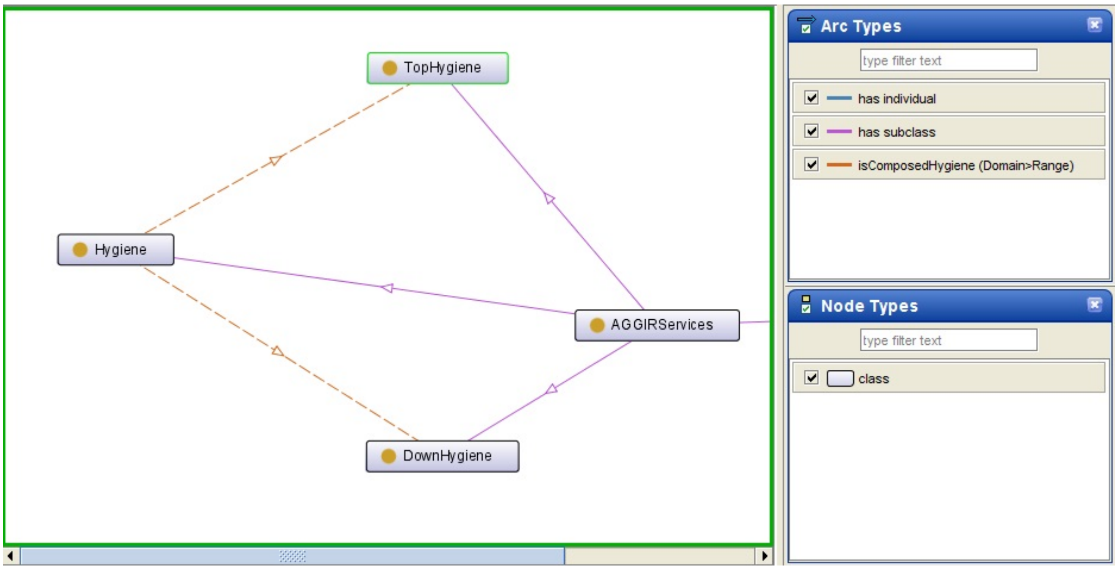

Figure 4.6: The different level and data.

of logical expressiveness will give our system a logical reasoning and help it to give the right information. Those are not the only objectives to use the rules. Another use of them is to link two different domains. For example, if we need to link information of person such us first and last name to another domain such us allowance, we can write rule that link the class of person existing in the two domain, by telling the system that those two classes are equivalent. The logic is a powerful tool for reasoning, taking decision and to give other information that can be used. Rules in ontology will be applied to properties, classes and the relation between the properties.

5 Implementation

Our research aims to evaluate the dependency of an elder person. To achieve this, we proposed architecture. Each level of this architecture use different technology. The challenge is to bring those technologies so that they work together. The use of RDF and ontology will help us to fill this gap between the different levels because we will work with the semantic of data not the data itself. For example, when we extend our information in the "$AmbiantService$", we do not use the raw data but the RDF data. We used two projects for our work. The first project will gather the data given by iCASA's simulator. The second project will work with Jena's API and Protégé to get the RDF model and the extension OWL. We use SWRL to evaluate the activities.

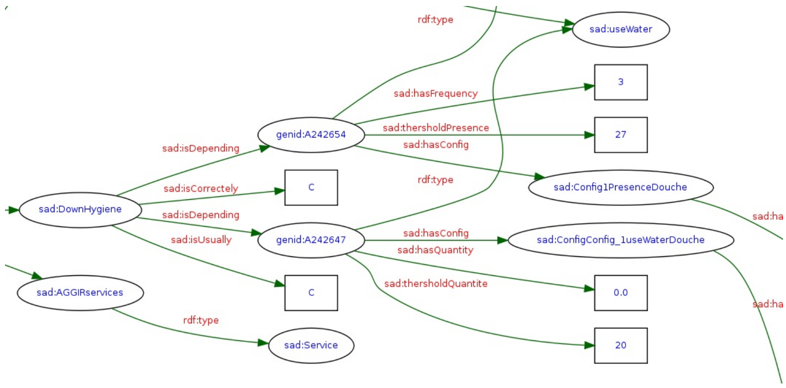

Figure 5.1: Screen shot representing a RDF graph.

FIRST PROJECT : iCASA simulator is based on iPOJO model, which developed by using OSGi framework. It resembling an application server built for dynamic loading and management of software components (called service bundles). These bundles can be dynamically instantiated by the framework to implement of specific services. The components can be remotely installed, started, stopped, updated, and uninstalled without requiring a reboot. The iPOJO is model natively support all the dynamism of OSGi. With iCASA that based on iPOJO, we will have all the benefit of OSGi and simplicity of iPOJO. For our project we used the iCASA to gather data that represent the state changement of our sensors. The data are stored in a file text, containing all the data that can be used by us in order to evaluate the dependency.

SECOND PROJECT : The second project has different objectives. The first objective is to preprocess the data given by the firs project. We use for this a text parser. The second objective is to create the model RDF, that represents our data, for this we used the Jena's API [13]. Jena is API in the Java programming language, for the creation and the manipulation of RDF graph. It has a query engine ARQ that supports the SPARQL, RDF Query language. Figure 5.1is a screen shot represents our RDF graph; this graph is validated using the RDF validator. Jena also allows a range of inference reasoners to be plugged in it. By using this reasoner, the ontology information, the axioms and rules associated with the reasoner we will be able to get additional facts for the instance data and class description.

In our scenario presented in the previous section, we aim to evaluate the hygiene dependency of the person. We will present next how our system reacts to evaluate this activity. First, we create a bundle in iCASA that detect the state changement of "presence sensors" and "faucet sensors". we filter our data to work only with the sensors in "shower" loaction. Now we have the data presenting the presence in the "shower" location and the use of the faucet in this location.

Two classes are created to get the information of those two sensors. Each class implements a listener to detect the change regarding the state of any properties. The information will be gathering in file that has the same name of the type of sensor. For example, the "presence sensor" data will be gathered in the file "sensorpresence". at this stage we used the first project. Then we used the second project to get the RDF model of our data. like we monsionned before, the second project has two objectives ; the preprocessing, which remove the duplicated data and the construction of the XML/RDF model for the data of sensors.

Up to now, we have only represented the data given by our sensors, but the goal of our study is to evaluate the dependency of the person using this information. to do this we extend our model to get the "$AmbiantServices$" and their value. For that we use SPARQL to get our information. A simple request will help us to get the information that we need. For example, to get the frequency of presence in location shower, we can write a request 1.1 using SPARQL and specifying only that the sensor has type "$sensor\_presence$". then we specify that "$sensedPresence$" = "$true$" and the "$Location$" = "$shower$". The system will search for the sensors that satisfy our request.

$$\begin{array}{ll} select\,?z where & \\ \{?x < sad : hasValue > ?z. & \\ ?x < rdf : type >< sensor : sensor\_presence > . & \\ ?z < sad : Location > ?w. & Request(1.1) \\ ?z < sad : presenceSensor.sensedPresence > ?a & \\ filter(?w = "Location" \,\&\&?a = "true").\} & \end{array}$$

Now, we calculate the line's number of response that satisfies our request to get the frequency of presence. With this information, we can extend our graph by adding this information to our "$AmbiantServices$"; "$Presence$". At this step we are in the level 2 of our architecture. We need now to get our modalities for our "$AGGIRServices$". We use SPARQL again to get the information about the frequency and the threshold of presence. this request in SPARQL aims to give us the threshold of $Presence$.

$$\begin{array}{l} select?zwhere \\ \{< sad : Presence >< sad : isDepending > ?x. \\ ?x < sad : thershold" + serviceAmbiant + " > ?z.\} \end{array}$$

With this two informations, we can evaluate the property of "$isUsualy$" by given it one of the modalities (A,B or C). After the evaluation of all the adverbs, we apply the this SWRL rule to evaluate our "$TopHygiene$" service:

$$TopHygiene(?x), isCorrectely(?x,?val), isCompletly(?x,?val),$$
$$isSpontanoesly(?x,?val), isUsually(?x,?val) -> hasValueAGGIR(?x,?val)$$

After the evaluation of "$TopHygiene$" and "$DownHygiene$" services, we can evaluate "$Hygiene$" by using this rule :

$Hygiene(?x), isComposed(?x,?y), isCompsed(?x,?z), hasValueAGGIR(?y,?val),$ $hasValueAGGIR(?z,?val) -> hasValueAGGIR(?x,?val)$

By the evaluation of all the services, we can apply the AGGIR algorithm to get the dependency of the person. By knowing his dependency, we can define his profile and we specify his need in assistance or not. If the person needs assistance, we can share this information with the responsible of assistance service, by specifying the equivalence between two classes of the same person

6 CONCLUSION AND PERSPECTIVE

In this work, we evaluate the dependency of elder person by using the data provided from iCASA simulator. Those data are raw and the sensors are heterogeneous, that is why we propose to give data a semantic by representing them in RDF. This representation brings more information about the data and sensors. The use of RDF will facilitate the extension to the ontology; this one will give us the ability to use SWRL to get more information explicitly. To achieve our goal; the evaluation of the elder person, we propose architecture with 3 layers, each layer has his own objective. The first layer aims to get the raw data and to represent them in RDF form. The second layer uses the aggregation of the data from the first level to give us other information that will be used in the third layer. The third layer, which is the AGGIR services is used to represent the AGGIR variable, those variable are evaluated by 4 adverbs that give us one of the modality A, B or C. By getting the modalities for each variable, we compute the dependency's value of the elder person with the AGGIR algorithm to get his class.

Having architecture with 3 layers improves the evaluation of the elder person's dependency, the second layer, which represents the ambient services, has more than one roles, the first role is to represent the aggregation of data and to give them semantic by representing them in RDF form, the second role is to separate the sensor's complexity from the dependency's evaluation, with this separation we will be able to modify any sensor or it is configuration without the need of changing all the system.

This work need more research and tests to evaluate the dependency efficiently. We have proposed architecture and used the ontology to help us to get more information. Using the SWRL adds to our system a logical power. But Protégé limits this power, indeed some rules are not expressed by SWRL, this problem is related with limit of the predicate logic that cannot express the fuzzy expressions, possibility, necessity and temporal logic. Other limits of ontology must be discussed is confidential level, the evaluation of the dependency of the elder person does not concern all the domains and the information is confidential because it concerns the personâĂŹs life. Those problems must be considered in the future to get a powerful system, which can evaluate the dependency and share the information in confidential way.

REFERENCES

[1] João Claudio Américo, Walter Rudametkin, and Didier Donsez. Managing the dynamism of the osgi service platform in real-time java applications. In *Proceedings of the 27th Annual ACM Symposium on Applied Computing*, SAC '12, pages 1115–1122, New York, NY, USA, 2012. ACM.

[2] Thomas Calmant, Joao Claudio Americo, Olivier Gattaz, Didier Donsez, and Kiev Gama. A dynamic and service-oriented component model for python long-lived applications. In *Proceedings of the 15th ACM SIGSOFT symposium on Component Based Software Engineering*, CBSE '12, pages 35–40, New York, NY, USA, 2012. ACM.

[3] Pedro Chahuara, Francois Portet, and Michel Vacher. Context aware decision system in a smart home : knowledge representation and decision making using uncertain contextual information.

[4] B. Chandrasekaran, J.R. Josephson, and V.R. Benjamins. What are ontologies, and why do we need them? *Intelligent Systems and their Applications, IEEE*, 14(1):20–26, 1999.

[5] SSyndicat National de Gerontologie Clinique. Aggir guide pratique pour la codification des variables. principaux profils des groupes iso-ressources. In *Embedded Software and Systems, 2009. ICESS '09. International Conference on*, pages 249–259, 1994.

[6] S. Decker, S. Melnik, F. van Harmelen, D. Fensel, M. Klein, J. Broekstra, M. Erdmann, and I. Horrocks. The semantic web: the roles of xml and rdf. *Internet Computing, IEEE*, 4(5):63–73, 2000.

[7] Peter F. Patel-Schneider Ian Horrocks, Benjamin Grosof Said Tabet, Macgregor, and Mike Dean. Swrl: A semantic web rule language combining owl and ruleml. 21 May 2004.

[8] Li Jiang, Da you Liu, and Bo Yang. Smart home research. In *Machine Learning and Cybernetics, 2004. Proceedings of 2004 International Conference on*, volume 2, pages 659–663 vol.2, 2004.

[9] Eunhoe Kim and Jaeyoung Choi. An ontology-based context model in a smart home. In *Computational Science and Its Applications - ICCSA 2006*, volume 3983, pages 11–20, 2006.

[10] Philippe Lalanda, Julie A. McCann, and Ada Diaconescu. In *Autonomic Computing : Principles, Design and Implementation*, pages 279–284, 2013.

[11] Fatiha Latfi, Bernard Lefeve, and céline Descheneaux. Ontology-based management of the telehealth smart home, dedicated to elderly in loss of cognitive autonomy.

[12] Tayeb Lemlouma, Sebastien Laborie, and Philippe Roose. Toward a context-aware and automatic evaluation of elderly dependency in smart homes and cities. In *World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2013 IEEE 14th International Symposium and Workshops on a*, pages 1–6, 2013.

[13] B. McBride. Jena: a semantic web toolkit. *Internet Computing, IEEE*, 6(6):55–59, 2002.

[14] E. Meshkova, J. Riihijarvi, P. Mahonen, and C. Kavadias. Modeling the home environment using ontology with applications in software configuration management. In *Telecommunications, 2008. ICT 2008. International Conference on*, pages 1–6, 2008.

[15] Thomas R.Gruger. A translation approach to portable onthology specification. April 1993.

[16] Thomas R.Gruger. Toward principales for the design of ontologies used for knowledge sharing. August 1993.

[17] M. Weiser. The computer for the 21st century. *Pervasive Computing, IEEE*, 1(1):19–25, 2002.