International Journal of Semantic Computing
Vol. 6, No. 1 (2012) 111–130

DOI: 10.1142/S1793351X1250002X

ONTOTV: AN ONTOLOGY-BASED SYSTEM FOR THE MANAGEMENT OF INFORMATION ABOUT TELEVISION CONTENTS

JOSÉ LUIS REDONDO-GARCÍA*

Computer Science Department, University of Extremadura
Campus Universitario S/N, Cáceres, 10071, Spain
jluisred@unex.es

ADOLFO LOZANO-TELLO

Quercus Software Engineering Group
Escuela Politécnica, Cáceres, 10071, Spain
alozano@unex.es

Nowadays, there are a huge number of digital television platforms and channels, so it is not easy for the viewer to decide what they want to watch. Some television providers offer information about the programs they broadcast, but this information is usually scarce and there is no possibility to perform advanced operations like recommendation ones. For this reason, viewers could benefit from a system that integrates all the available information about contents, and applies semantics methodologies in order to provide a better television watching experience.

The main objective of this research is the design of a television content management system, called OntoTV, which retrieves television content information from various existing sources and represents all these data in the best way possible by using knowledge engineering and ontologies. These semantic computing techniques make it possible to offer the viewers more useful operations on the stored data than traditional systems do, and with a high degree of personalization. Additionally, OntoTV accomplishes all of this regardless of the TV platform installed and the client device used. The viewers' satisfaction when using this system has been also studied to prove its functionality.

Keywords: Digital television; information management; data collection; ontology; linked data.

1. Introduction

The audiovisual sector, particularly television, occupies a prominent place in the information society. The number of digital TV platforms that offer lots of content to their users is growing every day. However, this wide variety of television contents makes it more difficult for information to be correctly structured and easily accessed. For example, when a user looks for new programs to watch, they have to consult all the available platforms one by one: they review the DTT programming guide, the cable operator's information, the channel list of their favorite website, etc. Another

*Ronda del Carman, N° 9, 4° A, 10002, Cáceres, Extremadura, Spain.

problem is that information is usually scattered and incomplete. Imagine that one of the available channels offers the TV series "The Walking Dead". The user will usually find a very brief description about it, but if he wants more information, he has to resort to alternative sources like Internet searches. Also, there are advanced functionalities like custom searches and content recommendations, which day after day are becoming more familiar to the user. They have already been implemented in some Internet scenarios like YouTube, but they still have not taken the leap to television industry.

In short, a new system for managing television content information is needed. This system needs to offer a more universal and easy-to-use service that integrates various available television data sources. When information is scarce, it will access external resources to gather it in a timely and transparent fashion. This way it provides a common information access point for viewers, regardless of whether they use a mobile device, a personal computer, or a decoder in their living room. From a technical point of view, the implementation of this kind of solution is perfectly possible. Actually, there are various digital TV systems with advanced capabilities in application development and interactivity, such as MHP (Multimedia Home Platform) [1] or Google TV (http://www.google.com/tv/). For this reason, they can incorporate an advanced content information management system, which also would give them an added value.

Specifically, in this research, ontologies have been used in order to facilitate the representation of the collected content information. This shrinks the semantic gap in search and recommendation operations, making them as effective as possible. In the areas of multimedia and digital television, there are already some examples of systems that also use ontologies. The project AVATAR [2] incorporates ontologies in OWL format, focusing primarily on the storage of user preferences. It uses semantic reasoning techniques to perform more accurate recommendation operations. NoTube [3] is a television content management system that is able to access a huge number of data sources: electronic guides for various digital platforms, users' preferences based on their behavior on sites like Twitter and Facebook, Semantic Web information, etc. Also, the Sensee system [4] integrates different television resources, such as programming from the BBC, XMLTV guides, and descriptions of films from the IMDB website.

The above systems have significants capacities when using semantics in television, however they lack some desirable features. On the one hand, AVATAR does not elaborate on the way television content descriptions are retrieved. It simply assumes that providers offer all the necessary information about the programs. But the reality is quite different today: users demand more and more complete and fine-grained descriptions about whatever they watch. So instead of this, the system proposed in this research provide mechanisms that integrate data from various existing sources in order to obtain extra information. On the other hand, NoTube and Sensee systems access as many data sets as possible and then they interlink this collected information with the more adequate audiovisual fragments in the television contents. However, they are not focused on the development of advanced algorithms over an ontology model, so they do not take advantage of the implicit semantic information that can be represented in these formal structures. The system proposed aims to overcome this issue by implementing different ontology inference algorithms that improve searches and recommendations operations. This way it is possible to offer the viewer a more complete television system, with the most suitable features, which can be deployed in a real scenario.

2. OntoTV System

The solution proposed here for the television data management problem is the design of the OntoTV information system (ONTOlogy-based System for Digital Television). This system collects relevant television information, like program descriptions and broadcast schedules, from various existing sources and represents it by using ontologies. Viewers can access this data and perform personalized operations, regardless of the device used. This way, a complete and more attractive solution for the management of digital television information is presented.

2.1. *System architecture*

The system has been developed according to a modular and centralized design. Some modules collect the data, others process the information, and the remaining ones show the results. This kind of design allows the system to be scalable and flexible to future changes, and to introduce new modules when necessary. Figure 1 shows that OntoTV implements a client-server architecture, just like many interactive services offered today:

- The area at the top represents the server side. At the top left side of the figure, the OntoTV-InfoCollector module collects information from different content description sources. The collected data may be incomplete or poorly structured, so this module performs additional processes that enrich the data and gives it uniformity. Then, the module OntoTV-ParserOWL stores this information in OntoTV-Data, where the ontologies are maintained. The viewer's preferences collected by OntoTV-InferredRules are also present in these ontologies. OntoTV-Core executes search and recommendation operations, and OntoTV-UserSession supports the communication between the client and the system.
- The area at the bottom corresponds to OntoTV-Client module, which runs on the client's television receiver. It implements all the functions related to data exchange with the server, as well as the development of graphical interfaces.

With these design decisions, OntoTV becomes a more extensible and powerful information management system. It offers better features than those implemented inconventional decoders, so it will be more attractive to viewers who want to enjoy multimedia contents in a much more effective way.

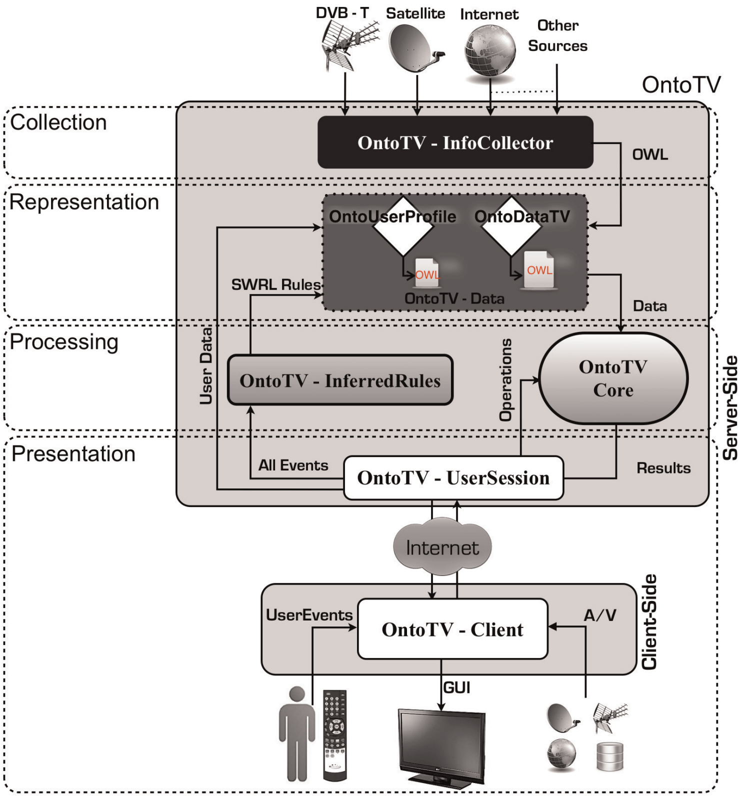

Fig. 1. General Schema of the OntoTV System.

3. Data Collection

OntoTV system accesses information about programs and television channels available in several sources, including Linked Data resources. Then, the system pre-processes this data and includes it in the knowledge base so it is available for performing the semantic processes.

3.1. *Standards for information exchange*

OntoTV requires a well-established format able to support the process of extracting information from the sources and integrating this data into a common ontological model. TV-Anytime (http://tech.ebu.ch/tvanytime) has been chosen as the most suitable option for these collection purposes. The reasons are its high capacity for describing contents and user preferences and the fact that it is clearly focused to the television area. Also, it has been developed as a result of the extensive work made by some of the most relevant companies in the area of television and audiovisual content: Sony, BBC, etc. This increases compatibility with other television systems and makes easier the process of adding new features.

However, it is necessary to point out other alternatives for multimedia content description that has been considered but finally rejected. One of them is MPEG7 [5], which is an XML-based standard developed to describe the attributes, the semantics and the structure of any kind of multimedia content. MPEG7 is not appropriate for OntoTV because it includes various low-level descriptors that are considered not relevant in television scenarios. For example, descriptors for colors and textures like "Color-Structure"(a color histogram) and "Texture Browsing" (for categorizing different types of image textures) are useless and too much abstract from the user's viewpoint. Also, these features would require fine-grained queries that rely on very complicated user interfaces for a television screen. What TV-anytime does is to import only some of those MPEG7's parts that can be suitable for television. For example, for modeling users' profiles TV-Anytime reuses the "User Interaction" element, which makes it possible to represent how the broadcasted information is accessed and consumed.

Also, another considered format is XMLTV (http://xmltv.cvs.sourceforge.net/). Its advantages are that it is specially created for storing electronic program guides, and that it is supported by various television applications like MythTV (www.mythtv.org/) or SageTV (www.sage.tv/). However, its structure is too simple to fulfill the information requirements of the operations described in Sec. 5.2. Actually, XMLTV only considers representing two kinds of entities: channels and programs. Channels have four attributes ("display-name", "icon", "url", "id"). Programs have more than thirty, but most of them ("language", "length", "country", etc.) are very poor detailed. About the user preferences, they are not taken into account so it is not possible to perform personalization tasks. These factors have led to the rejection of this format as a valid alternative.

3.2. *OntoTV-infoCollector module*

The OntoTV-InfoCollector module accesses external data sources to collect information about television contents. Figure 2 shows a more detailed diagram of it.

There are several components inside of it, which will be described in the next subsections. They work together in order to retrieve all the possible information about the contents that are broadcasted on the digital platforms supported in the system.

3.2.1. *Read-type components*

The "Read-type" components extract data directly from the digital TV platforms supported by the system. For example, OntoTV has specific hardware for receiving DTT (Digital Terrestrial Television). It has a component called "DTT-Read-Component" that implements a mechanism for accessing information contained in DTT stream using DVB-SI and DVB-EIT, as described in [6]. This information is translated into TV-Anytime format.

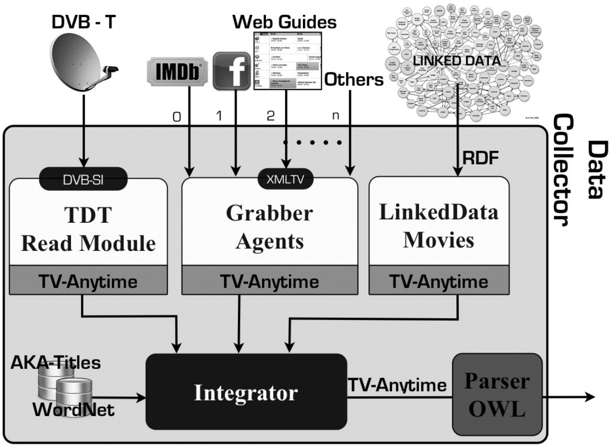

Fig. 2. OntoTV-InfoCollector module.

3.2.2. *GrabberAgents components*

The information sent by the providers is sometimes too short and incomplete. For this reason, OntoTV accesses alternative Internet programming guides for enriching the previously collected data. The "GrabberAgents" components are a set of agents that connect to external servers using the TCP/IP protocol, in order to retrieve the information stored in them. Some of these resources will be in the results section.

Furthermore, because the TV-Anytime standard allows describing viewer's preferences, it is possible to add new agents that collect information from social networks like Twitter or Facebook, where extensive data about user profiles is present [7]. This way, more precise description about users could be created.

3.2.3. *Linked Data components*

Next component enriches movie descriptions by accessing Linked Data resources. The guides collected by the other components may have some empty attributes, so additional enriching processes should be performed. This component accesses Linked Data Resources, identifies information about films, and completes the movie descriptions of the TV-Anytime programming guide. In this research, the semantic information mashup called SIG.MA (http://sig.ma/) has been used. The advantages are clear: it is possible to access relevant information from various datasets without performing a slow crawling process, so this component is relieved of intensive operation. This approach also avoids spelling mistakes and even recognizes language alternative movie titles.

The basic mechanism for accessing Linked Data on the Web is to dereference HTTP URIs into RDF. In the case of SIG.MA mashup, it is necessary to make

```
PREFIX sigma: <http://sig.ma/property/>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
SELECT ?director ?name
WHERE {
    ?film  sigma:director ?director.
    ?director rdfs:label ?name.
}
```

Fig. 3. SPARQL code for retrieving data about movies from SIG.MA mashup.

requests with the URL http://sig.ma/search?q=moviename, where "moviename" is a string that represents the movie's title. When the RDF file that describes the movie is already available, data can be extracted from it by using SPARQL language as seen in Fig. 3.

There are other movie items that can be collected by using the same strategy described before. For example, "language" item can be retrieved by querying <sigma:language> property, the movie's length can be obtained by looking for <sigma:runtime> property, etc.

3.2.4. *The integrator component*

The next component is called Integrator and includes in a single data file all the TV-Anytime information collected by the three previous components. It takes all the descriptions that belong to the same content and creates a single programming guide in TV-Anytime, using similarity criteria described in [8]. Once the information is properly formatted and stored in a single TV-Anytime document, it is added to the knowledge base. The ParserOWL component translates the generated TV-Anytime file into the OWL language used in the system's ontologies. OntoTV uses the OWL API for such purposes.

4. Representation: Using Ontologies

The digital television domain is composed of a large number of programs, series, movies, etc. with their corresponding attributes. All of them are very suitable to be represented using ontologies. For this reason, OntoTV system uses an ontological model for the storage and representation of the information.

Ontologies play a crucial role informally describing a domain and generating new knowledge by using deduction processes. They have numerous advantages over the traditional techniques of information representation [9]. For example, conventional systems are often too rigid when categorizing a particular content, and they tend to produce predictable results when performing complex operations such as content recommendations. But ontologies make it possible to create a more suitable and extensible data model, implement advanced ways of representing user preferences, and support production rules that improve the recommendation processes.

4.1. Ontology management: OntoTV-Data module

This module performs reading and writing operations on the instances of the two ontologies. One of them stores television content descriptions and the other represents the viewers' profiles. This module includes Jess [10], an inference engine that works with the production rules stored in the viewer ontology. These rules are written in the SWRL language and describe users' habits and preferences.

4.1.1. *Television content ontology: OntoDataTV*

This ontology stores all the information related to television contents. It is based on the above-mentioned project AVATAR. OntoDataTV includes a hierarchy of programs with a main class called "TVPrograms", as well as some subclasses that represent program categories (sports, movies, series...). Each content occurrence of a particular date, time and channel is represented as an instance of these classes. These instances of a certain television program can also be related to other classes in the hierarchy of genres by using the property "alsoSubclassOf". This way, it is possible to support a more sophisticated way of content categorization, like, for example "comedy sport series" or "love musical films". Other characteristics of the proposed model are listed below:

- It has enough data properties for storing the instances with a high degree of detail.
- It has been built based on the elements of TV-Anytime standard, so the process of adding the previously collected information to the knowledge base is easier than normal.
- It has a wide variety of television content categories, approximately 21 different ones.
- It also includes another important classes in the television field like actors, awards, directors, etc.

4.1.2. *Viewer's ontology: OntoUserProfile*

OntoTV stores information about viewers, especially their preferences about television contents, to perform custom searches and recommendations. It uses an ontology that is divided into two parts.

The first part is static. It is composed of a set of classes and properties that contain users' information, following the model proposed in GUMO (General User Model Ontology) [11], and more specifically the branch "BasicUserDimensions". However, this is still a very wide part of the ontology, so elements that are not needed in the TV domain have been eliminated. For example, classes under category "Emotional State", "Physical State" and "Facial Expression" are too much specific and variable over the time, so they require too much efforts to be updated. Also, subclasses of "Nutrition", "Role", "Motion", and "Ability and Proficiency" are not relevant to OntoTV system. Only the entities under "Contact Information", and "Personality" have been maintained for storing the user' personal data, as well as

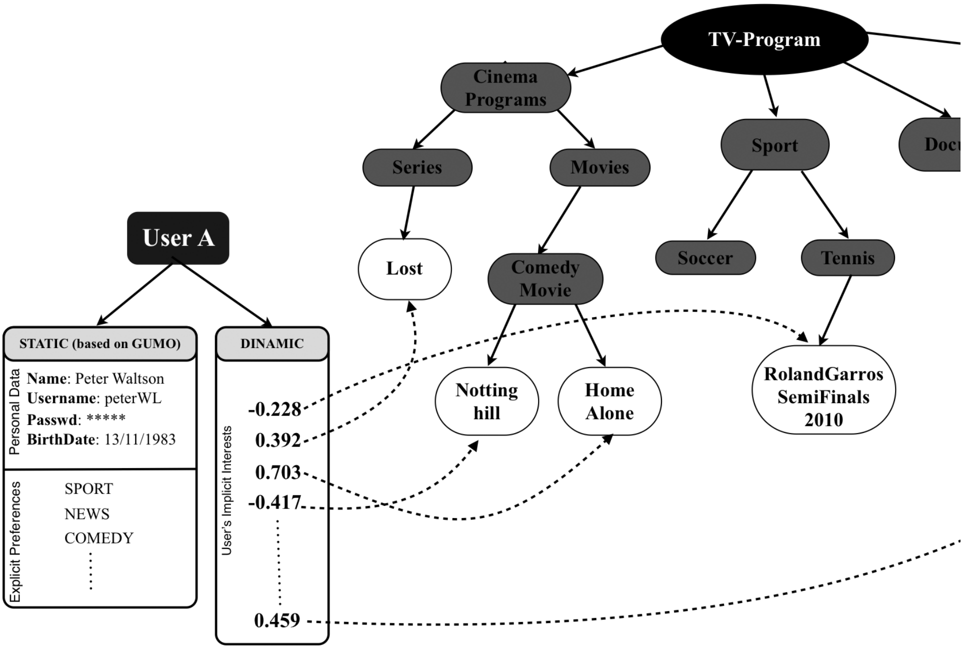

Fig. 4. The OntoUserProfile ontology and the references to OntoDataTV ontology.

some general aspects of their personality (calm, enthusiastic, optimistic, pessimistic, etc.). Also, apart from the GUMO's classes, some extra properties has been added for storing graphic interface customizations and explicit television content preferences like "preferred movie genre" or "favorite sport category".

However, preferences are more accurately reflected in the second part of the ontology, which is dynamic. Every time the viewer rates contents, or new preferences are inferred based on their behavior, a new pair of the type [Reference, Value] is added in it, as shown in Fig. 4. More closely, these pairs are formed by:

- A reference that points to an instance or class in the content ontology. By referencing original elements in OntoDataTV, it is possible to immediately access all the elements available attributes, and to ascend or descend the hierarchy while gathering extra information that improves the accuracy of the operations. This approach also saves memory by avoiding duplicate data.
- A numeric value that represents the user's affinity to the content. It is a real number, normalized in the range $[-1, 1]$. As in AVATAR [2], values close to -1 denote rejection by the viewer and those close to 1 denote high affinity. This value changes according to the viewer's behavior:

  (a) If he accepts watching a recommended content, the rating is increased in a particular ratio that depends on the time he has spent making this decision (Ac, Acceptance) and the opposite occurs when he rejects it.

(b) Another parameter that modifies this affinity value is the percentage of the television content that has been consumed (Cp, Consumed percentage).
(c) If there are explicit preferences in the static user profile, and these preferences match with the current content, the index have to be increased (Ep, Explicit Preferences).
(d) It also depends on the rating of the neighbor contents, especially those who have a very close ancestor in the hierarchy (Hr, Hierarchical relations).

$$Index_{\text{Affinity}} = \rho_1 \times A_c + \rho_2 \times C_p + \rho_3 \times E_p + \rho_4 \times H_R A_c,$$
$$C_p, E_p, H_R \in [0,1] \rho_1 + \rho_2 + \rho_3 + \rho_4 = 1. \quad (1)$$

Equation (1) shows how the final affinity index is calculated by performing a weighted addition of the individual factors. After various tests, and taken into account the particularities of the television domain, the following weights have been considered as appropriated: $p_1 = 0.45$, $p_2 = 0.25$, $p_3 = 0.12$, $p_3 = 0.18$. These calculations are performed every time the user accesses a television content for previewing, watching, or looking up information about it. This way this index can increase or decrease depending on the changes in the user's interests.

Finally, OntoUserProfile also stores production rules about user preferences. Analyzing viewer's events automatically generates these rules. OntoTV-InferredRules module carries out this process, described in the following section. Jess inference engine fires the rules whose antecedent evaluates to true. These rules increase or decrease the affinity values of all the content references that fit with the preconditions. For example, if Peter usually watches sports on the weekend, the module OntoTV-InferredRules, which is described in the next section, creates a rule that describes this behavior. The inference engine will trigger this rule every weekend, so the affinity values for sports contents (football, basketball, baseball, motor sports...) will be increased to some degree. This way, the information expressed by the rules is taken into account in operations that require personalization.

$$\textit{if}(User == Peter) AND(isWeekend()) \Rightarrow IncreaseIndexs(Sport, 0.10). \quad (2)$$

This process allows refining the knowledge about the user's preferences by modifying the indexes stored in his/her profile, thus resulting in more accurate suggestions.

Finally, it is necessary to point out that users' behavior changes over time, so all these patterns that are expressed in rules may sometimes become outdated. The current version of the system does not implement strategies for deleting these old patterns, but in future OntoTV's developments is intended to allow the user to manually reset these patterns, or identify all those rules not fired by inference engine the during a certain period of time in order to remove them from the system.

5. Processing: Advanced Features for the Viewer

OntoTV system has modules that carry out operations for meeting the users' needs. On the one hand, searches and recommendations are very important, because they make it possible for users to receive information about contents. On the other hand, the viewer's behavior is analyzed to infer his/her habits and preferences.

5.1. *Inferring user's habits: OntoTV-InferredRules module*

This module receives all the events generated by the user in the client device, for example content search requests, changes of channel, etc. and creates log files. New information about users can be inferred by applying data mining techniques [12] and learning algorithms on this data.

First, the module examines the log files, looking for contents that have comprised a majority of the total viewing time. The examination will show the viewer's most-viewed content, which will indicate his favorite genre. Then, some actions that repeat periodically are detected to infer user patterns. In both processes, new rules are created and stored in the form of SWRL rules in the OntoUserProfile ontology. For example, if the user always watches news channels early in the morning, the system detects these repetitions, and generates a rule that describes this behavior. As a result, during these hours search and recommendation engines give priority to the viewer's preferred content because the appropriate rules are fired.

These mechanisms increase the degree of personalization in the system's operations. New strategies could be added to this module in the future in order to enhance the learning process.

5.2. *Features for the viewer: OntoTV-Core module*

The OntoTV-Core module executes operations on the previously stored data. The inputs are an instruction that indicates the kind of processing to be executed, and also the data stored in OntoTV-Data. The outputs are lists of content descriptions that satisfy the premises specified in the input operation. These lists are sent to the viewers who request the information. The best way to represent the lists is using TV-Anytime format. This standard becomes the basis for information interchange in OntoTV because it supports not only data collecting but also the representation of the results.

Every description has an identifier that allows the client device to find the physical location of the content, in order to play it. CRIDs (Content Reference Identifiers), proposed by the TV-Anytime standard, have been used in this case. They are basically URI's that separate the identification process from the location of the content.

5.2.1. *OntoTV-SearchEngine component*

OntoTV searches information about television programs on the stored data with the OntoTV-SearchEngine component, which includes the algorithms that are responsible for the processing of these operations. Content searches by attributes (title, date, description, etc.), by category or by similarity to other contents are also considered. OntoTV-SearchEngine receives the search instructions and extracts the parameters explicitly entered by the viewer using the input device.

The SPARQL query language has been used in this component. It allows high-level queries that avoid more complex operations over the ontological structure. Results obtained with these techniques are filtered and sorted according to explicit and implicit user preferences.

5.2.2. *OntoTV-Recommender component*

Content recommendation is one of the most important features of OntoTV system. Its aim is to provide a list of contents that are consistent with the user's preferences as defined in their respective profiles. The decision of using ontologies has numerous advantages in recommendation operations [13]. The inference processes on the class hierarchy extract not only the information explicitly represented in each instance, but also all the implicit knowledge in the ontological model.

According to the origin of the information about user preferences, there are explicit or implicit recommenders. The OntoTV-Recommender component works in both ways. The user explicitly provides information by filling in the GUI forms that are shown on his device's screen. This information is stored in the OntoUserProfile ontology and it is ready to be used. And also the implicit information about user preferences is generated by the "OntoTV-InferredRules" component, which looks for patterns in the user's events and create rules that describe his behavior.

But recommendation systems can be mainly classified into two major categories: content-based systems and collaborative recommendation systems [14]. Both alternatives present weaknesses: content-based recommenders specialize too much in programs that are similar to those that the viewer has seen before, and collaborative recommenders are too dependent on the personal opinion of the users. Taking into account these issues and based on some strategies proposed in [15], the OntoTV-Recommender component implements a hybrid recommendation mechanism:

- Content descriptions stored in OntoDataTV are compared with the viewer's preferences to make *content-based recommendations*. Classics approaches only take into account the distance between every of the attributes of the compared items. But OntoTV system takes advantage of the semantic relationships between the matched programs in two different ways:
  (a) The OntoDataTV ontology classifies the TV contents in a hierarchical structure. So, the closer two programs are in this hierarchy, and the deeper their nearest common ancestor in it, the stronger the hierarchical relationship between both contents [16]. Of course, if the nearest common ancestor of the two matched programs is the hierarchy root class, there is no hierarchical similarity between them. In Fig. 4 there is a hierarchical relationship between "Lost" and "Nothing Hill".

(b) It is possible to discover implicit relations between contents that share semantic characteristics in certain attributes represented in the ontology (like actor, producer, etc). May be that the compared attributes have not the same value, so classic methods discard them as matching candidates. But the semantic approach concludes that if these values are instances — equal or different — of the same ancestors of the TV ontology, it is possible to relate them.

- The OntoUserProfile ontology stores data about the viewer, such as explicit preferences, contents he has rated, etc. By finding users' profiles that are similar to the current viewer's profile, the system can recommend programs rated highest by these users. The first step is to form groups of viewers that have similar interests. The method used for this purpose is the creation of rating vectors for all of the viewers. The vectors' components correspond to the affinity indexes stored in the dynamic part of the OntoUserProfile ontology. According to some related works [17], the Pearson-r correlation between the rating vectors is computed, resulting in the highest values for those users who prefer contents that belong to the same hierarchy classes.

In a final stage the results obtained by both methods are ranked, in order to recommend the more relevant contents. OntoTV gives more priority to content-based recommendation results, so it increases their recommendation scores before selecting the candidates. Also, if one of the contents is a solution in both methods, it is immediately considered as a recommendation result.

5.2.3. *Other components*

OntoTV-Core module has been designed to add extra components in OntoTV-Core when new requirements are needed. For example, a component called OntoTV-Guide can be created for generating personalized program guides. In these guides, all the contents are consistent with the preferences of the viewers who requested the operation, regardless of the channel that broadcast the contents. This is an example of how additional components can be added to meet new processing requirements in the OntoTV-Core module.

6. Presentation: Independence of the Platform

OntoTV has been designed to be used regardless of the particular characteristics of the viewer's device. The only requirements are an Internet connection that supports the client-server architecture and a TV system that meets certain characteristics that ensure the execution of the client application.

Nowadays, some MHP implementations are installed in many receivers. They run Java applications called Xlets, support TCP/IP Internet protocols, access graphical libraries, use different input devices, etc. Therefore, it is possible to develop an OntoTV client application for MHP, as explained in [18]. Another new option is

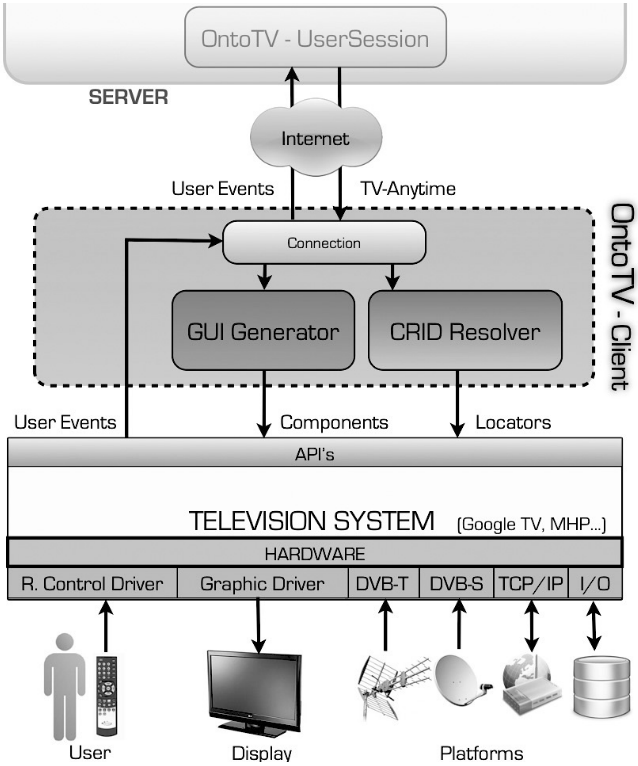

Fig. 5. Schema of the OntoTV-Client module.

Google TV, which provides access to multiple digital platforms and incorporates advanced graphic interfaces based on HTML5 and CSS3. Google TV is valid as well.

Figure 5 shows show the OntoTV-Client application connects to the server, sends the user's requests, and receives the results derived from these requests. This application runs on a television system, installed on the client device, which provides the necessary libraries for graphical interfaces, user event handling, etc. It consists of three components, each of them with a specific functionality.

- The "GUI Generator" generates the graphical interfaces. Dialog boxes, menus, lists of contents, etc. are displayed on the client screen to show the processing results. Also, data entry forms are built so they viewers can add information to the system.
- The "CRID Resolver" takes the CRID reference included in the description of each television content and finds the digital platform, channel, date and time at which the provider broadcasts it. With this information, OntoTV-Client is able to play the content. To carry out this location process it is necessary to connect to external authorities that store [CRID-location] pairs, like DNS services do.
- "Connection" component is responsible for the TCP/IP communication between the user device and the OntoTV server. On the server side, the OntoTV-UserSession

module acts as a communication port (see Fig. 1). It listens for users' connections and allows the OntoTV-Client to exchange messages with the server.

7. Implementation and Results

In this section, the results obtained by an implementation of OntoTV system are going to be presented. This way, it is demonstrated that the user's experience is improved when using this system.

7.1. *OntoTV implementation*

The details of the OntoTV implementation for each of the key aspects are shown below. First, OntoTV-InfoCollector module will collect data from the following TV content information sources in Spain:

- DTT (Digital Terrestrial Television). It is the public television platform in this country, and it offers around 30 channels, simple programming guides based on DVB-SI, and broad coverage throughout the territory. The DDT-ReadComponent performs this task using strategies described in [6].
- The web page "La GuíaTV" (http://www.laguiatv.com/). It stores schedules and content descriptions of programs for major television channels in Spain. There is one component in "GrabberAgents" that uses the open source package xmltv-0.5.59 for converting the HTML code of the page into a XMLTV file. Then, this file is translated into TV-Anytime format used in OntoTV.
- The web page "Mi GuíaTV" (www.miguiatv.com/). It is similar to the above case.
- The IMDB database, where detailed film descriptions can be retrieved.
- Information about films retrieved from Linked Data resources as described in the previous sections.

The use of grabbers may violate copyrights unless the content provider explicitly accepts their use. In this work, "La GuiaTV" and "Mi GuiaTV" have allowed us to perform this extraction processes for research purposes. As the objective here is only to test the proposed data integration strategies over non-structured sources, there are no legal issues. Anyway, these aspects have to be reconsidered if the system is finally deployed in a commercial way.

Regarding the knowledge base, television domain ontology OntoDataTV has been implemented based on the AVATAR one, and viewers' preferences have been stored into OntoUserProfile ontology. OntoTV-Core includes three components, OntoTV-SearchEngine, OntoTV-Recommender and OntoTV-Guide.

The server-side modules have been executed on a sufficiently powerful personal computer with Internet connection. For the execution of the client application, a small form computer with Wi-Fi connection has been used. It has a DVB-T tuner card and an infrared remote control, and the video output to TV is done through

HDMI interface. OntoTV-Client runs on this hardware as a C++ application under *Debian* GNU/Linux operating system. Graphical interface has been developed using Qt, and V4L-DVB drivers get the DTT video stream from the DTT card. There is no need for a CRID Resolver, due to the simplicity of the data sources. Although this software is not specific for digital television, the required functionalities have been successfully implemented.

7.2. *Study case*

This study aims to measure the degree of satisfaction that users feel when using OntoTV, compared to other ways of accessing television content information. Thirty-six people have been selected for the test. It is composed of students from the Polytechnic School at the University of Extremadura and some of their relatives. This way results are independent of the subjects' previous experience in the use of new technologies.

Each participant must answer a set of questions about television contents that are going to be broadcasted during the week. For make the test more independent of television schedules, the 36 person sample has been divided into 3 groups of 12, and each of them has completed the test in different weeks. Finally, not all the groups use the same resources when answering the questions. Four of them will use a DVB-T Strong SRT5510 set top box, which only offers the viewer the possibility of navigating over a classic programming guide and sequentially locating the desired content. Another four people will have the same decoder and also a computer with Internet connection to look for further information. In this case, apart from exploring the SRT5510's programming guide, the viewer can use the available search engines on Internet for performing queries in the computer. The rest uses the mini-computer where OntoTV-Client is running, so they can take profit of the operations that can be performed in the system: search, recommendation, and generation of personalized program guides. Anyway, the user does not need to manually create the queries in any of the cases; he only will use the already implemented features that every particular alternative provides.

In the case of OntoTV-Client, the GUI has been designed trying to make the interaction with the system as easy as possible. For this reason OntoTV always shows the user the minimum information in every step. Indeed, most of the time the viewer does not watch any element on the screen apart of the audiovisual content itself. Only when periodic recommendations are executed, a very simple window with the titles of the suggested programs appears. Also, when the user presses the "start" button, a menu with the three available operations is shown on the screen. The dialog forms for every of these operations are easy to fill in: for example, for executing a query it is only necessary to provide a "keyword", "category", and/or the "Date" as shown in Fig. 6. The results of these operations are lists of contents that match the given criteria. Finally if the user selects one of the programs in the list, some actions

Fig. 6. The search dialog in OntoTV.

are suggested: watch the program (it is being broadcasted now), record, obtain extra information, or return to the main menu.

Also, it should be noted that before performing the experiment, OntoTV users have filled out a short form on the screen about their TV preferences. It does not last long, so it has not been taken into account for the experiment.

The results have been obtained by annotating the time spent by the subject in answering each question, as well as a numeric value between 1 and 10 that expresses the degree of user's satisfaction when he/she solves a specific question with the available resources.

By analyzing the average time to solve each question, it is possible to find out various facts. First, when task complexity increases, more minutes are needed to resolve the task when using a conventional decoder. By using the Strong decoder and the personal computer, the response time also increases with the complexity of the question, but to a lesser degree. This is because Internet has more advanced sources and tools for information retrieval. Finally, with OntoTV system response times remains approximately constant, due to the fact that all questions can be solved directly by a single operation.

By analyzing user's ratings, the decoder looks suitable only for very simple questions about the program guide information, like Question 1. By adding the possibility of using the computer, ratings for the more difficult questions have decreased, but in a much lower degree. The problem is that viewers have to combine Internet search operations with the use of the decoder, and this is not comfortable at all. Finally, users have been pleased with the ease of use of OntoTV. They have quickly resolved the considered questions. Even their opinion about the system has

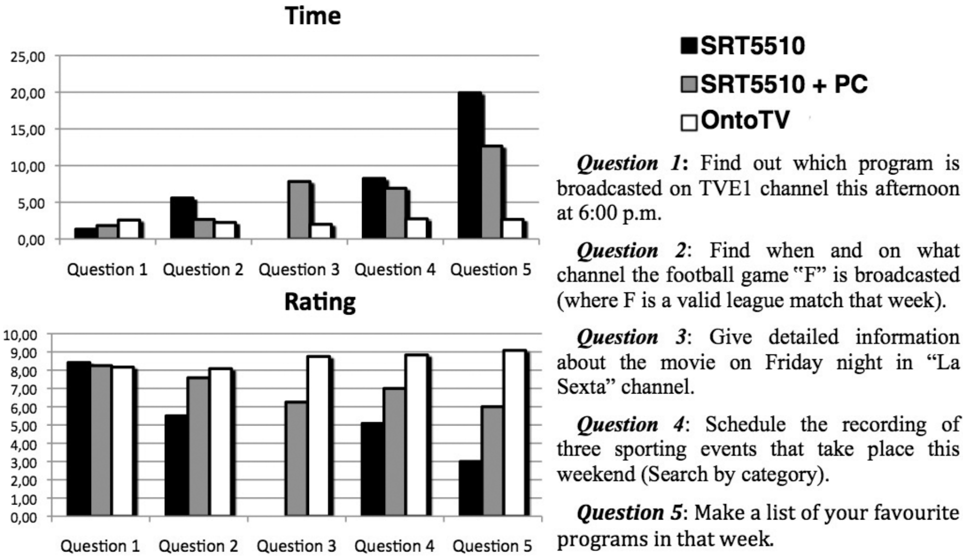

Fig. 7. Results of the average expended time and rating for every of the proposed questions.

improved when they realize that OntoTV solves the more difficult questions with the same efficiency than the easy ones.

8. Conclusions

Nowadays, users do not have any service that offers information about the television contents in an efficient and organized way, and regardless of the TV platform and the characteristics of the viewer's device. OntoTV system has been designed to overcome these problems by using semantic multimedia techniques. This system collects television information and represents it in an ontology-based model, which improves the precision of the operations and makes easier to take into account users' habits and preferences.

More specifically, OntoTV integrates information from various digital television platforms and collects program descriptions, program schedules, user opinions, etc. If this information is incomplete, it can also access alternative resources that improve it. Then it represents information about television contents and viewers through the use of ontologies. This way, the stored knowledge can be shared, and it is flexible to future changes in requirements. Inferences over the existing data can be made for executing advanced operations on the available information, such as searches with a high degree of personalization, recommendations, or generations of programming guides according to the viewers' preferences. The goal is to provide the user with truly useful features. Finally, the system implements a client/server architecture that allows any device with a network connection to use this semantic information management system, regardless of the digital TV system that has been used.

After a first version of OntoTV has been developed, it was found that the proposed model improves the viewer's experience. It has succeeded in defining a global service for television content management that is able to represent not only television content description, but also implicit and explicit viewer's preferences. Also, this offers more functionalities than traditional set top boxes.

Actually it is necessary to continue working on this system. Due to the flexibility of the model, new agents for collecting data about contents and users can be added to it, and it is possible to create new client applications for supporting other television platforms. OntoDataTV ontology can be improved by adding more complex relations between the items, as for example synonym and meronymy, in order to refine the way of structuring and representing the television domain. Also, it would be very interesting to add more data mining methods for including as many implicit preferences as possible in the user ontology. However, the basic architecture for a more universal content management system has been defined. And it was probed that the semantic methodologies applied in it are useful for the viewers because they can enjoy a more interesting television system for accessing information about television content.

Acknowledgments

José Luis Redondo-García is supported by the research grant PD10006 from Junta de Extremadura. This work has been partially funded by the Spanish Ministry of Education and Science and ERDF (the European Regional Development Fund), under contract TIN2011-27340.

References

[1] Multimedia Home Platform (MHP) Specification 1.2.2, Digital Video Broadcasting, ETSI TS 102 727 V1.1.1 (2010-01).

[2] Y. Blanco-Fernández *et al.*, Exploiting synergies between semantic reasoning and personalization strategies in intelligent recommender systems: A case study, *Journal of Systems and Software* **81** (2008) 2371–2385.

[3] B. Schopman *et al.*, NoTube: Making the Web part of personalised TV, *WebSci* **10** (2010).

[4] L. Aroyo *et al.*, SenSee framework for personalized access to TV content, *Interactive TV: A Shared Experience*, 2007, pp. 156–165.

[5] B. S. Manjunath, P. Salembier and T. Sikora, *Introduction to MPEG-7 Multimedia Content Description Interface* (Wiley, 2002).

[6] J. L. Redondo-Garcia, P. Valiente-Rocha and A. Lozano-Tello, Ontology-based system for content management in digital television, in *CISTI, 2010*, pp. 277–283.

[7] S. J. Davis, C. H. Ritz and I. S. Burnett, Using social networking and collections to enable video semantic acquisition, *IEEE MultiMedia* **16** (2009) 52–61.

[8] J. L. Redondo-Garcia and A. Lozano-Tello, Recolección de datos sobre contenidos televisivos en el sistema OntoTV, *CISTI, 2011*, pp. 91–97.

[9] A. Gomez-Perez, O. Corcho and M. Fernandez-Lopez, *Ontological Engineering: With Examples from the Areas of Knowledge Management, e-Commerce and the Semantic Web* (Springer-Verlag, 2004).

[10] E. Friedman-Hill, *Jess in Action: Java Rule-Based Systems* (Manning Pub. Co., 2003).

[11] D. Heckmann *et al.*, Gumo — the general user model ontology, *User Modeling*, 2005, pp. 428−432.

[12] W. E. Spangler, M. Gal-Or and J. H. May, Using data mining to profile TV viewers, *Commun. ACM* (2003), pp. 66−72.

[13] L. Buriano *et al.*, The role of ontologies in context-aware recommender systems, in *MDM2006*, (2006), p. 80.

[14] J. Salter and N. Antonopoulus, CinemaScreen recommender agent: Combining collaborative filtering and content-based filtering, *IEEE Intelligent Systems* (2006), pp. 35−41.

[15] G. Lekakos and P. Caravelas, A hybrid approach for movie recommendation, *Multimedia Tools and Applications* (2008), pp. 55−70.

[16] P. Resnik, Semantic similarity in a taxonomy: An information-based measure and its application to problems of ambiguity in natural language, *Journal of Artificial Intelligence Research* (1999), pp. 95−130.

[17] B. Sarwar, G. Karypis, J. Konstan and J. Riedl, Item-based collaborative filtering recommendation algorithms, in *10th World Wide Web Conference* (2001), pp. 285−295.

[18] J. L. Redondo García *et al.*, The new digital television and the interactivity in its multimedia applications, in *SIGMAP 2009*, pp. 167−172.