Linked Data Methodologies for Managing Information about Television Content

José Luis Redondo-García[1], Vicente Botón-Fernández[2] and Adolfo Lozano-Tello[2]
[1]*Multimedia Modeling and Interaction Department,* EURECOM, Sophia-Antipolis, France
[2]*Quercus Software Engineering Group,* Universidad de Extremadura. Cáceres, Spain

***Abstract* — OntoTV is a television information management system designed for improving the quality and quantity of the information available in the current television platforms. In order to achieve this objective, OntoTV (1) collects the information offered by the broadcasters, (2) integrates it into a ontology-based data structure, (3) extracts extra data from alternative television sources, and (4) makes possible for the user to perform queries over the stored information.**

This document shows the way Linked Data methodologies have been applied in OntoTV system, and the improvements in the data consumption and publication processes that have been obtained as result. On the one hand, the possibility of accessing to information available in the Web of Data has made possible to offer more complete descriptions about the programs, as well as more detailed guides than those obtained by using classic collection methods. On the other hand, as the information of the television programs and channels is published according to the Linked Data philosophy, it becomes available not only for OntoTV clients, but also for other agents able to access Linked Data resources, who could offer the viewer more fresh and innovative features.

***Keywords* — Linked Data, OntoTV, ontology, consuming, publishing.**

I. Introduction

NOWADAYS, the number of television platforms and channels number is growing significantly, so it is not easy for the viewer to decide what he want to watch in a certain moment of the day. Even when providers offer some descriptions about the programmes they broadcast, this information is not detailed enough and does not permit to perform advanced operations like content recommendations.

The creation of a management information system that gives solution to these problems could be very beneficial for the viewers as well as their television experience. This system aims to become a universal and easy-to-use television solution, able to offer more advanced features than those implemented in classic set-top-boxes. When information is scarce, it should access to external sources in order to complete the missing data, in a transparent and flexible fashion. This way the clients have access to a common television information service, no matter the particular device that is being used: a mobile phone, a decoder, or a personal computer. In previous researches in this same direction [1], OntoTV system was created in order to fulfil these requirements.

OntoTV collects information about television contents from various sources and represents all the data using knowledge engineering and ontologies. However, there are still some problems related to the way OntoTV manages the television information. First, the system uses a kind of software components called “Crawlers”, which retrieve information from non-structured sources like HTML Web pages. These components consume many computational resources, and have to be customized to fit the particularities of every of the considered data sources. Secondly, only the clients who are compatible with the OntoTV’s specifications can access the information stored in his knowledge base.

In this situation, the Linked Data consuming and publishing methodology [2] is gaining presence and importance in the Web. It consists of a set of principles for structuring and interlinking data that make information more useful and easy to reuse by others. As this methodology is built on the top of widely used standards in the Web, such as URI and HTTP, the information shared in this way becomes accessible to both humans and machines. At the end, this interlinked and easy accessible information obtained from different sources is what we commonly known as “Web of Data”.

The main objective in this research is to apply the Linked Data methodology in OntoTV, in order to improve the data collection processes and the viewers’ television experience. To to achieve that, some components for consuming television information from Linked Data sources have been designed. More specifically, OntoTV will retrieve extra information about movies, obtaining more complete electronic programming guides than before. Also, the data stored in the knowledge base will be now published according to Linked Data principles, so it will be available in the Web of Data for all these agents who are able to access to it.

II. OntoTV System

The OntoTV system (ONTOlogy-based management system for digital TeleVision) is a television content information management system that allows the viewers to access data about programs that have been or will be broadcasted in the various digital platforms. Due to the fact that this system incorporates appropriate mechanisms for data acquisition, it can provide the user detailed content descriptions and allows him to perform advanced search and recommendation operations. The system OntoTV was previously presented in [4], where the most important features were shown:

- To **integrate** all the possible information about television content by using different collection mechanisms for accessing the different existing sources.
- To **represent** the collected data by using ontologies, making possible to perform complex reasoning processes and inferences that generate new knowledge [5].
- To **execute** operations over the knowledge base that are interesting for the user. For example searches and recommendations, with a high degree of personalization.
- To allow the user to **interact** with the system in an easy and intuitive way. The client device sends requests for the execution of certain operations, receives the results from the server, and displays them to the user. The viewers can access the system, no matter which kind of implementation is running on their devices: MHP, Google TV, Media Centers, etc.

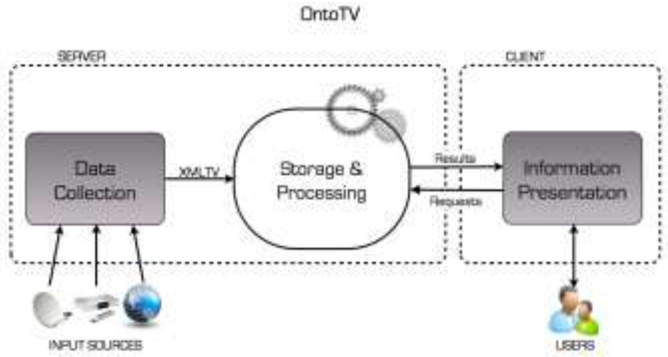


Fig. 1. Main modules inside OntoTV system.

Figure 1 shows a schema of the OntoTV system. The "Storage and Processing" module includes the television content ontology and the different search and recommendation algorithms that are executed over the knowledge base. "Data Collection" and "Information Presentation" modules will be described in more detail below, since they are the ones that will be modified for being compliant to Linked Data principles. This is done to improve the way OntoTV system consumes and publishes the data.

A. Data Collection module

This module directly reads the data from the sources supported by the system. The process consists of being able to interpret the format of a certain input source, and transform the extracted information to the XMLTV format, which is the one used in the system for representing the input files.

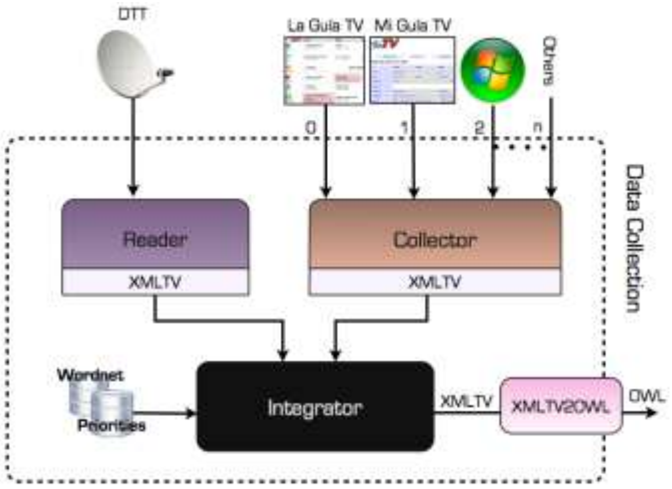


Fig. 2. "Data Collection" module in OntoTV

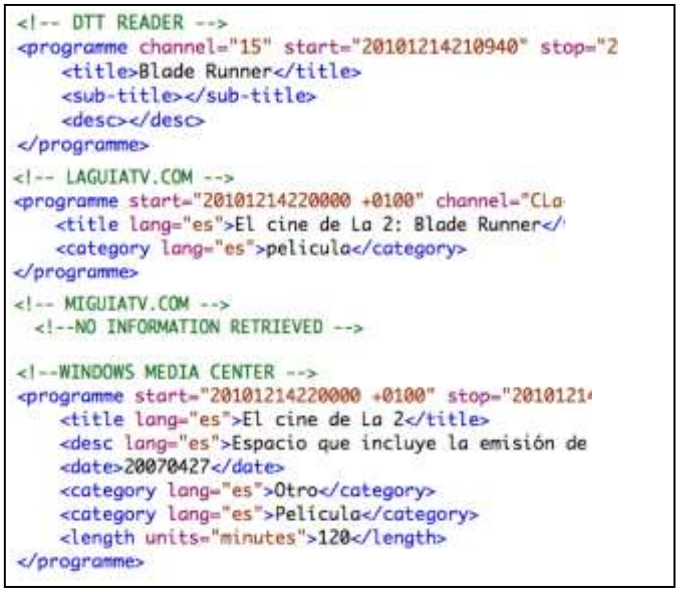
In Figure 2 the different components inside the "Data Collection" module can be seen. "Reader" components extract the data that television broadcasters offer in their platforms and channels. "Collector" components connect to external server, normally using the TCP/IP protocol, in order to obtain alternative programming guides. Finally, the component called "Integrator" includes in a single file all the information in XMLTV format that has been previously retrieved by the other two components.

Data Sources supported in OntoTV

According to the schema shown in Figure 2, "Reader" and "Collector" components access the following television data sources to feed OntoTV's knowledge base:

- Information included in the DTT data stream, accessed by performing the processes described in [1].
- Accessing to "La Guía TV", a Web page that contains information regarding to television contents broadcasted by the major television channels in Spain. It is necessary to perform translation processes from HTML to XMLTV.
- Accessing to "Mi Guía TV". It is a Web page with similar characteristics than the previous case.
- Accessing to "Windows Media Center" guide. Microsoft offers very complete programming guides for the main television channels in Spain. OntoTV extracts information from these guides and converts them into XMLTV format.

Figure 3 shows three fragments of XMLTV files related to the film "Blade Runner", broadcasted in Spain on the channel "Antena 3" on December 14, 2010. There are some differences in the level of detail provided by each source: for example the fragment corresponding to "Mi GuíaTV" is completely empty, while the "Windows Media Center" one contains precise information about the categories associated with that particular content.

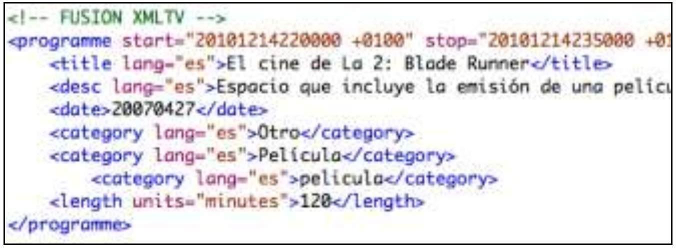


```
<!-- DTT READER -->
<programme channel="15" start="20101214210940" stop="2
    <title>Blade Runner</title>
    <sub-title></sub-title>
    <desc></desc>
</programme>
<!-- LAGUIATV.COM -->
<programme start="20101214220000 +0100" channel="CLa
    <title lang="es">El cine de La 2: Blade Runner</
    <category lang="es">pelicula</category>
</programme>
<!-- MIGUIATV.COM -->
  <!--NO INFORMATION RETRIEVED -->

<!--WINDOWS MEDIA CENTER -->
<programme start="20101214220000 +0100" stop="2010121
    <title lang="es">El cine de La 2</title>
    <desc lang="es">Espacio que incluye la emisión de
    <date>20070427</date>
    <category lang="es">Otro</category>
    <category lang="es">Película</category>
    <length units="minutes">120</length>
</programme>
```

Fig. 3. XMLTV fragments collected from the various considered data sources.

Merging Duplicate Instances of Television Programs. OntoTV is able to detect if descriptions from different sources refer to the same content. Duplicate descriptions about the same program are identified and resolved according to mechanisms described in [4]. Various criteria are taken into account in this process: *spatio-temporal similarity* of content (if two descriptions refer to the same channel, beginning and ending almost at the same time, then it is highly possible that both belong to the same program), *similarity in the titles*, (applying relative comparison string functions as the Levenshtein one [3]), or *global similarity* (given two descriptions, we look for words that appear in both description, regardless of the exact position in the text).



```
<!-- FUSION XMLTV -->
<programme start="20101214220000 +0100" stop="20101214235000 +01
    <title lang="es">El cine de La 2: Blade Runner</title>
    <desc lang="es">Espacio que incluye la emisión de una pelicu
    <date>20070427</date>
    <category lang="es">Otro</category>
    <category lang="es">Película</category>
        <category lang="es">pelicula</category>
    <length units="minutes">120</length>
</programme>
```

Fig. 4. XMLTV description obtained after merging the information from the considered data sources.

Once all the descriptions that belong to the same content have been identificated, it is neccesary to merge them into a single instance, as shown in Figure 4. If a description provides one attribute that is missing in the rest of sources, this field is taken immediately. However, if there is some overlapped parameters in the descriptions, the involved fields are concatenated if possible. If not, those who come from less important sources are discarded. At the end of this step for each content we obtain a unique description that is more complete and detailed than those extracted individually from each source.

Disadvantages of this Approach

As can be seen, all the considered sources provide information about television content. The problem is that the consuming data strategies used in each case are different: the access to DTT is done by interpreting DVB-SI tables, information from Web pages is extracted from certain HTML tags, etc. So each time a new data source needs to be incorporated to the system, is necessary to implement a new access method, as well as integrate it into the global data collection workflow. This process usually requires considerable engineering efforts, which makes more difficult for OntoTV to access new data stores where new television information can be found.

In addition to this lack of uniformity in the collection methods, the processes involved in them are usually very resource intensive because the information is not sufficiently structured.

B. Presentation of the Information

The client-server architecture that has been implemented in OntoTV makes possible that a great variety of television devices can access to the functionalities offered by this system regardless of their particular characteristics. This fact is especially important today, given the different options that are available on the market: MHP set-top-boxes, Google TV televisions, mobile devices with Android operating system, etc. For all these platforms it is possible to develop a client application, called "OntoTV-Client", which performs all the necessary functions to present the television information to the viewer. The premises are to have an Internet connection (for stablishing the client-server communication), as well as being able to use platform-specific libraries for tracking the user's actions, generate graphical interfaces, and interchange messages between client and server.

TABLE I
MESSAGE INTERCHANGING IN ONTOTV'S CLIENTS

Type of Message	Output	Input
Content Management	- Search request, taking into account various criteria. - Request for a detailed description of a particular content. - Ask for a personalized electronic programming guide.	List of contents that match the selected criteria. Description of a particular content. List of contents that match the user preferences.
User Data	- Sending of local events (like button presses, menu navigation, etc.) - Sending of information available on the explicit preferences menu.	User profile that is stored on the server.
Server Connection	- Open connection request. - Closing conncetion request.	Confirmation of successful connection. Confirmation of successful disconnection.

However this information exchange is done by using certain types of messages and a communication sequence that have been defined beforehand and are exclusive for OntoTV system. Then, for establishing a valid communication with the server, a client must implement this particular set of requests and responses.

Table 1 lists the most important messages the client sends and receives when communicating with OntoTV server. The HTTP protocol and the interchange of XMLTV files over TCP/IP are the basis for implementing those messages.

Disadvantages of this Approach

The problem with this approach is that, despite being independent of the platform used by the consumer, it is always necessary to implement this specific set of messages, even when the agent is not exactly a OntoTV client but another entity that eventually needs television information.

For example, a website that offers the user some miscellaneous information can access OntoTV for retrieving the broadcast times of different television programs, but it needs to incorporate all the communication logic that an OntoTV's client's is supposed to use.

III. Applying Linked Data Methodology

After analizing the way OntoTV operates when providing different features to the viewers, various problems have arised. On the one hand, traditional mechanisms for extracting information from television sources have been probed to be inefficient, due to the heterogeneity in the access methods and the accessing to non-structured information. On the other hand, only clients that are compliant with OntoTV specification can access its television information. This section aims to solve these problems by applying the Linked Data consuming and publishing principles [6], continuing the research line initiated by other television systems that also have used semantic technologies, as Notube [7], [8].

A. Linked Data Consumption

This section shows how to incorporate new Linked Data consumption strategies in the module "Data Collection", in order to increase the amount of television content information available in its knowledge base.

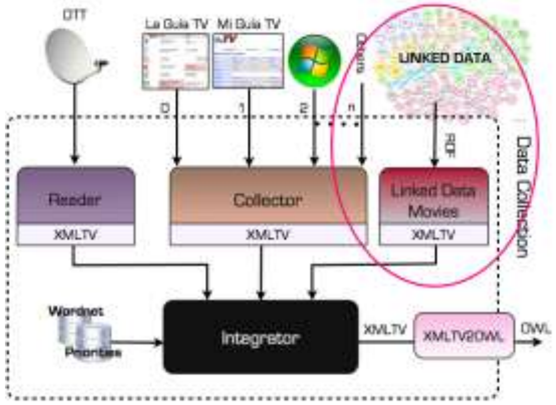


Fig. 5. Extended "Collection Data" module that accesses the Web of Data.

Specifically, the objective is to describe the way the new component called "LinkedData Movies" operates (see Figure 5). As summary, this component accesses Linked Data resources, identifies certain information about movies that is available on the Web of Data, and complete the missing parts of the XMLTV program guides that OntoTV has previously retrieved.

Alternatives for accessing the Web of Data

Several alternatives for accessing information about television content in the Web of Data have been studied. The most significant ones are shown below:

- Accessing to LinkedMDB dataset. It is possible to execute SPARQL queries over the entry point that this dataset provides, in order to obtain information about movies. However, although LinkedMDB is intended to be in the Web of Data the same than IMBD is today in Web of Documents, there are still a lot of films entries missing.
- To implement the method described in [6], which applies the "Crawling" consumption pattern. It consists of using Jena TDB[2] to create a local storage structure where the information collected by the Linked Data crawler DSpider[3] is continuously added. The disadvantage of this approach is that it has a high computational cost. In addition, the collection process is slow and must be repeated periodically to ensure that the information inside the local storage is not out-dated.
- Access the semantic mashup SIG.MA. The advantage of this alternative is that it is possible to access to relevant information from a great variety of semantic sources, without executing very intensive and slow collection processes. In addition, SIG.MA performs frequent updates in their data indexes so the obtained information about movies is updated enough.

The "LinkedData Movies" component

The component "LinkedData Movies" has been coded in Java and performs the following actions in order to extract information about movies from the Web of Data:

a) Getting the movie descriptions in RDF format. The basic mechanism to access Linked Data on the Web is to resolve HTTP URIs for retrieving a certain RDF data fragment. In the case of the SIG.MA mashup, it is necessary to perform an HTTP request to the following URL "http://sig.ma/search?q=moviename", where "moviename" is a string indicating the name of the movie we are looking for. Code 1 shows how to obtain RDF data from SIG.MA using the library "org.apache.commons.httpclient":

Code 1. Requesting RDF Data to the Server

[2] http://openjena.org/TDB/
[3] http://code.google.com/p/ldspider/

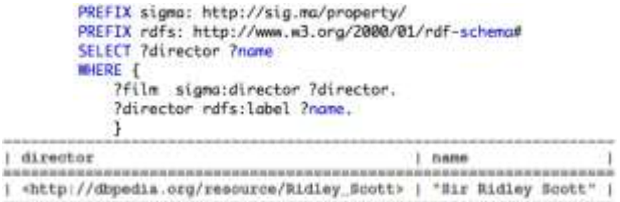
```
import org.apache.commons.httpclient.*;
   //Get Method
   HttpClient client = new HttpClient();
   HttpMethod method = new GetMethod(urlsigma + filmName);
   method.addRequestHeader("Accept",
   "application/rdf+xml");
   int responseCode = client.executeMethod(method);
   //Write RDF to FILE
   InputStream is = method.getResponseBodyAsStream();
   OutputStream os = new FileOutputStream(rdfFile);
   byte[] buffer = new byte[4096];
   for (int n; (n = is.read(buffer)) != -1;)
       os.write(buffer, 0, n);
```

b) Use SPARQL queries to extract the desired information from the previously obtained RDF file. The RDF file, which contains information about a particular film, is already available in the consumer side. So it is possible to extract the desired fragment of information by executing SPARQL queries over it. The "Jena ARQ" library has been used for this purpose, as shown below.

CODE 2. EXECUTING SPARQL QUERIES OVER THE RDF FILE

```
   import com.hp.hpl.jena.query.*;
   Model m;
   m = ModelFactory.createMemModelMaker().createModel("");
   model.read(in,null);
   //Execute the Query
   Query query = QueryFactory.create(stquery);
   QueryExecution qe;
   qe = QueryExecutionFactory.create(query, m);
   ResultSet results = qe.execSelect();
```

Code 2 is able to execute the SPARQL query stored inside the variable "stquery". Figure 6 shows an example that extracts the name of the film's director by accessing the property "director", which is included on the SIG.MA vocabulary (http://sig.ma/property/).



```
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.
    }
---------------------------------------------------------------
| director                                     | name              |
===============================================================
| <http://dbpedia.org/resource/Ridley_Scott> | "Sir Ridley Scott" |
---------------------------------------------------------------
```

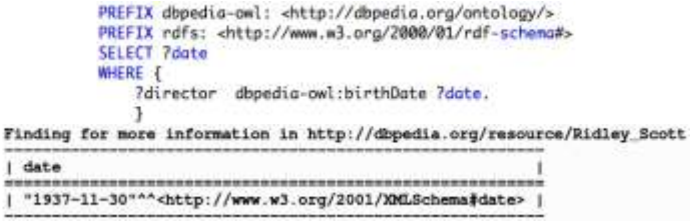
Fig. 6. SPARQL query for retrieving the name of the film's director.

In a similar way, it is possible to obtain also more film's attributes like the language, the country, the length, and others shown in Table 2. This way descriptions about movies that are available in OntoTV system become more detailed and complete than those obtained before accessing the Web of Data.

TABLE II
OTHER ITEMS ABOUT FILMS THAT CAN BE RETRIEVED FROM SIG.MA

Item	XMLTV Element	SIG.MA Property
Language	tv.programme.language	<sigma:language>
Length	tv.programme.length	<sigma:runtime>
Country	tv.programme.country	<sigma:country
Rating	tv.programme.rating	<sigma:ratings>
Director	tv.programme.credits.director	<sigma:director>
Actor	tv.programme.credits.actor	<sigma:starring>
Writer	tv.programme.credits.writer	<sigma:writer>
Producer	tv.programme.credits.producer	<sigma:producer>
Composer	tv.programme.credits.composer	<sigma:music_composer>
Image	tv.programme.icon	<sigma:picture>

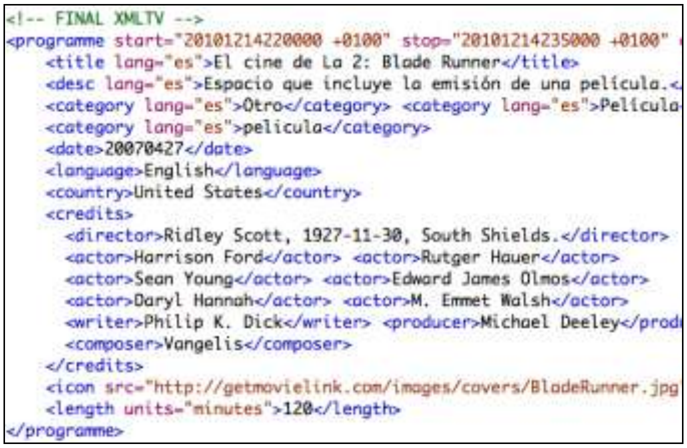
c) Accessing to Other Datasets. The Linked Data philosophy is based on the idea of navigating through the global knowledge. For this reason, if the information that SIG.MA offers is insufficient, it is possible to retrieve alternative data by following the links available in the RDF triples. For example, in Figure 6, the URI for the director Ridley Scott refers to a document in the DBpedia dataset. Additional data can be obtained when URI is resolved with the same process described avobe, as seen in Figure 7:



```
PREFIX dbpedia-owl: <http://dbpedia.org/ontology/>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
SELECT ?date
WHERE {
    ?director  dbpedia-owl:birthDate ?date.
    }
Finding for more information in http://dbpedia.org/resource/Ridley_Scott
----------------------------------------------------------
| date                                                     |
==========================================================
| "1937-11-30"^^<http://www.w3.org/2001/XMLSchema#date> |
----------------------------------------------------------
```

Fig. 7. SPARQL query for retrieving the director's birth date.

In the end, OntoTV stores much more information about the film on which the collection process has been applied. For example Figure 8 shows how the description of "Blade Runner" is much more detailed than before the access to the Web of Data (check again Figure 4 for a better comparison). All this extra information allows the system to offer the viewers more accurate results when executing operations, such as search and recommendations.



```
<!-- FINAL XMLTV -->
<programme start="20101214220000 +0100" stop="20101214235000 +0100"
    <title lang="es">El cine de La 2: Blade Runner</title>
    <desc lang="es">Espacio que incluye la emisión de una película.<
    <category lang="es">Otro</category> <category lang="es">Película
    <category lang="es">pelicula</category>
    <date>20070427</date>
    <language>English</language>
    <country>United States</country>
    <credits>
      <director>Ridley Scott, 1927-11-30, South Shields.</director>
      <actor>Harrison Ford</actor> <actor>Rutger Hauer</actor>
      <actor>Sean Young</actor> <actor>Edward James Olmos</actor>
      <actor>Daryl Hannah</actor> <actor>M. Emmet Walsh</actor>
      <writer>Philip K. Dick</writer> <producer>Michael Deeley</prod
      <composer>Vangelis</composer>
    </credits>
    <icon src="http://getmovielink.com/images/covers/BladeRunner.jpg
    <length units="minutes">120</length>
</programme>
```

Fig. 8. Description available in OntoTV system about the movie "Blade Runner", after accessing information in the Web of Data.

Analyzing the entire collection workflow, it is clear the benefits obtained when consuming information available on the Web of Data over traditional accesses to unstructured data sources. The use of URIs and the HTTP protocol provides a more uniform access to different datasets and makes easier to incorporate new sources in OntoTV system. Likewise, the fact that the data is represented in RDF format and structured according to certain vocabularies (such as SIG.MA), greatly facilitates the way the information is interpreted and processed.

B. *Publishing Data according to Linked Data principles*

As noted in paragraph 2.b, the only way to access the information stored in OntoTV's knowledge base is to implement a predefined and specific communication logic for the interchange of information between the client and the server. This section explains the changes made in OntoTV in order to publish television content descriptions by following Linked Data principles. This way any agent that is able to access the Web of Data can also take profit of them.

Television Domain Ontology

The first step in order to publish data using Linked Data principles is to choose a valid domain vocabulary that allows representation of television information. In previous works, OntoTV used the ontology proposed in AVATAR [9]. However, for the current research this ontology has been replaced by the one used by the BBC (British Broadcasting Corporation), called BBC Programmes. This organization has created this vocabulary by using its wide experience in the use of semantic technologies. This background knowledge has led to consider this alternative as the most suitable one for representing television programs and channels in a standard way, that is one of the main principles in the Linked Data philosophy.

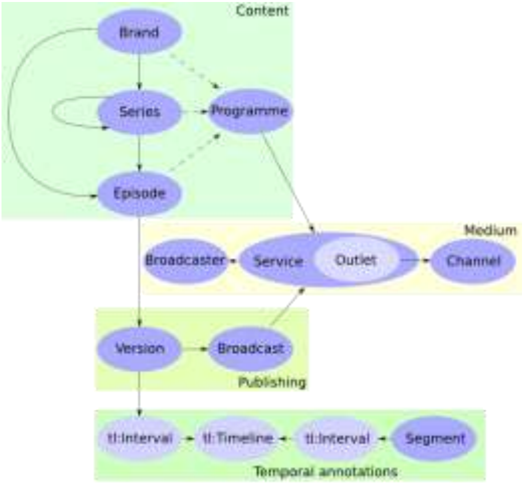


Fig. 9. BBC Programmes (www.bbc.co.uk/ontologies/programmes).

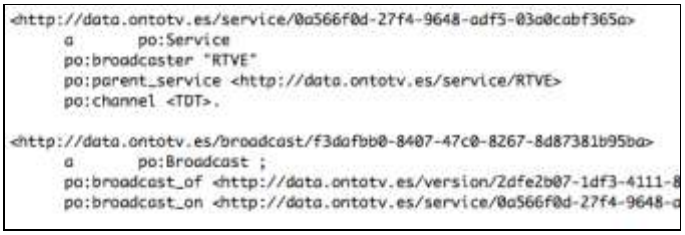
This ontology provides a simple vocabulary that includes multiple classes related to the television content and broadcasters domain. In the Figure 9, the box "Content" (Programme, Brand, Series, Episode) contains classes for representing different types of television content. Inside the "Medium" box we can find the class "Channel" for representing the different kinds of transmission mediums, as well as the class "Broadcaster " for modelling the television organization. The "Publishing" box includes the class "Version", which is very important for representing the different occurrences of a particular program in a certain channel, date and time. Classes inside the box "Temporal Annotations" have not been considered in this research.

Generating the RDF Data

This section describes the different steps to be performed in order to transform the XMLTV information about television channels and programs (previously extracted by the "Data Collection" module) into instances of the BBC Programes ontology.

The software component that performs this translation process is XMLTV2OWL. As seen in Figure 2, this component is included inside the "Data Collection" module. However, it plays an important role in the process of making this information available in the Web of Data, because it is the one who generates the instances that will conform the RDF code. The stages of this process are described in more detail below:

a) Step 1. Each element of type "<channel>" in the XMLTV file is transformed into an instance of the class "Service" in the BBC ontology. Before including this new individual in the knowledge base, it is necessary to check that there are no collisions with those instances that are already stored in OntoTV (because the channel has been previously inserted in the system). Also, an instance of the class "Broadcast" is created in order to relate the current instance of the program with the particular channel that broadcasts it. Figure 10 shows the RDF example code for the film "Blade Runner" in Turtle notation.

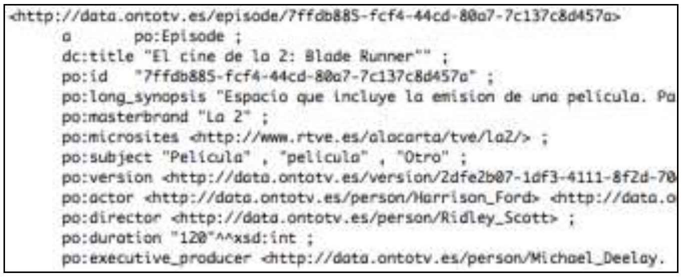


```
<http://data.ontotv.es/service/0a566f0d-27f4-9648-adf5-03a0cabf365a>
        a       po:Service
        po:broadcaster "RTVE"
        po:parent_service <http://data.ontotv.es/service/RTVE>
        po:channel <TDT>.

<http://data.ontotv.es/broadcast/f3dafbb0-8407-47c0-8267-8d87381b95ba>
        a       po:Broadcast ;
        po:broadcast_of <http://data.ontotv.es/version/2dfe2b07-1df3-4111-8
        po:broadcast_on <http://data.ontotv.es/service/0a566f0d-27f4-9648-a
```

Fig. 10. Instances of the classes "Service" and "Broadcast" in the BBC Programmes" ontology.

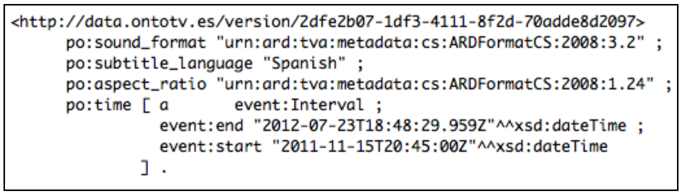
b) *Step 2*. For each "<programme>" XMLTV element: An instance of the class "Programme" is created in the OntoTV's knowledge database, by transforming certain XMLTV fields into their corresponding properties in the BBC ontology. Before including that instance in the knowledge base, it is necessary to check if this particular program has not been previously added to the system. The Figure 11 shows the RDF example code for the movie "Blade Runner":



```
<http://data.ontotv.es/episode/7ffdb885-fcf4-44cd-80a7-7c137c8d457a>
       a       po:Episode ;
       dc:title "El cine de la 2: Blade Runner"" ;
       po:id   "7ffdb885-fcf4-44cd-80a7-7c137c8d457a" ;
       po:long_synopsis "Espacio que incluye la emision de una pelicula. Pa
       po:masterbrand "La 2" ;
       po:microsites <http://www.rtve.es/alacarta/tve/la2/> ;
       po:subject "Pelicula" , "pelicula" , "Otro" ;
       po:version <http://data.ontotv.es/version/2dfe2b07-1df3-4111-8f2d-70
       po:actor <http://data.ontotv.es/person/Harrison_Ford> <http://data.o
       po:director <http://data.ontotv.es/person/Ridley_Scott> ;
       po:duration "120"^^xsd:int ;
       po:executive_producer <http://data.ontotv.es/person/Michael_Deelay.
```

Fig.11. Instance of the class "Episode" for the movie "Blade Runner".

- An instance of class "Version" in the BBC ontology is created. This instance stores the attributes "start" and "stop" that are present in every “<programme>” XMLTV element. Also, this instance is associated with the one created in the previous step by using the property “po:version” in the class “Program”. Before including it in the knowledge base, XMLTV2OWL looks again for possible collisions between individuals. If some duplicates are found, only the most recent instance is maintained. Figure 12 shows the corresponding RDF code for the "Blade Runner" example:



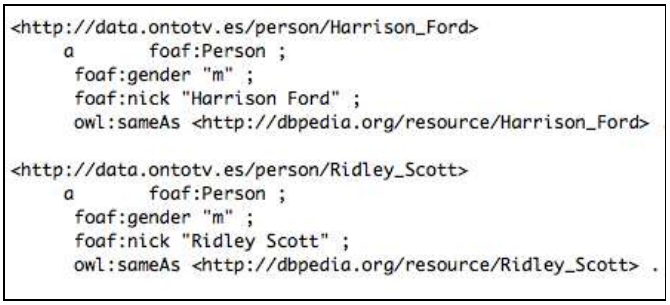
```
<http://data.ontotv.es/version/2dfe2b07-1df3-4111-8f2d-70adde8d2097>
      po:sound_format "urn:ard:tva:metadata:cs:ARDFormatCS:2008:3.2" ;
      po:subtitle_language "Spanish" ;
      po:aspect_ratio "urn:ard:tva:metadata:cs:ARDFormatCS:2008:1.24" ;
      po:time [ a       event:Interval ;
                event:end "2012-07-23T18:48:29.959Z"^^xsd:dateTime ;
                event:start "2011-11-15T20:45:00Z"^^xsd:dateTime
              ] .
```

Fig. 12. Instance of the class “Version” for the movie “Blade Runner”.

Interlinking with other Linked Data Datasets

The Linked Data methodology put special emphasis on the need of establishing links between data fragments that are semantically related in some way [10]. This makes possible to browse the entire knowledge, jumping from one concept to another. For this reason, OntoTV executes some special processes that try to match the local instances available in the RDF base with other similar individuals from external datasets. This way it is possible to create links between OntoTV’s triples and other resources in the Web of Data:

- Links to the DBpedia dataset: DPpedia is considered to be the core of the Web of Data cloud. It contains information about any domain, so it has become a reference dataset in the Linked Data research field. Here, the instance matching process has been performed by applying simple lexical similarity functions over the textual attributes in the classes "Service" and “Programme" (like for example, "producer "," director "," actor ", etc.) Figure 13 shows examples of such links:



```
<http://data.ontotv.es/person/Harrison_Ford>
     a       foaf:Person ;
      foaf:gender "m" ;
      foaf:nick "Harrison Ford" ;
      owl:sameAs <http://dbpedia.org/resource/Harrison_Ford>

<http://data.ontotv.es/person/Ridley_Scott>
     a       foaf:Person ;
      foaf:gender "m" ;
      foaf:nick "Ridley Scott" ;
      owl:sameAs <http://dbpedia.org/resource/Ridley_Scott> .
```

Fig. 13. Persons and their corresponding links to instances in DBpedia.

- Links to the “Geonames” dataset. Certain individuals in the knowledge base refer to geographical places. In these cases, OntoTV checks whether these instances are geographically equivalent to others in “Geonames” dataset, which contains over eight million names of places that are available for search.
- Links to “LinkedMDB” dataset. Although this dataset still contains only a few records of certain movies, it will become the reference dataset for information about films in the Web of Data. For this reason, OntoTV will try to identify possible alignments between the local instances and the ones stored in this dataset, especially for some attributes like "director", "actor" and "film". Again, string similarity functions on the titles will be applied.

The module "LD Publishing" (see Figure 14) is responsible of accessing to external datasets in order to execute all these the instance matching processes. As shown in Figure 13, the links found with this method are expressed in the form of <owl:sameas> triplets.

Finally, it is neccesary to mention the existence of some data publishing frameworks such as Openlink Virtuoso[4], which stores RDF triples, generates HTML pages containing the data (so they can be browsed online), and creates a SPARQL endpoint where this kind of queries can be executed. However, this possibility has not been addressed in this research.

C. OntoTV after applying Linked Data methodologies

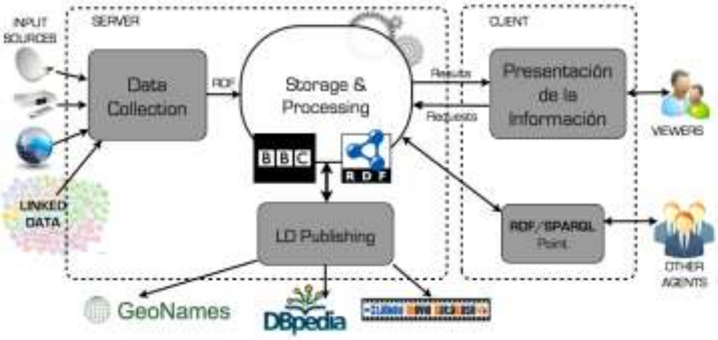


Fig. 14. OntoTV schema after applying Linked Data principles.

Figure 14 shows the changes occurred in OntoTV after the application of Linked Data methodologies. On the one hand, the “Collection Data” module adds a new source: The Web of Data cloud. Furthermore, the “Storage and Processing” module now contains the RDF information represented according to the BCC ontology and conveniently linked to other external dataset. Regarding the way the information is presented to the user, not only the OntoTV’s clients have access to the data, but also all the agents who are able to access resources in the Linked Data cloud.

IV. Conclusions

Nowadays viewers have to do a considerable effort every time they want to find, access or compare television programs, due to the great variety of them available in the different platforms. OntoTV system has been designed for giving a solution to this problem. It uses advanced data collection techniques and ontology-based representation methods.

[4] http://virtuoso.openlinksw.com/

However, the previous version of OntoTV accessed to non-structured data sources, so the collection mechanisms had to be fully customized for each considered resource. Furthermore, only the clients that were compatible with OntoTV's information interchange protocol could access the data stored in the system.

This paper describes how Linked Data principles have been applied in OntoTV system in order to solve these problems. On the one hand, Linked Data resources have been accessed to complete the information about movies available in the system; on the other hand, a mechanism for publishing information about television content and channels has been designed.

Regarding the *data consumption*, it has been probed that the data collected from Linked Data sources has been useful to enrich the scarce content descriptions originally sent by the providers. As the considered sources are compliant with Linked Data principles, the data extracted from them is well structured and includes semantic links between concepts that are not present in classic HTML links. In this situation it is straightforward to extract the desired information, not only in the case of film description, but also for other types of content. Furthermore, the decision of accessing a resource like SIG.MA, which automatically integrates many others Linked Data sources, has provided advantages over the crawling strategies and the execution of SPARQL queries. As the information comes from various sources, it is possible to find movie descriptions for almost any title.

Regarding the *data publishing*, information can now be accessed not only by OntoTV's clients, but also by any other agent able to consume Linked Data resources. And all of this without having to implement a specific logic for message interchange or interpret particular formats like XMLTV. Also, the decision of using the BBC's ontology, which is widely agreed in the television domain, has been very appropriate because the information collected by OntoTV system becomes available in the Web of Data in a more standard way.

Despite the improvements achieved, it is still necessary to continue enhancing the processes that transform the collected XMLTV data into instances of the BBC ontology. Other future research line is trying to incorporate better mechanisms for finding inconsistences in the data and detecting instance collisions, especially when adding instances of the programs. Finally, the algorithms for aligning information with LinkedMDB and DBpedia datasets can be also improved because until the moment they only use simple lexical comparisons.

In conclusion, the application of Linked Data methodologies has been very beneficial for improving the performance of systems that consume and publish data, like OntoTV does. With these information management strategies applied to the television domain, viewers will have access to a more accurate, complete and useful information.

V. Acknowledgment

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] Redondo-Garcia, J.L, Valiente-Rocha, P., Lozano-Tello, A.: Ontology-based system for content management in Digital Television. CISTI, pp. 277-–283, (2010).

[2] Berners-Lee, T.: Linked Data. International Journal on Semantic Web and Information Systems, vol. 4, no. 2, W3C (2006).

[3] Li Yujian and Liu Bo.: A normalized levenshtein distance metric. IEEE Trans. Pattern Analysis and Machine Intelligence, pp. 1091–1095, (2007).

[4] Redondo-Garcia, J.L., Lozano-Tello, A.: Recolección de Datos sobre Contenidos Televisivos en el Sistema OntoTV. CISTI, Accepted in press (2011).

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

[6] Heath, T., Bizer, C.: Linked Data: Evolving the Web into a Global Data Space. Synthesis Lectures on the Semantic Web: Theory and Technology, Morgan & Claypool (2011).

[7] Schopman, B., Brickley, D., Aroyo, L., van Aart, C., Buser, V., Siebes, R.: NoTube: making the Web part of personalised TV. Proceedings of the WebSci10 (2010).

[8] Buser, V.: NoTube: experimenting with Linked Data to improve user experience, Summer School on Multimedia Semantics, Amsterdam, 3 September 2010

[9] Y. Blanco-Fernández, et al, "Exploiting synergies between semantic reasoning and personalization strategies in intelligent recommender systems" Journal of Systems and Software, vol.81, pp. 2371-2385, 2008.

[10] Volz, J., Bizer, T., Gaedke, M., Kobilarov, G.: Discovering and Maintaining Links on the Web of Data. International Semantic Web Conference (ISWC2009), Westfields, USA, pp. 650—665, (2009).

José Luis Redondo-Garcia is a PhD student in the Multimedia Modeling and Interaction group at the EURECOM institute, in France. His research interests include Semantic Web, Knowledge Representation, Interactive Digital Television Applications, and Multimedia Technologies. Jose Luis has a MS in Computer Science and a BS in Computer Science and Systems from the University of Extremadura (Spain), where he also worked as research assistant for the QUERCUS Software Engineering Group.

Vicente Botón-Fernández is a PhD student at Computer Science in the QUERCUS Software Engineering Group, University of Extremadura, Spain. His research interests include Ambient Intelligence, Domotics, Data Mining, Machine Learning and the use of ontologies and semantic rules applied to the previous areas. He has a MS in Computer Science and a BS in Computer Science from the University of Extremadura (Spain).

Adolfo Lozano-Tello is teaching/research assistant professor of Computer Languages and Systems at the University of Extremadura from 1996 and Director of the Telefonica Chair in this university from 2009. He received his BS in Computer Science from the University of Granada, Spain (1993). He is a Ph.D. (2002) at Computer Science, University of Extremadura, Spain. He has published more than 80 papers on the above issues on Software Engineering and Knowledge Engineering. He belongs to several related projects.