NoTube

Networks and ontologies for the transformation and unification of broadcasting and the Internet

FP7 – 231761

D6.3 NoTube Integrated System
2nd Prototype

Coordinator: A. Conconi, F. Cattaneo (TXT)
With contributions from:
R. Del Pero, L. Vignaroli, F. Negro (RAI)
P. Altendorf, R. Zimmermann, M. Riethmayer (IRT)
S. Dietze, N.J.L. Benn (OU)
P. Mihaylov (OT)
M. Minno, F. Grucci (AS)
L. Miller (BBC)
C. Van Aart (VUA)

Quality Assessor: Alex Conconi
Quality Controller: Lyndon Nixon

<table>
<thead>
<tr>
<th>Document Identifier:</th>
<th>NoTube/2009/D6.3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Version:</td>
<td>1.0</td>
</tr>
<tr>
<td>Date:</td>
<td>23/12/2010</td>
</tr>
<tr>
<td>State:</td>
<td>Final</td>
</tr>
<tr>
<td>Distribution:</td>
<td>PU</td>
</tr>
</tbody>
</table>
EXECUTIVE SUMMARY

This document represents a companion to the second release of the NoTube integrated services platform developed in WP6 and also integrates the description of the integrated prototypes developed in WP7 and due M23. Its purpose is to provide an overview of the integrated services platform including the real-world integration approach, as well as individual achievements in terms of prototypes/services integration for the three use cases.

This deliverable provides also a high level view of the developed and exploited services for the main categories envisaged in NoTube, in each prototype, and according to the previously occurred design phase.

The NoTube services platform is the result of a design and implementation work that accommodates requirements originating from several different stakeholders (e.g. service developers, use case designers, broadcasters, end users) and integrates heterogeneous technologies into a unified reference framework. This process required an effort from both a technical design and coordination perspective.

NoTube services are clustered in three main areas: user services, metadata services and content services. User services deal with user credentials management, user profiling, security and privacy. Metadata services implement metadata models, conversion and enrichment processes, as well as recommendations. Content services concern the actual delivery of content, and include services such as video transcoding and cropping and audio harmonization.

The main aim of this document is to represent a technical reference useful for software architects and developers who integrate NoTube services into their applications, in part or as a whole. It provides broad guidelines and an integration vision, pointing out main functional blocks and interconnections, typical workflows as well as potential issues to be considered such as integration of legacy systems, Service Provider-Home Ambient interdependency, security and privacy preservation.

The internal evaluation plan in terms of actors, goals, target groups and methodology has been presented as well focusing on field-specific issues like legacy systems and data/privacy protection, as part of the very next steps in the WP6 activities.
DOCUMENT INFORMATION

<table>
<thead>
<tr>
<th>IST Project Number</th>
<th>FP7 - 231761</th>
<th>Acronym</th>
<th>NoTube</th>
</tr>
</thead>
<tbody>
<tr>
<td>Full Title</td>
<td>Networks and ontologies for the transformation and unification of broadcasting and the Internet</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Project URL</td>
<td><a href="http://www.notube.eu/">http://www.notube.eu/</a></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Document URL</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EU Project Officer</td>
<td>Leonhard Maqua</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Deliverable Number</th>
<th>6.3</th>
<th>Title</th>
<th>NoTube Integrated System 2nd prototype</th>
</tr>
</thead>
<tbody>
<tr>
<td>Work Package Number</td>
<td>6</td>
<td>Title</td>
<td>Architecture for personalized TV content</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Date of Delivery</th>
<th>Contractual</th>
<th>Actual</th>
</tr>
</thead>
<tbody>
<tr>
<td>Status</td>
<td>version 1.0</td>
<td>Final</td>
</tr>
<tr>
<td>Nature</td>
<td>prototype ✔ report □ dissemination □</td>
<td></td>
</tr>
<tr>
<td>Dissemination level</td>
<td>public ✔ consortium □</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Authors (Partner)</th>
<th>TXT, RAI, IRT, OU, OT, AS, BBC, VUA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Responsible Author</td>
<td>Alex Conconi</td>
</tr>
<tr>
<td>Name</td>
<td>E-mail: <a href="mailto:alex.conconi@txt.it">alex.conconi@txt.it</a></td>
</tr>
<tr>
<td>Partner</td>
<td>Phone: +39.0225771826</td>
</tr>
</tbody>
</table>

Abstract (for dissemination)
This document presents NoTube platform from the integration point of view, further demonstrated by the three implementation stages of the three planned application scenarios. The main aim of this document is to provide the broad lines of the integration vision including legacy systems, Service Provider-Home Ambient interdependency, security and privacy preservation. An evaluation plan for the abovementioned topics is also included. The description of the individual prototypes is not part of the integrated system reporting, but instead of the WP7.a/b/c, as the core activity of each foreseen test case.

Keywords
Prototype, Services, Integration

Version Log
<table>
<thead>
<tr>
<th>Issue Date</th>
<th>Rev. No.</th>
<th>Author</th>
<th>Change</th>
</tr>
</thead>
<tbody>
<tr>
<td>28/10/2010</td>
<td>0.1</td>
<td>F.Cattaneo, A. Conconi</td>
<td>ToC</td>
</tr>
<tr>
<td>08/12/2010</td>
<td>0.2</td>
<td>F.Cattaneo, A. Conconi</td>
<td>First draft integrating contributions</td>
</tr>
<tr>
<td>23/12/2010</td>
<td>1.0</td>
<td>F.Cattaneo, A. Conconi</td>
<td>Final</td>
</tr>
</tbody>
</table>
## Project Consortium Information

<table>
<thead>
<tr>
<th>Participants</th>
<th>Contact</th>
</tr>
</thead>
</table>
| Vrije Universiteit Amsterdam                      | Guus Schreiber  
  Phone: +31 20 598 7739/7718  
  Email: schreiber@cs.vu.nl |
| British Broadcasting Corporation                 | Libby Miller  
  Phone: +44 787 65 65 561  
  Email: libby.miller@bbc.co.uk |
| Asemantics SRL Uninominale                        | Alberto Reggiori  
  Phone: +39 0639 7510 78  
  Email: alberto@asemantics.com |
| Engin Medya Hizmetleri A.S.                       | Ron van der Heiden  
  Phone: +31 6 2003 2006  
  Email: ron@engin.tv |
| Institut fuer Rundfunktechnik GmbH                 | Christoph Dosch  
  Phone: +49 89 32399 349  
  Email: dosch@irt.de |
| Ontotext AD                                       | Atanas Kiryakov  
  Phone: +35 928 091 565  
  Email: naso@sirma.bg |
| Open University                                   | John Domingue  
  Phone: +44 1908 655 014  
  Email: j.b.domingue@open.ac.uk |
| RAI Radiotelevisione Italiana SPA                  | Alberto Morello  
  Phone: +39 011 810 31 07  
  Email: a.morello@rai.it |
| Semantic Technology Institute International        | Lyndon Nixon  
  Phone: +43 1 23 64 002  
  Email: lyndon.nixon@sti2.org |
| Stoneroos B.V.                                    | Annelies Kaptein  
  Phone: +31 35 628 47 22  
  Email: annelies.kaptein@stoneroos |
| Thomson Grass Valley France SA                    | Raoul Monnier  
  Phone: +33 2 99 27 30 57  
  Email: raoul.monnier@thomson.nett |
| TXT Polymedia SPA                                 | Sergio Gusmeroli  
  Phone: +39 02 2577 1310  
  Email: sergio.gusmeroli@txtgroup.com |
| KT Corporation                                    | Myoung-Wan Koo  
  Phone: +82 2 526 6347  
  Email: mskim@kt.co.kr |
## Table of Contents

### TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>INTRODUCTION</td>
<td>9</td>
</tr>
<tr>
<td>1.1. DOCUMENT SCOPE</td>
<td>9</td>
</tr>
<tr>
<td>1.2. DOCUMENT OUTLINE</td>
<td>9</td>
</tr>
<tr>
<td>2. NOTUBE PLATFORM</td>
<td>10</td>
</tr>
<tr>
<td>2.1. ARCHITECTURE OVERVIEW</td>
<td>10</td>
</tr>
<tr>
<td>2.2. INTEGRATION APPROACH</td>
<td>11</td>
</tr>
<tr>
<td>2.3. PROVIDER-HOME SYSTEM INTEGRATION</td>
<td>13</td>
</tr>
<tr>
<td>2.4. INTERFACE WITH CONTENT MANAGEMENT SYSTEM</td>
<td>13</td>
</tr>
<tr>
<td>2.4.1. CMS used in Broadcasting</td>
<td>14</td>
</tr>
<tr>
<td>2.4.2. Interfacing Legacy CMS with Notube</td>
<td>15</td>
</tr>
<tr>
<td>2.5. USER PROFILE MANAGEMENT</td>
<td>16</td>
</tr>
<tr>
<td>2.6. RECOMMENDATION AND ENRICHMENT</td>
<td>17</td>
</tr>
<tr>
<td>2.7. DATA PROTECTION</td>
<td>19</td>
</tr>
<tr>
<td>3. USE-CASES INTEGRATED PROTOTYPES OVERVIEW</td>
<td>21</td>
</tr>
<tr>
<td>3.1. PERSONALIZED SEMANTIC NEWS</td>
<td>21</td>
</tr>
<tr>
<td>3.1.1. Overview</td>
<td>21</td>
</tr>
<tr>
<td>3.1.2. Integration</td>
<td>24</td>
</tr>
<tr>
<td>3.2. PERSONALIZED TV GUIDE WITH ADAPTIVE ADVERTISING</td>
<td>24</td>
</tr>
<tr>
<td>3.2.1. Overview</td>
<td>24</td>
</tr>
<tr>
<td>3.2.2. Integration</td>
<td>27</td>
</tr>
<tr>
<td>3.2.3. Retrieving enriched EPG data</td>
<td>27</td>
</tr>
<tr>
<td>3.2.4. Profile-based TV programme recommendation</td>
<td>28</td>
</tr>
<tr>
<td>3.3. INTERNET TV IN THE SOCIAL WEB</td>
<td>28</td>
</tr>
<tr>
<td>3.3.1. Overview</td>
<td>28</td>
</tr>
<tr>
<td>3.3.1.1. Sub-Prototype 1</td>
<td>29</td>
</tr>
<tr>
<td>3.3.1.2. Sub-Prototype 2</td>
<td>29</td>
</tr>
<tr>
<td>3.3.1.3. Sub-Prototype 3</td>
<td>30</td>
</tr>
<tr>
<td>3.3.2. Integration</td>
<td>30</td>
</tr>
<tr>
<td>4. NOTUBE PLATFORM SERVICES</td>
<td>32</td>
</tr>
<tr>
<td>4.1. OVERVIEW</td>
<td>32</td>
</tr>
<tr>
<td>4.2. SERVICES DESCRIPTION AND ANNOTATION</td>
<td>32</td>
</tr>
<tr>
<td>4.2.1. Lightweight Services Annotation: the Linked Services approach</td>
<td>33</td>
</tr>
<tr>
<td>4.2.2. Service Annotation and Integration via SmartLink</td>
<td>33</td>
</tr>
<tr>
<td>4.3. USER PROFILE SERVICES</td>
<td>34</td>
</tr>
<tr>
<td>4.3.1. Overview</td>
<td>34</td>
</tr>
<tr>
<td>4.3.2. Use Cases Connection</td>
<td>35</td>
</tr>
<tr>
<td>4.3.3. Data Sets Connection</td>
<td>36</td>
</tr>
<tr>
<td>4.3.4. Integration</td>
<td>36</td>
</tr>
<tr>
<td>4.4. METADATA SERVICES</td>
<td>36</td>
</tr>
<tr>
<td>4.4.1. Metadata Conversion Services</td>
<td>36</td>
</tr>
<tr>
<td>4.4.1.1. Overview</td>
<td>36</td>
</tr>
<tr>
<td>4.4.1.2. Use Cases Connection</td>
<td>37</td>
</tr>
<tr>
<td>Use case 7.a</td>
<td>37</td>
</tr>
<tr>
<td>Use case 7.b</td>
<td>37</td>
</tr>
<tr>
<td>Use case 7.c</td>
<td>37</td>
</tr>
<tr>
<td>4.4.1.3. Data Sets Connection</td>
<td>38</td>
</tr>
</tbody>
</table>
6. CONCLUSIONS

5. EVALUATION PLANNING

5.1. INTERFACE WITH LEGACY CMS

5.1.1. Goal

5.1.2. What, Where and How

5.2. SECURITY AND PRIVACY PRESERVATION

5.2.1. Goal

5.2.2. What, Where and How

5.3. ROADMAP

4.5. CONTENT SERVICES

4.5.1. Video Processing Services

4.5.2. Audio Processing Services

4.5.3. Personal Video Recorder and Media Storage services

4.5.4. Integration

4.4.3. Enrichment Services

4.4.3.1. Overview

4.4.3.2. Use Cases Connection

4.4.3.3. Data Sets Connection

4.4.3.4. Integration

4.4.2. Recommendation Services

4.4.2.1. Overview

4.4.2.2. Use Cases Connection

4.4.2.3. Data Sets Connection

4.4.2.4. Integration

4.4.1. PRESERVATION

4.4.1.1. Goal

4.4.1.2. Use cases connection

4.4.1.3. Data Sets Connection

4.4.1.4. Integration

4.3. CMS

4.3.1. Overview

4.3.2. Use Cases Connection

4.3.3. Data Sets Connection

4.3.4. Integration

4.2. SELECTION SERVICES

4.2.1. Overview

4.2.2. Use Cases Connection

4.2.3. Data Sets Connection

4.2.4. Integration

4.1. PERSONAL PREFERENCE SERVICES

4.1.1. Overview

4.1.2. Use Cases Connection

4.1.3. Data Sets Connection

4.1.4. Integration

Use case 7.b

Use case 7.a

4.4.4. Audio Processing

4.4.4.1. Overview

4.4.4.2. Use Cases Connection

4.4.4.3. Data Sets Connection

4.4.4.4. Integration

4.3.3. Overview

4.3.3.1. Personal Video Recorder and Media Storage services

4.3.3.2. Integration

4.3.3.3. Data Sets Connection

4.3.3.4. Use cases connection

4.2.2. Overview

4.2.2.1. Audio Processing Services

4.2.2.2. Integration

4.2.2.3. Use Cases Connection

4.2.2.4. Data Sets Connection

5.1.1. Overview

5.1.1.1. Interface with Legacy CMS

5.1.1.2. Goal

5.1.1.3. What, Where and How

5.1.1.4. Interface with Legacy CMS

5.1.1.5. Goal

5.1.1.6. What, Where and How

Page 6 of 49
List of Figures

Figure 1 – NoTube Platform Integration Architecture .......................................................... 10
Figure 2 - Composition of media asset .................................................................................. 14
Figure 3 – User Management Integration - Authentication ...................................................... 16
Figure 4 – User Management Integration – Application Logic, Social Networks .................. 17
Figure 5 – Recommender Direct Integration ...................................................................... 18
Figure 6 – Enrichment Broker Integration .......................................................................... 19
Figure 7 – WP7.a Service Provider Integration Diagram ...................................................... 22
Figure 8 – WP7.a Home Ambient Integration Diagram .......................................................... 22
Figure 9 – WP7.a : Personalised Newscast Playback ............................................................. 23
Figure 10 – WP7.a : Dynamic Links to External News ........................................................... 23
Figure 11 – WP7.b : Storyboard 1/3 .................................................................................... 25
Figure 12 – WP7.b : Storyboard 2/3 .................................................................................... 26
Figure 13 – WP7.b : Storyboard 3/3 .................................................................................... 26
Figure 14 – WP7.b Integration Diagram .............................................................................. 27
Figure 15 – WP7.c : On-demand Video Browser Showing Programme-To-Programme Recommendations .................................................. 29
Figure 16 – WP7.c : Real Time Non-Text Interactive Voting Application ............................ 30
Figure 17 – WP7.c Integration Diagram ............................................................................. 31
Figure 18 – Linked Open Data Cloud Snapshot .................................................................. 43
# List of Acronyms

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AMI</td>
<td>Ambient Intelligence</td>
</tr>
<tr>
<td>EPG</td>
<td>Electronic Program Guide</td>
</tr>
<tr>
<td>NIC</td>
<td>News Item Container</td>
</tr>
<tr>
<td>LOD</td>
<td>Linked Open Data</td>
</tr>
<tr>
<td>OWL</td>
<td>Web Ontology Language</td>
</tr>
<tr>
<td>PDA</td>
<td>Personal Digital Assistant</td>
</tr>
<tr>
<td>RDF</td>
<td>Resource Description Framework</td>
</tr>
<tr>
<td>SEE</td>
<td>Semantic Execution Environment</td>
</tr>
<tr>
<td>SKOS</td>
<td>Simple Knowledge Organisation System</td>
</tr>
<tr>
<td>SOA</td>
<td>Service-Oriented Architecture</td>
</tr>
<tr>
<td>SOAP</td>
<td>Simple Object Access Protocol</td>
</tr>
<tr>
<td>SPARQL</td>
<td>Simple Protocol and RDF Query Language</td>
</tr>
<tr>
<td>STB</td>
<td>Set-Top Box</td>
</tr>
<tr>
<td>SWS</td>
<td>Semantic Web Services</td>
</tr>
<tr>
<td>TV-A</td>
<td>TV-Anytime</td>
</tr>
<tr>
<td>VOD</td>
<td>Video On Demand</td>
</tr>
<tr>
<td>WSDL</td>
<td>Web Services Description Language</td>
</tr>
<tr>
<td>WSMO</td>
<td>Web Service Modeling Ontology</td>
</tr>
<tr>
<td>XML</td>
<td>eXtensible Markup Language</td>
</tr>
<tr>
<td>TV-A</td>
<td>TV-Anytime</td>
</tr>
</tbody>
</table>
1. Introduction

1.1. Document scope

The aim of this document is to present an overview of the NoTube integrated platform to support the developers during the integration tasks to be carried out during the project. While deliverable D6.1 is focused on the general architecture, applied to the three foreseen use cases, D6.3 describes the NoTube platform in terms of available services.

This document represents a companion to the second release of the NoTube integrated services platform developed in WP6 and also integrates the description of the integrated prototypes developed in WP7 and due M23. It provides a broad description of the NoTube platform in terms of overall capabilities, including currently available as well as planned services. NoTube services have been implemented following a generalisation process on requirements that emerged from individual application scenarios.

This document is also the basis for forthcoming development activities since it provides information on typical processes and workflows that are part of a NoTube-based application. Detailed technical information such input and output parameters, software dependencies, and entry points of NoTube services is now maintained through the dedicated on line tool SmartLink.

1.2. Document outline

This document is composed of four main sections: Chapter 2 presents the NoTube platform integrated architecture, focusing on WP6 envisaged tasks and general workflows. Chapter 3 provides a synopsis of the NoTube prototypes related to the three planned use cases with emphasis on current integration achievements and relationship with the NoTube general reference architecture; further details about the prototypes are provided in WP7 deliverables. Chapter 4 is about NoTube individual services clustered in their main categories, namely: user profile management, metadata services and content services. Finally, Chapter 5 presents the evaluation planning for what concerning WP6 main goals.
2. NoTube Platform

2.1. Architecture Overview

The following picture provides an updated overview of the NoTube platform integration architecture reflecting the 2nd prototypes’ design.

![NoTube Platform Integration Architecture](image)

*Figure 1 – NoTube Platform Integration Architecture*

It is easy to refer to the general architecture depicted in D6.1 by means of colors, each one representing a service category. This color-coding approach has been adopted in the previous WP6 deliverables in order to effectively map the three prototypes’ internal modules onto the general platform architecture. Similarly the diagram above provides the overview of the platform’s integration architecture through the same color paradigm:
**Front-End** – Yellow color indicates every kind of GUI component interacting with the end-user, including multichannel delivery and the related devices (PC, Mobile, STB, etc.). It’s part of the Ambient Intelligence.

**Application Logic** – Includes the scenarios-specific implemented logic (Business Logic) and any potentially adopted Legacy System (i.e. : iFanzy in WP7.b)

**User Profile** – Refers to the « NoTube User », including the concept of user identity (local or remote, implemented through various standards such as OpenID) static details (i.e. : name, surname, age, sex, etc.), user activities collected from the chosen social networks, personal interests and the whole set of techniques adopted to guarantee the privacy preservation (i.e.: OAUTH)

**Metadata-Oriented Modules** – Refers to several NoTube services that can be grouped in : recommendation services, enrichment services and metadata conversion services. Such tools are connected internal/external repositories (i.e. : Lupedia, DBPedia, IMDB, etc.) as the main working datasets.

**Content-Oriented Modules** – Refers to the NoTube's services aimed at ingesting, processing and stream multimedia files (both audio and video). Similarly to the metadata-oriented modules, content-oriented modules are connected to internal/external contents (i.e. : broadcaster’s CMS, LOD MDB – Linked Open Data Media DB, FilmCrave1, etc.)

**Services Annotation and Business Process Choreography** – It deals with the semantic brokering techniques adopted to annotate services, compose and orchestrate them to implement specific goals in a seamless way

### 2.2. Integration Approach

The integration of the three prototypes has been implemented in the scope of WP7 following a top-down approach basing on the following main actors:

- The NoTube user profile seamlessly connected to personal social aspects
- The available platform services providing: metadata conversions, recommendations, enrichment, content ingestion, processing and streaming
- The specific requirements coming from the individual application scenarios focusing on different topics (News, ads, social activities)
- The available semantic middleware capable of developing and providing services orchestration for specific goals

Each technical work package (WP1-5), in the scope of its internal R&D activities, worked towards the creation of a unified platform by providing a set of flexible solutions enabling different integration possibilities. In particular:

- Providing REST-based endpoints for the available services for direct invocation
- Annotating services using RDF in order to make them available to the semantic broker
- Defining a commonly agreed format for the input and output parameters with respect to a particular service category (i.e.: the recommenders, the enrichers, etc.)

---

1 [http://en.wikipedia.org/wiki/FilmCrave](http://en.wikipedia.org/wiki/FilmCrave)
- Designing metadata conversion services on top of the elaborated models tailored to the broadcaster’s environments and needs
- Quickly developing custom functionalities to enable the prototyping of new ideas and features

In the same way WP7a, b, c leaders iteratively provided a list of updated formal and informal requirements that have been translated by the respective technical leaders, upon the WP6 supervision, into development actions supporting the described scenario and the requested functionalities.

In order to adhere to the designed platform, this activity triggered a lot of interactions between the use cases and the technical leaders driving the integration in the proper direction.

For the above reasons and due to the sometimes specific requirements of the three use cases the final technical choice for the services integration has been left to the prototypes’ developers by always providing two options:

1. Leverage on the Business Process Choreography layer
2. Directly invoking the NoTube’s platform services

The first solution is the most transparent for the application developer perspective since it provides a single entry point for accessing the NoTube services without having to know them all in detail, giving at the same time the possibility to quickly achieve the final goal.

The drawback from a user perspective could be the slower performance produced by the semantic engine overhead, although this heavily depends on the orchestration workflow required by the application.

On the semantic broker side, this approach leads to the need of collecting specific requirements for the requested goal, design the proper orchestration workflow, perform the semantic annotation of the involved services and develop the entry point.

The second solution envisages the direct invocation of the requested services from the application logic. The developer must have a clear understanding of the specific service input and output parameters and formats as well as the communication protocol adopted (REST-based, as a commonly agreed architecture choice).

There’s no orchestration here although there could be some contexts where the overhead introduced by the semantic layer is not justified.

As a final consideration it is worth noticing that despite the level of freedom provided to the application developers for low level integration choices, the overall picture provided in the Figure 1 is the commonly shared approach when implementing the different solutions. In other words, every application scenario (present and future) is envisaged:

- to have a unique entry point for the user profile management, properly implementing the privacy protection mechanisms, and giving access to user details, preferences, interests, etc.
- to leverage on a set of platform services in the preferred way (directly or through the broker)
  - to get recommendations, enriched contents, etc. transparently taking care of external repositories and different data formats via metadata conversion services
  - to ingest, process and stream contents regardless of the final output channel

At the same time, it is also expected:

- to internally implement the business logic specific for the application, including the data exchanges with potentially existing CMS and repositories, even though the latter are fully enabled to be extended and empowered with NoTube capabilities.
to implement and customise the front-end for the desired application, including the support of different output devices, even though they are enabled by the platform itself to access streamed multimedia contents (the Home Ambient back-end, in this particular scenario acts as a bridge).

2.3. **Provider-Home System Integration**

The integration between the Service Provider and the Home Ambient (we will call them SP and HA from now on to shorten up) has been performed on a case-by-case basis, since heavily depending on the application scenario's focus. In particular, referring to Figure 1, this aspect is even more true when thinking about the Application Logic block.

Analysing the various broadcaster's internal environment in the scope of the respective use cases, the following scenarios emerged.

- **WP7.a** Personalized Semantic News has a heavy focus on both Newscasts and individual News items, owned by the Service Provider and then dispatched in various ways to the Home Ambient. Here the “Legacy System” (named ANTS) is part of RAI’s internal network and then the logic behind the News-based contents preparation and metadata relies there. The Home Ambient acts, very roughly, like a “smart cache”, aimed at ingesting contents tailored to the specifically considered user profile. Then it also embeds the logic that provides the front-end (specific of the final output device) with the prepared, personalised output. The latter component can be identified as the “Business Logic” block in the integration diagram. The integration between the SP and the HA sides leverages on XML-based feeds exposed by the Provider in order to notify about upcoming Newscasts (on IP or air) as well as available prepared metadata. The Home Ambient, in turn, is capable of setting up the proper ingestion mechanism, create a connection between the ingested media files and the available metadata, prepare the personalised Newscasts in light of the current user profile and finally provide the front-end layer with the final output.

- **WP7.b** Personalized TV Guide with Adaptive Advertising is focused on TV programmes and advertising, managed by the proprietary legacy system named iFanzy. The latter acts, at Service Provider side, as the Application Logic back-end that, in turn, is capable of integrating additional features provided by NoTube thanks to the Semantic Broker. The Home Ambient side is composed by the iFanzy's client counterpart, capable of running on mobile devices (i.e.: the popular iPhone) as well as on standard browsers. The integration is then limited to the communications between the server and the client components, transparently managed by the iFanzy legacy system.

- **WP7.c** Internet TV in the Social Web deals mainly with user's social activities in order to provide the end user with interesting proposals about media entertainment. Thus the SP is heavily based on the user profile module as well as the recommendation and the enrichment services provided by NoTube. The HA is somewhat similar to WP7.b in the sense that it doesn't include much logic but, instead, focuses on the integration of common multimedia devices aimed at consuming the final experience (i.e.: mobile devices, pads, set-top boxes, etc.). The integration between SP and HA, then, is mainly a low level technical challenge dealing with connections protocols or remote connectivity and control mechanisms focused on messaging and real-time communication (i.e.: XMPP)

2.4. **Interface with Content Management System**

In the past years, TV production has more and more migrated from tape-based to file-based production workflows. In a file-based TV production environment, content management systems (CMS) play an essential role, both in the actual production of programmes and in
the content distribution. The CMS is the place where the media content is stored and managed.

In broadcasting, content is defined as essence and metadata (see Figure 2). The essence is the media itself (e.g. video, audio, graphic and text files) while the metadata is additional data associated to the essence which holds technical information about the essence and also describes the content. Thus, the metadata is essential to manage the essence in the CMS for TV production purposes.

If the content is extended by rights information this is referred to as an asset (see Figure 2). However, since rights are often seen as delicate information they are typically managed in separate databases. If all data is contained in one system, this is referred to as a Media Asset Management System (MAM).

![Figure 2 - Composition of media asset](image)

Other systems involved in the TV production process are linked to the CMS (or MAM), e.g. play-out servers or archives. These systems might operate with metadata in other formats so that the metadata consumed or provided needs to be mapped. Such metadata conversions (or translations) are typically necessary when content is exchanged between the numerous components that are involved in the production process.

While the essence file is often stored separately from the metadata, essence and metadata can be put into one single file for the exchange or distribution of content using container formats. AVI or Quicktime are container formats often used for distribution to the end consumer. In the professional media production domain MXF (Material Exchange Format) is the most common format for content exchange. MXF is standardised by SMPTE\(^2\) and provides also a basic metadata scheme (DMS-1) as well as registries for proprietary metadata. Even though MXF is intended for the exchange of content, it can nowadays also often be found in TV archives.

### 2.4.1. CMS used in Broadcasting

**CMS for Production**

In the past, CMS in broadcasting have primarily been used for the actual production of TV programmes. Many different manufacturers offer CMS solutions for professional broadcasters. CMS that are used among German Public Service Broadcasters, but that are also commonly known and internationally used, are:

- **Interplay Media Asset Manager by AVID\(^3\)** - This is a product coming from the TV archive side and formerly known as Media Archive by Blue Order\(^4\) which was

---

\(^2\) SMPTE 377M: The MXF File Format Specification  
\(^3\) http://avid.com  
\(^4\) www.blueorder.com
taken over by AVID at the end of 2009. It is now also focusing on the integration of production systems.

- **VPMS by S4M** - In contrast to Media Archive, this is a product coming from the TV production side, now also focusing on archiving.
- **Viz Ardome by Vizrt** - A product originally coming from Swedish manufacturer Ardendo, Viz Ardome complements the Vizrt product line with a focus on real-time graphics for broadcasting by a content management and playout system.

CMS in file-based TV production have not been around for too long, but in the context of NoTube, they can be seen as legacy CMS. However, a shift from monolithic structures as they are typical for the systems mentioned above to flexible service oriented architectures can currently be observed. Open Source CMS for professional broadcast applications are not known at the time of reporting.

**CMS for Online Distribution**

Recently, more and more dedicated CMS for the distribution of TV content using online applications like catch-up TV have been set up and used at broadcasters to meet the increasing demand from the consumers. Mostly used at German PSBs is Sophora from subshell\(^7\). Others used are e.g. TeamSite by Autonomy\(^8\) and CoreMedia CMS by CoreMedia\(^9\).

**Open Source WCMS for Online Distribution**

In general, CMS are also used for Web applications, i.e. to create, manage and control Web content. In this context they are called **Web Content Management Systems** (WCMS) and play an essential role to manage content on Web sites. Many different open source CMS are available for this purpose and some of them also offer solutions to repurpose existing broadcast content in professional broadcast formats for the Web, e.g. Magnolia On Air\(^10\) for Magnolia, an open source java-based CMS, or Razuna\(^11\), an open source digital asset management system. Others worth mentioning are the Typo3 extension DAM (Digital Asset Management\(^12\)) and Media for Drupal\(^13\). However, it is not known that these CMS are used at broadcasters. Typo3 has been evaluated by the German PSBs but was not found to be suitable for the broadcaster’s needs.

2.4.2. **Interfacing Legacy CMS with NoTube**

Interfacing CMS is always an issue when access to the content in a CMS is needed and cooperation with the system manufacturers is often indispensable. The essence is typically stored in common formats and even if not, conversions of essence formats are usually not too much of a problem. The metadata, in contrast, is more complex and lots of work is needed to translate the metadata from the source format to the target format. Besides, the metadata model used is typically not publicly available.

In NoTube, this issue has been addressed by using the **Broadcast Metadata exchange Format** (BMF\(^14\)). BMF is specifically designed to allow a unified exchange of broadcast metadata in the TV production life-cycle from planning to play-out. WP2 is developing a set

\(^{14}\) http://www.irt.de/bmf
of Web Services that will provide an interface to the NoTube network which allows the contribution of content with associated metadata in BMF. The WP2 services will then convert this metadata to TV-Anytime (TVA), which is to be used as internal metadata format in NoTube. For instance, the legacy CMS from RAI, named ANTS, used in NoTube’s Use Case 7a provides metadata in the PrestoSpace format which will be converted to TVA via BMF.

2.5. User Profile Management

The User Profile Management, in NoTube, is performed in the scope of WP3 through the development of the module internally named “Beancounter”, supported by WP6 for what concerns the enforcement of security and privacy preserving policies.

This component interfaces with:
- NoTube applications – Via the back-end exposed through RESTful services
- Social Networks – Through ad-hoc software connectors, working as plugins

Internally it embeds the logic to:
- Manage the user identity, also supporting the OpenID credential standard.
- Manage user’s social networks credentials, through the OAUTH security standard.
- Collect and aggregate user activities from the various social networks, storing them locally using triples in order to facilitate further semantic processing – A SPARQL endpoint is available to the NoTube applications, as part of the abovementioned back-end, enabling complex queries on the aggregated data
- Infer user interests from the observation of the collected personal activities.

The integration of this component is the basis for every NoTube-powered application, including the three envisaged use case scenarios since it provides a single entry point for the authentication of the NoTube users, managing the profile and credentials and, moreover, it gives the possibility to leverage on the collected activities through any sort of SPARQL-based query.

With regards to the three NoTube application scenarios, the Beancounter has been integrated, at the time of reporting, with different specific goals sharing a common requirement which is the possibility to obtain content recommendations in light of the user’s personal profile. Due to the nature of the Beancounter itself, the user profile is dynamic one thanks to the continuous monitoring of the user’s activities.

Below is a high-level diagram showing the typical integration scenario of the User Profile Management – Authentication:

![User Management Integration - Authentication](image)

Below is a high-level diagram showing the typical integration scenario of the User Profile Management showing the internal activities of the Beancounter with respect to Social
Networks as well as the integration of a potential NoTube-powered application that requests user's details.

![Diagram of User Management Integration - Application Logic, Social Networks](image)

**Figure 4 – User Management Integration – Application Logic, Social Networks**

### 2.6. Recommendation and Enrichment

Recommendation and Enrichment functionalities are provided by WP3 and WP4 respectively.

Their integration is based on a close connection with the user profile component, as the main input on one side, and the metadata repositories used to perform the expected operation, namely the recommendation or the enrichment, on the other.

The typical scenario foresees that the NoTube user is authenticated (log in) then, within the specific workflow of the considered application, the business logic invokes the enricher or the recommender (or both) directly or through the broker, setting a specific goal.

As an example, let's take the 7.a scenario. The user sits in front of the television, within the Home Ambient, the set-top box is already switched on and logged with the user’s account. The user chooses to playback the daily news. The 7.a Home Ambient back-end provides the front-end with the personalised newscast, prepared in advance, on top of the ingested news provided and enriched by the Service Provider and then filtered and sorted starting from the output of the news recommendation service that, in turn, crosses the current user profile (interests) with the news’ metadata.

Below is a high-level diagram showing the typical integration scenario of the Recommendation services, through direct invocation.
Here the idea is that the application logic directly invokes the required services by interfacing the recommendation back-end that, in turns, triggers a set of actions including the exchanges with the user profile module, returning in the end the result to the application layer.

Please note that the Recommendation Engine has been running prior the application invocation (“run” arc on the Recommendation Engine timeline) since recommendation algorithms typically cannot run in real time but instead caching the obtained results at specific intervals, ready to provide recommendation request when queried.

Below is a high-level diagram showing the typical integration scenario of the Enrichment services, through the Semantic Broker.
In the previous diagram the idea is that the Semantic Broker, on the basis of the enrichment-goal, orchestrates the direct invocation of potentially different enrichment services and, more in general, any NoTube service needed (e.g. for metadata conversion), returning the result to the application logic.

### 2.7. Data protection

The scope of this section is to give an overview of the motivation behind the business requirements of broadcasters related to data privacy, to be considered when integrating the NoTube platform with such kind of stakeholders.

The process of program production for television has transformed from traditional analogue production to digitised formats. Due to this transformation, adaptations in the production workflow were necessary. The possibility of handling content in file form which allows the use of standard IT networks and components requires significant efforts in information management (content and asset management systems) which relies on the creation and use of metadata. In this context the term “asset” denotes the digitised content together with the rights-of-use information, where “content” refers to the essence plus related metadata (Figure 2). Different requirements related to privacy apply to the different types of data which are explained in the following.

- **Essence.** Since the start of migration to IT-based production environments in the 1990s, different audio and video file formats have been introduced for both standard and high definition applications using different coding technologies to reduce the amount of data. Depending on the requirements regarding the quality of image and sound, different formats are specifically designed for the production or distribution of content. While the high-quality production formats are used for handling and archiving the essence on the content owner's side, only the distribution formats which typically have a lower quality are used for providing the audio and video essence to the consumer. The distribution can be either free-to-air or cryptographically secured in pay-TV scenarios depending on the content provider’s business model. However, also for the distribution of free-to-air programmes DRM mechanisms need to be integrated to guarantee trouble-free access to the essence on DRM systems.
• **Metadata** covers all data that is required for the efficient and largely automated processing of useful data in broadcast environments (video, audio and additional data), for its administration and storage, and in particular for its exchange between the processing processes involved and the users. As a result of the rapid rise of computer technology in all areas of programme production, metadata must be compiled in IT formats, distributed in digital networks, automatically verified and saved. In an IT-supported system, direct access to the content is no longer possible. Access can only be achieved via metadata that describe the content. Privacy requirements apply especially to descriptive and administrative metadata, itself, but metadata is also used for access control and DRM. Descriptive metadata is manually created in a labour-intensive process to provide detailed information about the content during the production process and for the retrieval of archived material. This data also serves as basis for high-level descriptions of the final programme which is intended for the announcement and distribution of the content. In this context, privacy requirements apply since the programme descriptions are only used within the content provider’s network or given to contracting parties. Administrative metadata contains information related to business processes. This data is sensitive and shall not be distributed to the end consumer.

• **Rights information** is also metadata which holds information about usage rights that have been negotiated with the rights holders involved in the production. This data contains all restrictions regarding the use of content and is of importance for the production and the distribution of content. This information can be stored as metadata, which is again subject to privacy.

Concluding, it can be said that privacy constraints apply to production-related metadata which is only for internal use and distribution-related metadata which is restricted to the content provider’s network and contracting parties.
3. Use-Cases Integrated Prototypes Overview

This chapter aims at providing an overview of the 2nd use cases prototypes from the perspective of the integration with NoTube services. It is not intended to provide a detailed description of each use case component nor details of the individual demonstrators, which are part of WP7 deliverables. Please note that the 2nd integrated prototypes are scheduled to be finalised for March 2011, thus what is reported here reflects the status at the time of this deliverable due date (December 2010). The main goal then is to present the rationale behind the current status of achievements in light of the integration activities.

3.1. Personalized Semantic News

3.1.1. Overview

In the second prototype the Service Provider component now addresses issues related to the integration with legacy systems, both in the content ingestion area and in the persistence area.

The Personalized Semantic News use case deals with different kinds of contents, coming from several sources and stored in various sort of legacy CMS systems. In particular, in addition to the broadcaster’s ANTS system (RAI), already considered in the first prototype as the main internal metadata source for News, in 2nd prototype also live ingestion module has been adapted and integrated in order to produce and manage multimedia contents (Newscasts) independently.

In particular the live ingestion module is useful to acquire the Main Stream Quality Content (MSQC) in two different formats suitable for subsequent video processing (mainly cutting and cropping operations provided by WP4). It's worth noticing that, similarly to the Home Ambient component, the live ingestion feature can be controlled through scheduling info and input rules.

In the 2nd Personalised Semantic News prototype all ingested and derived contents are stored inside a local legacy CMS. At the moment, to support agile development and prototyping, this is a simple ad-hoc CMS running on top of the local file system. The adopted strategy of interfacing the NoTube platform with this legacy persistence layer led to make the platform’s internal references to the contents independent from the real physical location of the contents themselves. This has been achieved thanks to the introduction of the Content Reference ID (CRID) aimed at associating to simple ID a set of contents physical location.

A CRID resolution service is then needed to handle it by retrieving the multimedia file location.

In the second prototype this module has been developed as a local service in the scope of WP7a SP-HA integration, considering that in future developments it will be exposed as a public end-point enabling Service Broker content access for orchestration.

All the created News Item Containers (NIC), embedding metadata about individual News, are now made persistent as XML files in TV-Anytime format (NoTube persistence official format, as described in WP2) and they are saved inside a legacy repository which exposes only CRUD (save/extract/delete to be more specific) functionalities.

To improve the performance of NICs retrieval within such repository an indexing system running on a SQL-like database has been developed. The following diagram gives an idea of the internal current SP architecture focusing on the integration. For more detailed information please see D7a.3.
The same approach for NIC persistence and content resolution has been reflected in the Home Ambient, moreover a number of internal refinements have been implemented in order to reflect Service Providers’ improvements. The Home Ambient component is directly connected to the NoTube user profile manager developed in WP3 that exploits the users data to produce News recommendations, once again integrated in the HA back-end. The front-end has been redesigned, developed and integrated, enabling the end-user to perform standard tasks such as logging on and modifying his/her profile and interests as well as consuming the personalised Newscasts surrounded by enriched metadata coming from the Service Provider. The following picture represents the Home Ambient Integration Diagram.
Below are a couple of screenshots taken from the Web-based GUI in a typical usage scenario in which the user is enabled, once logged in, to playback the personalised Newscast.

Figure 9 – WP7.a : Personalised Newscast Playback

The playlist of individual News items is shown in top right corner. The upper/lower bars enable the user to access other functionalities including controlling the playback and logging out.
While the Newscast is played, external links to related contents are shown in the bottom-right corner and the following screenshot presents the user choosing one of those, within the just described context.

Figure 10 – WP7.a : Dynamic Links to External News
More details can be found in D7a.3.

### 3.1.2. Integration

With reference to the integration diagrams presented above, the Personalized Semantic News scenario integrates a subset of the NoTube services from the described categories, in particular:

- **Metadata-oriented services for creation and access to News Items Containers (NIC) using the NoTube persistence format (Tv-Anytime).** On one hand the focus has been to leverage NoTube services in order to create a new NICs using the NoTube persistence format, starting from the legacy output (ANTS) in Prestospace-derived format. On the other hand the focus is to exploit metadata CRUD services to transparently access and manage information which is saved inside the NIC. Such services, since not ready in WP2, have been mocked-up internally while the whole integration infrastructure has been developed and it's now ready. The CRUD services will be accessed through the Broker (WP5) in order to enable potentially more complex workflows (i.e.: involving further conversions/metadata extraction).

- **Metadata-oriented services for News Item enrichment.** In 2\(^{nd}\) prototype the *Lupedia* service provided by WP4 has been integrated by means of a the exposed Semantic Broker goal.

- **Audiovisual content services for News Item enrichment.** In 2\(^{nd}\) prototype we've designed the introduction of WP4 smart cropping service. The integration is on-going at the time of reporting.

- **EPG retrieval services.** In 2\(^{nd}\) prototype specific services about the retrieval of Italian EPG have been integrated via the Semantic Broker that improved the first year achievements by means of new goals.

- **Metadata CRUD services (HA side) in order to handle News Items Containers**

- **User profile management services (HA side) in order to handle user’s details, activities and session.**

- **News Recommendation Services (HA side) in order to create the personalised News playlists**

Moreover the integration between SP and HA components has been empowered through the improvement of the live ingestion module, within the home ambient, now capable of supporting industry standards for audio and video raw data (MPEG2-TS). In the same way the first step of the integration between the Home Ambient back-end and the newly developed front-end has been performed providing connectivity for user authentication and personalised News playlist retrieval, including the live streaming of A/V contents.

### 3.2. Personalized TV Guide with Adaptive Advertising

#### 3.2.1. Overview

During the first year the Use Case 7.b partners designed three sub-scenarios built around the concept of the Personalized EPG (PEPG):

- Internet PEPG (*iFanzy*) Ads
- Using *iFanzy* in social context
- Adaptive Ads in video

The three sub-scenarios leverage on a proprietary component named *iFanzy* television guide. This component is separated in two individual parts, the *iFanzy* client (Home Ambient), which is custom-built for the various platforms which interacts with the second part, the *iFanzy* Server (Service Provider). This legacy component basically provides services for A/V assets (TV programmes, movies, Ads) management with focus on advanced filtering and detailed metadata.
Below is a quick summary of the main planned features of the three abovementioned sub-scenarios.

**Internet PEPG (iFanzy) Ads**
- Set recordings via internet-based client
- Availability of IMDB rating in programmes details
- Serving pre-roll advertisement before video-preview
- Upload video file (and meta-tags) as additional information to a programmes page

**Using iFanzy in Social Context**
- Receive reminders on a mobile client
- Overview reminders on a STB client
- Multi-user login on STB client
- English STB client interface and ads
- Manage favourites on mobile client with multi-modal interface
- Set automatic recording through the mobile client with multi-modal interface
- “Send to friend” function on the mobile client with multi-modal interface
- Recommendation of programmes similar to the one consumed by the user

**Adaptive Ads in Video**
- Multiple account support
- VOD portal with personalised ad placement and multi-language support
- Possibility to watch videos with or without Ads

During the second year a more focused approach has been applied in order to create a NoTube-compliant prototype that would take care of the major planned features presented above. A storyboard for the prototype has been derived and adopted for the software implementation of WP7.b, as described below.

![Figure 11 – WP7.b : Storyboard 1/3](image)

Sena, who lives in Holland, wants to know if there’s something interesting on TV this week. She visits [iFanzy.nl](http://ifanzy.nl) to have an overview of what is broadcasted on TV. Sena logs in and sees what TV programmes are recommended to her. She browses the EPG, sees one of her favourite programmes and decides to rate it and add it to her personal planner. Furthermore she sees a movies and watches the related trailer. Besides that, she also want to check whether this movie has a high IMDB rating or not. After browsing the [iFanzy](http://ifanzy.nl)
Website, reading some information about TV programmes, rating some programmes, pur some programmes in her planner and so on, Sena forget time and suddenly has to hurry to catch her flight to join her boyfriend Tarik in Turkey.

![Storyboard Image]

Figure 12 – WP7.b : Storyboard 2/3

Luckily Sena did catch her flight and landed in Istanbul safely. While waiting for her luggage at the airport, she is bored. She gets her iPhone and starts up the iFanzy iPhone application. She browses what’s on this evening and she sets a reminder for movie recommended to her by the system. There is her luggage already.

![Storyboard Image]

Figure 13 – WP7.b : Storyboard 3/3

Later that day, Sena and Tarik have a nice dinner together. After dinner, while they have a drink, Sena gets a notification that the movie she set a reminder before starts in about 15
minutes. Sena asks Tarik if he would like to watch this movie with her. Tarik is not really interested in this movie. However he likes the idea of watching a movie together that evening. He grabs his laptop, logs in on *iFanzy.tv* and searches for a movie they both like. They watch a movie together happily ever after.

Of course the one presented above is just a plot. In order to take a closer look of the prototype current implementation, please refer to WP7.b M23 deliverables and software.

### 3.2.2. Integration

The integration approach for the WP7.b use case is centred on the role of the Semantic Broker as the unique entry point for accessing NoTube services. The following picture provides the integration approach adopted by WP7.b developers:

![Figure 14 – WP7.b Integration Diagram](image)

The integration process for the considered use case focused on the implementation of the communication infrastructure with the Semantic Broker which can search the different Engin EPG feeds for programme information that falls within a given time period. The *iFanzy* client is then enabled to leverage transparently on this functionality.

### 3.2.3. Retrieving enriched EPG data

For the second WP7.b prototype, the focus was on integrating services that retrieved electronic programme guide (EPG) data. At its most basic this involved exposing, via the Broker, a higher-level service which allows searching the different available EPG feeds for programme information that (a) falls within a given time period, (b) matches the user language and (c) optionally also matches particular user search requests (e.g., keywords). Furthermore, the raw EPG data was enriched with *DBPedia, IMDB*, and SKOS entities that
provided additional information about each programme in the EPG feed. To offer this functionality, the Broker orchestrates a number of services which, for instance, harvest, filter and enrich EPG data from different sources. The *iFanzy* client was able to invoke this functionality via the Broker (specifically a goal called *GET-EPG-BY-KEYWORD-PERIOD-AND-LANGUAGE-GOAL*\(^{15}\)) and display it in the interface.

### 3.2.4. Profile-based TV programme recommendation

One of the main aims of the *iFanzy* client is to display TV program recommendations based on user’s activities and personal interests as stored in a user profile. Thus the core services that underpin this scenario are user profiling and recommendation services provided by WP3. The current services provided by WP3 have been semantically described in the Broker and exposed via goals to external applications. For example, two key functionalities from the Beancounter (WP3) that are exposed via the Broker offer functionality to retrieve a user profile and a list of user activities. With respect to the first of these functionalities, WP5 has provided a goal called *GET-BEANCOUNTER-USER-PROFILE-GOAL*\(^{16}\).

The service exposed via the Broker takes a known user and returns the profile for that user. So for example, there is a user in the Beancounter called *michele_minno*, thus to retrieve this user's profile, the following URL is invoked:


With respect to the second of these Beancounter functionalities, WP5 provided a goal called *GET-BEANCOUNTER-USER-ACTIVITIES-GOAL*\(^{17}\). The service exposed via the Broker takes a known user and returns the list of activities for that user. So for example, using the same *michele_minno* user as above, to retrieve this user's list of activities the following URL is invoked:


These goals provide an initial basis for integrating user profile-related functionalities into the WP7.b prototype and are intended to be incorporated into higher-level orchestrations based on particular requirements of the 7b scenario.

### 3.3. Internet TV in the Social Web

#### 3.3.1. Overview

The objective of work package 7.c is to exploit the state of the art technologies in providing Internet-based social features for television. In the first year WP7.c partners developed a single integrated prototype showing:

- Control of broadcast TV using the 'Buttons' API
- Reuse of a user's existing activity data from social media sites to generate personal programme recommendations

---

\(^{15}\) Goal documented here: http://www.notube.tv/wiki/index.php/Wp5-goals-documentation#GET-EPG-BY-KEYWORD-PERIOD-AND-LANGUAGE

\(^{16}\) Goal documented here: http://www.notube.tv/wiki/index.php/Wp5-goals-documentation#GET-BEANCOUNTER-USER-PROFILE-GOAL

\(^{17}\) Goal documented here: http://www.notube.tv/wiki/index.php/Wp5-goals-documentation#GET-BEANCOUNTER-USER-ACTIVITIES-GOAL
In year 2 they have taken a slightly different approach, using a rapid prototyping second screen environment (XMPP over HTTP) to create several sub-prototypes that show various aspects of the rapidly changing world of social media and TV.

Three sub-prototypes have been developed so far during the second year of the project, as described below.

### 3.3.1.1. Sub-Prototype 1

**On-demand Video Browser Showing Programme-To-Programme Recommendations**

This internal prototype uses BBC archives metadata, and in particular the Lonclass UDC-based classification system, together with the CRID resolver (see below) to generate 'best suggestions' based on number and diversity of links between two programmes. The user can browse through them and the play on a near or far screen. WP7.c partners have developed a Korean as well as an English version.

![Figure 15](image-url)

**On-demand Video Browser Showing Programme-To-Programme Recommendations**

### 3.3.1.2. Sub-Prototype 2

**Real Time Non-Text Interactive Voting Application**

This second sub-prototype uses the anonymous XMPP and the Twitter @Anywhere API to allow users to view relevant social network messages while voting for various options and seeing the results in real time.
3.3.1.3. Sub-Prototype 3

Extension Of The Year 1 MythTV Prototype With Social Network Services

This is a work in progress that allows the user to see EPG ‘hotspots’ based on local Twitter activity, plus messages and recommendations from friends if logged in, as well as the existing control of MythTV and feedback about what’s currently playing.

Two further pieces of infrastructural work have been developed to support this work:

- Development of the Buttons API - The Buttons API has been developed to include some on-demand commands
- CRID resolver - This takes pieces of metadata sent along with the broadcast and links them to exiting BBC http-resolvable URLs

3.3.2. Integration

The basic idea is to improve and develop the architecture designed during the first year. The XMPP protocol provides a bridge between two paired applications, and the Buttons API describes the messages sent (e.g. 'PLAY', 'PAUSE') and received. In year 2 WP7.c partners have been using XMPP over HTTP to speed up application development and integration with the Twitter API. As before, data services are provided over HTTP for example enhancement, programmes metadata, recommendations services, social networking data and profile data.
Figure 17 – WP 7.c Integration Diagram

Please refer to section 4.1 for a better understanding of the coloured boxes.

The use case 7.c is in a significantly different position to the other use case partners for two reasons:
- All BBC programmes metadata is already publically available in various machine-readable formats
- The world of social networking is extremely fast-moving

The first of these enables WP7.c partners and the latter compels WP7.c partners to prioritise speed over service reuse. The CRID-resolver can be rolled into the Semantic Broker when stable; and work has begun on integrating XMPP into the Broker. Feedback has also been provided about the current services based on the prototyping experiences.

Concerning the low-level integration with the NoTube platform, the third sub-prototype uses services from WP4/1 directly for enrichment.
For the Korean version of sub-prototype 1 various conversion services have been used, to go from Lonclass to Korean Wordnet via English Wordnet.
It is planned that sub-prototype 3 will also use a new Lonclass-based service from WP4. Sub-prototype 1 could also be potentially improved with profile services from WP3.
4. NoTube Platform Services

4.1. Overview

The rationale behind this chapter is to introduce the services of the NoTube platform as provided by the technological R&D work packages (WP1, 2, 3, 4, 5) and as depicted in D6.1. Such services have been grouped in categories reflecting the general architecture design as well as the integration diagram presented in Chapter 2.

In summary:

- Identity Management
- Authentication with Privacy Preservation
- User Interests Management
- Activity Logging and Retrieval

- Format Conversion
- Recommendation
- Enrichment

- Ingestion
- Streaming
- Conversion
- Audio and Video Processing

The technical details of the various services have been collected initially in the scope of WP6 (more details in D6.2 Annex) and then, due to the need of obtaining a formal annotation in order to enable the Semantic Broker (WP5) to properly operate, demanded and supported by a Web Tool called Smart Link Service. The next section provides an overview of it, more details could be found in WP5 of course, including services annotation for semantic purposes.

4.2. Services Description and Annotation

This section describes the work done in supporting the task of semantically describing and annotating the services in the NoTube platform. One of the requirements for the services work in the project (done under both WP5 and WP6) is the semantic annotation of TV data and service sources (see in particular Task 5.2 as set out in the Description of Work and as anticipated in D5.1a/b). Thus, the work here is motivated by a need to document, in both human-readable and machine-understandable form, the services throughout NoTube, and to provide a basis for formal service descriptions that automate the discovery of services.
4.2.1. Lightweight Services Annotation: the Linked Services approach

In order to support annotation of a variety of services, the EC-funded project SOA4ALL, has developed *iServe*, a novel and open platform for publishing semantic annotations of services based on a direct application of linked data principles\(^{18}\).

*iServe* supports publishing service annotations as linked data - Linked Services - expressed in terms of a simple conceptual model that is suitable for both human and machine consumption and abstracts from existing heterogeneity around service kinds and annotation formalisms.

In particular *iServe* provides:
- Import of service annotations in a range of formalisms (e.g., SAWSDL, WSMO-Lite, MicroWSMO, OWL-S) covering both WSDL services and Web APIs;
- Means for publishing semantic annotations of services which are automatically assigned a resolvable HTTP URI;
- Support for content negotiation so that service annotations can be returned in plain HTML or in RDF for direct machine consumption;
- SPARQL endpoint allowing querying over the services annotations;
- REST API to allow remote applications to consume and provide annotations.
- Support for linking service annotations to existing vocabularies on the Web.

In order to cater for interoperability, *iServe* uses what can be considered the maximum common denominator between existing SWS formalisms which we refer to as the *Minimal Service Model* (MSM). The MSM is a simple RDF(S) ontology able to capture (part of) the semantics of both Web services and Web APIs in a common model. MSM is extensible to benefit from the added expressivity of other formalisms.

4.2.2. Service Annotation and Integration via SmartLink

With the conceptual foundation of the MSM in place as the basis of the Linked Services approach, the next step was to build a tool that would allow MSM service annotations to be generated from scratch. *SmartLink* was built for such a purpose. *SmartLink* is short for "SeMantic Annotation EnvironmenT for Linked Services". Simply put, it is an easy-to-use Web application aiding users in the creation of Linked Services (which, as stated above, are semantic service annotations following Linked Data principles). Amongst other things, it provides an interface to populate and query the Linked Services repository *iServe*.

*SmartLink* builds on existing technologies and standards to enable wide reach of its annotations. Users can annotate arbitrary services - whether REST-ful or WSDL/SOAP-based - via a simple Web form. Annotations are stored in RDF following established service schemas, namely the Minimal Service Model\(^{19}\) which follow a light-weight approach to Semantic Web Services. Storage of annotations is spread across two public RDF-stores: *iServe*\(^ {20}\) handles all functional properties defined in the MSM schema while an additional and *SmartLink*-specific SESAME repository hosts further non-functional service properties. These non-functional properties are, for instance, contact person, developer name, Quality of Service (QoS), development status, service license, and WSMO goal reference. The latter property directly contributes to facilitate our vision of allowing MSM models to refer to existing WSMO goals which utilise the same service entity; that is, it facilitates our model referencing vision between MSM and WSMO models. In addition, by allowing developers to directly annotate existing REST-ful services and APIs, *SmartLink* directly provides another

\(^{18}\) [http://linkeddata.org/]
\(^{20}\) [http://iserve.kmi.open.ac.uk](http://iserve.kmi.open.ac.uk)
contribution to enable our service model integration vision based on allowing the annotation of WSMO goal requests – which in fact are REST-ful services themselves – as MSM service instances.

*SmartLink* currently provides mechanisms that enable the export of particular (MSM) service instances as RDF or human-readable HTML. In order to facilitate service model transformation and augmentation between MSM and WSMO, current research deals with the establishment of an export mechanism of MSM service models as WSMO instances. While current implementation work is concerned with adding corresponding export facilities to *SmartLink*, model transformation is just enabled on a manual basis at the moment.

In summary, the main *SmartLink* features include:

- A simple Web annotation form
- An RDF schema defined by OU and VU (WP1) based on existing service model standards
- The ability to define references from a service annotation to any existing Linked Data vocabulary (either NoTube or external vocabularies)
- Storage in *iServe* and an additional OWLIM repository
- Export of individual service descriptions into RDF and human-readable HTML
- Service browsing and search facilities
- Classification of services based on the NoTube services taxonomy (ongoing work with WP6) and other established, general-purpose service classification schemes
- A SPARQL endpoint
- *OpenID* authentication

The aim of *SmartLink* is to be the main environment to document, annotate and search for services within NoTube. As such, it replaces the existing service documentation efforts (i.e., the services catalogue and the internal documents related to it) with a single tool that allows not only documentation for humans (i.e., developers) but also structured, machine-understandable annotations that are accessible for the Broker or third-party applications.

Furthermore, *SmartLink* provides a SPARQL endpoint, allowing us to share data about all the NoTube services publicly in a structured way. While *SmartLink* currently mainly aims at providing structured, machine-processable annotations and query facilities – without supporting service execution - current work aims at an API layer which allows to discover and execute in a more automated manner. Finally, the long-term plan is that the *SmartLink/iServe* environment will provide capabilities to fully replace the current brokering environment that is based on IRS-III. The new tools will allow the OU to provide its services via a single coherent environment based on the emerging Linked Services approach.

4.3. **User Profile Services**

4.3.1. **Overview**

Briefly summarising WP3 and WP1 activities with respect to the definition of the NoTube User Profile, we consider the following attributes as part of it:

- Static data (i.e.: name, surname, age, sex, etc.)
- Application specific data (i.e.: chosen TV channels, News interests, etc.)
- Activities

While the first two bullets are trivial to be explained, more interest has been put into user activities. Within the scope of the component develop by WP3 named Beancounter, we
consider them as a list of couples (interest, weight). This represents a simplified version of the adopted *Weighted Interests Ontology*\(^{21}\).

An interest can be everything that can represent a relevant fixed point in the endless stream of user Web activities. It can range from an actual object of a user activity (i.e.: a movie, a song, etc.) to a very generic topic or concept (i.e.: computer science, horror genre, etc.).

Any computed interest comes with a weight telling the strength of that interest in the user recent behaviour. The weight is expressed as a rational number between 0 and 1. The user profile is computed by the Beancounter starting from the activity stream of the user, gathered among the various social networks the user hangs out at. Then the different activities are aggregated and semantically lifted according to the *Atom Activity Stream Ontology*\(^{22}\) we devised.

In order to keep track of the user activities which gave rise to a certain interest, each computed couple (interest, weight) comes with a list of reasons. Each reason is a textual description of an activity the user performed and that has brought to the computation of the interest itself.

One of the crucial point of the user profile generation has to do with the inferring of the very general categories of user activity objects, when it is possible. So a big challenge during the development of the user profile services was the definition of another service not directly called by other NoTube components, but internally responsible for the computation of the most relevant categories for each object. A good set of interests is strongly dependant on a good categorization of user activity objects.

**4.3.2. Use Cases Connection**

The requirements that led to the past, current and future WP3 services development have been collected from the very beginning of the project, focusing on the three envisaged scenario as the most realistic starting point. Gradually following the application scenario additional details that has emerged during the months, the WP3 internal work has been adapted and has evolved in the second year of the project trying to accommodate any deviations in the application scenarios as well as following internal R&D goals.

More details are provided in WP3 deliverables, however below it’s an overview of the different aspects that have been considered until the release of the second integrated prototype.

- The WP7\textsuperscript{a} requirements are about user authentication, providing user interests as categories and the current device as user profiling context.
- WP7\textsuperscript{b} added more requirements about user personal information which, if available must be present, in the profile, such as: age, gender, language, favourite programme list and interest list. And more requirements about user profiling context:
  - Time of day (dinnertime, evening)
  - Day of week (Saturday, Monday)
  - Time of year (summer, winter)
  - Device, multimodal capabilities
  - Mood, feelings
- The WP7\textsuperscript{c} emphasized the following requirements:
  - Allow users to add and remove social data accounts to their respective Beancounter account
  - Provide an RDF-based output of the profile

4.3.3. Data Sets Connection

As the main input to the user profiler, the Beancounter module uses the different web REST services made available by the various social networks servers. These servers contain all the information about user activities and return responses in different proprietary formats. The Beancounter is capable of aggregating all these responses, lifting them to a semantic layer by using the Activity Ontology devised within the NoTube project and mentioned before.

With regards to the categorisation of the activities objects, DBpedia\(^{23}\) has been used. It is the main all-embracing ontology, with data coming from the more human-readable Wikipedia\(^{24}\).

4.3.4. Integration

Each use case is targeting end-users, then it becomes obvious that user profile services are adopted by all the application scenarios, with different degrees of integration depending on the specific requirements and the current status of the prototypes. Please see use cases deliverables for more details about that.

In addition to the abovementioned features, the Beancounter provides also the possibility to directly ingest user activities. These activities are pushed into the triple storage using a format common to the three use cases. This format includes the following elements: verb, object, time and source of the activity. A REST service allows to push a user activity according to this format. The main purpose of this feature (apart from easily creating some test data about user activities) is to have feedbacks from the user. Specifically the third use case (WP7.c) requires the gathering of user activities on the TV channels involved in the recommendations, after the recommendations having been sent, in order to be able to assess the satisfaction index of the user about recommended media items. Also, to refine user profile in view of forthcoming better recommendations. The same kind of behaviour is required by the first use case (WP7.a), with the media domain narrowed to the News.

4.4. Metadata Services

4.4.1. Metadata Conversion Services

4.4.1.1. Overview

WP2 provides services for the conversion of input metadata to the internal metadata format used in NoTube. In particular, WP2 services allow the conversion from metadata in PrestoSpace format which is provided by use case 7.a and egtaMETA which is provided by 7.b to TV-Anytime (TVA).

TVA is the metadata format internally used in the NoTube platform. Furthermore WP2 provides use case 7.a specific metadata services for create, replace, update and delete operations (CRUD) on a News Item Container (NIC) in TVA respectively RAI-NIC format.

The main challenge was to map all the input metadata elements to appropriate metadata elements in TVA and to provide the means for CRUD operations on the TVA metadata considering the target metadata formats used in the use cases.

\(^{23}\) http://dbpedia.org
\(^{24}\) www.wikipedia.org/
4.4.1.2. Use Cases Connection

Use case 7.a

The metadata conversion services have been developed closely with the use cases. Use case 7.a provided very detailed requirements concerning the conversion of metadata and CRUD operations which have been reflected in the design and the implementation of the services. The requirements regarding the metadata conversion comprised a detailed description of all the metadata elements which need to be provided after the conversion process. Furthermore, a list of the CRUD operations that need to be provided through the WP2 services has been provided.

For use case 7.a (Personalised Semantic News) the essential role for the metadata exchange has the News Item Container (NIC). The characteristics and the related requirements of the NIC have been analysed and noticed during the implementation of the transformation services. This work has been done in close collaboration with the work package 7.a members.

The TV-Anytime (TV-A) format has been selected as internal NoTube format for the provider side as well for the Home Ambient. Therefore the NIC concept was mapped into TV-A with respect to the required information. The alignment of the required data to describe a NIC and the TV-A transformation implementation has been done during the technical evaluation phase in WP2. During this work, examples of TV-A instances have been developed and discussed with the members of use case 7.a. The results were reflected to the adjustment of the implementation of the transformation services.

The detailed description of the requirements for the CRUD operations, provided by use case 7.a members, has been used to design and to implement the technical interfaces. These interfaces will be accessible via the CRUD service, implemented by WP2. This service allows applications within the NoTube platform to receive and to manipulate NIC objects. This is essential to support the specified features for the personalised semantic news use case. Furthermore, the service can be used within other use cases to have access to News related information, if needed.

Use case 7.b

For use case 7.b, the requirements were less complex, even though, concerning the metadata conversion, have been similarly reflected by the design and implementation.

For use case 7.b (Personalised TV guide with Adaptive Advertising), the work has been realized on the EBU/EGTA scheme (for the description of the advertisements). The characteristics and the related requirements of the EBU/EGTA have been analysed and noticed during the implementation of the transformation services. This work has been done in close collaboration with the work package 7.b members.

The TV-Anytime (TV-A) format has been selected as the internal NoTube format for the Provider Side as well for the Home Ambient. Therefore the EBU/EGTA format was mapped into TV-Anytime with respect to the data required by use case 7.b. The TV-A transformation implementation has been done during the preliminary technical evaluation phase in WP2. During this work, examples of TV-A instances have been developed and discussed with the WP 7.b leader. The results were reflected to the adjustment of the implementation of the transformation services.

Use case 7.c

Currently it is not foreseen in use case 7c to use any of the metadata conversion services. More details will be part of D7c.2.
4.4.1.3. Data Sets Connection

The WP2 metadata conversion services need to be connected to both internal and external datasets.

Use case 7.a

For use case 7.a, the metadata conversion service needs to access the PrestoSpace metadata coming from RAI’s ANTS System which is provided on an FTP server. The converted metadata will be provided to the NIC repository which is under RAI’s responsibility.

As internal metadata format the TV-Anytime format has been selected. Therefore the NIC objects are stored as TV-A objects inside the NIC repository.

But in order to be future proof, a more generic approach has been implemented instead of a simple one-to-one mapping from PrestoSpace to TV-A. To enable other broadcasters to connect to the NoTube platform, a uniform interface is needed. Therefore, the Broadcast Metadata Exchange Format (BMF) was chosen by WP2 as the interface between the NoTube platform and external broadcast sources. The transformation process maps first the PrestoSpace data into BMF and then the transformation BMF-to-TV-A is done. This approach enables easily to add other metadata formats (e.g. FESAD, a widely used format by the German Public Service Broadcasters) at a later time. These different transformations are implemented as separate services and due this fact the modular approach is supported.

For the CRUD operations, the implemented service needs as input parameter TV-A datasets which are describing the NIC objects. Because of this parameter driven design of the CRUD service interfaces no direct access to the NIC repository is needed. The application which is using the CRUD service is responsible to have access to the NIC repository. The CRUD service is therefore independent of a specific repository technology. A detailed description of the CRUD service and the interfaces can be found in D2.3 of WP2.

Use case 7.b

For use case 7.b the metadata conversion service needs to access the EBU/EGTA metadata which is provided on an internal FTP Server.

As internal metadata format, the TV-Anytime format has been selected. The service which allows converting EBU/EGTA metadata in TV-Anytime format has been published and made available through a Web Service.

4.4.1.4. Integration

The services have been developed in close collaboration with WP5, WP6 and the use case partners. It was decided to integrate the services provided by WP2 via the Semantic Broker. This approach ensures re-usability between all the use cases and future developments. It also ensures the integration in the project-wide workflow and enables the Broker to perform the desired orchestration involving metadata conversion when needed.

Internally, WP2 services will use TV-Anytime as the common metadata format. All operations concerning the services can be called via Web Service operations. Input formats, provided via the Broker, will be converted into TV-Anytime (where necessary), processed and re-transformed into the requested output format (e.g. RAI-NIC format).

4.4.2. Recommendation Services

Recommendation services have been researched and developed in a joint effort between WP3 and WP1.
Since the application scenarios scope concerns News Items (for WP7.a), advertisements (for WP7.b) and media programmes (for WP7.c), the general idea is to link those items to the user profile interests by means of the interlinked semantic data spread on the Web. In this way we can leverage existing knowledge in order to find the best recommendations for each user.

4.4.2.1. Overview

The main challenge of any recommendation services in the Semantic Web context is to gather the proper additional information in order to perform the right recommendation to the specific user. Since the Linked Data Cloud\textsuperscript{25} owns a huge mass of structured data, divided in different areas and domains, any recommender service must choose to leverage some relatively little portions of the cloud. For the already implemented services, it has been chosen to exploit a very specific, but broadly used ontology: SKOS\textsuperscript{26} (Simple Knowledge Organization System). It is an ontology shaped to classify concepts and to set relations between resources (that is, instances of concepts) and concepts. It is used in DBpedia to describe resources. Thus the recommender services that leverage SKOS accept as input user profile and EPG lists which contain references to DBpedia resources, or directly to SKOS concepts. It looks for common SKOS concepts in both sides, and recommends EPG media items with a high number of matching SKOS concepts.

Another kind of recommender that also leverages SKOS is the item-to-item recommender. Instead of recommending user-specific media items, it is a generic similarity meter between two pieces of content. Thus it recommends a certain media item starting from another similar media item, without taking as input any user profiles.

A third kind of recommender that also doesn't need any user profile to work is the property-browser recommender. It is still in a very prototypical status, so it isn't yet included among the available recommenders. This recommender takes as input only the activity objects of the user, without any computed interest. Then, as usual, takes as input the EPG list with all the media items to be recommended. It tries to 'browse' all properties that relate activity objects and EPG items to other resources, looking for potential property paths between an activity object and an EPG item. An example of path could be:

- Activity: user x likes Pulp Fiction
- Activity object: Pulp Fiction
- EPG item: From dusk till dawn
- Path: Pulp Fiction $\rightarrow$ is directed by $\rightarrow$ Quentin Tarantino $\rightarrow$ acts in $\rightarrow$ From dusk till dawn

Filtering in a proper way these kind of useful paths, interesting EPG items also not directly related to the user activities can be discovered.

On top of that, a parallel work on recommenders has been performed by WP1 as part of internal R&D activities, trying to focus more on the application requirements of the NoTube scenarios. In particular they have defined two types of recommendation services:

- Social TV recommendation - based on genre of the users, its friends and demographics)
- Semantic TV recommendation - based on semantic enriched tv broadcast in relation to semantic enriched user profiles.

\textsuperscript{25} http://linkeddata.org/  
\textsuperscript{26} http://www.w3.org/2004/02/skos/
For Social TV recommendations, a number of services have been created in order to enable integration and prototyping, mainly in the use case 7.c, which are now embedded in an internal experimental environment named "iZapper":

- based on levels of interestedness
- based on IMDB score
- based on genre a user has watched before
- based on brand a user has watched before
- based on a bad (<6) IMDB score
- based on genre friends have watched before
- based on brand friends have watched before
- based on brand a user has never watched before
- based on genre a user has never watched before

According to the analysis depicted above, for semantic recommendations two strategies have been applied:

- Direct match: equality between user interest and enrichment of broadcast
- SKOS match: found path between user interest (SKOS concept) and enrichment of broadcast

### 4.4.2.2. Use Cases Connection

Each use case needs different kinds of recommendation strategies in order to work.

The WP7.a use case requires to build a list of relevant News Items, namely a personalised News cast. So the recommender service should be able to compute an user satisfaction value for each News Item that would be potentially added to the list, in the Home Ambient. This value would be then used for computing the final list for the specific user. The input News Item comes with a list of DBpedia resources automatically extracted from the News text. This initial enrichment made directly by the content owner (RAI) made possible the use of SKOS Ontology to find potential matching between user profile interests and News Items.

The second use case WP7.b aims at providing the user with personalized advertisings. No specific requirements has been provided yet to WP3 in order to build further services since the prototype is capable of leveraging on the legacy iFanzy server.

The third use case WP7.c is built around the idea of having a TV experience augmented with some typical social Web features, like collaboration, user generation and interest sharing.

---

33 [http://services.notube.tv/zapper/recommendation.php?username=Chris&criteria=brand&social=true](http://services.notube.tv/zapper/recommendation.php?username=Chris&criteria=brand&social=true)
4.4.2.3. **Data Sets Connection**

To test the News recommender (WP7.a) WP3 used some News Item Containers (NIC) provided by RAI as part of the evaluation data set. Since data represents the XML-based resource containing metadata relevant to a single piece of News content.

WP3 needed also some more generic EPG content provider by leveraging the RESTful Web Services made available by WP1 through their internal EPG Dataware house\(^{37}\), mentioned later on with respect to enrichment-related activities, providing EPG in a chosen time range by different channels, among which also BBC channels.

From user side, a set of dummy users has been created with related accounts on different social networks and activities about media items (Web pages, songs, etc.).

4.4.2.4. **Integration**

Similarly to user profiling, for recommendations there’s been the need to agree on a common format, the one that then would be used in the three use cases prototypes. Currently:

- The News recommendation service for WP7.a use case is made available via REST. It accepts as input parameters the user identity and the NIC (News Item Container) file describing an actual News item and returns a value between 0 and 1 representing the expected user satisfaction level with respect to the considered individual News.
- Recommendation services for WP7.c are also released as well as RESTful Services. They differ in terms of involved algorithms but they all share the same input parameters: the user identity, starting date and channel of the EPG the recommender has to leverage in order to find interesting media items to be proposed. The services return a list including the mentioned interesting media items together with the reasons about their choice.

4.4.3. **Enrichment Services**

4.4.3.1. **Overview**

This section provides a detailed description of the text enrichment services in relation to metadata acquisition and enrichment. The motivation and needs behind the services have been defined in deliverables D4.1 and D4.2.

*Lupedia* is a text enrichment service whose main function is recognising public names in texts, relying on dictionary data from an external knowledge base. The initial version of the service has been described in deliverable D4.4 with subsequent extensions in D4.5. Over the course of the project, the needs of the various work packages and partners have become clearer and as a result *Lupedia* has been continually extended.

The initial release of *Lupedia* was capable of finding *DBPedia* entities in English texts and providing the entities together with their most generic class and position in the input text. The classes were *DBPedia*'s base classes Work, Place, Organisation, Person and Event and they proved to be too generic for most purposes. For example, films, books and songs were all identified as [http://dbpedia.org/ontology/Work](http://dbpedia.org/ontology/Work) instead of the corresponding most specific types.

The second version has addressed this problem by providing all classes from most generic to most specific, with the option to leave only the most specific one. In addition, users can

---

\(^{37}\) [http://services.notube.tv/epg/datawarehouse.php](http://services.notube.tv/epg/datawarehouse.php)
choose the classes used for matching and thus get very domain-specific results (e.g. matching only films by specifying http://dbpedia.org/ontology/Film).

In the setting of multilingual Europe, one of the most important improvements was adding support for seven additional languages: Bulgarian, Dutch, French, German, Italian, Korean and Turkish, thereby covering the languages of all project partners. Searching can be performed in one or more language at the same time and all languages but Korean have stop word filters.

The initial set of basic filters and options controlling Lupedia’s operation was extended with various new filters and options to match the needs of individual partners. As the number of options grew, it became unclear what the best (for typical usage) settings were. Therefore, WP4 evaluated the effect of the various filters and options on typical textual programme descriptions and were able to determine the best default settings for the service: skip short matches, skip matches composed of stop words, keep only first and longest match, keep only matches with the highest weight, keep only the most specific class, case sensitive. Specific use cases and scenarios can still choose individual options.

Lupedia is capable of using heuristics when the input does not match a given dictionary entry completely. This enables us to find matches like Pulp Fiction (film), typically encountered as Pulp Fiction, and The Pink Panther (sometimes encountered as Pink Panther).

DBPedia provides a rich set of predicates (attributes) for each entity and some predicates are more salient than others, e.g. the proper name is preferred over a nickname. A good example would be The Matrix, which is the main name of the film http://dbpedia.org/page/The_Matrix and a nickname for the person http://dbpedia.org/page/Andre_Dirrell.

WP4 introduced predicate weights to help deciding over preferred predicates, whereby the preferred ones will have a higher weight. Heuristics and predicate weights are combined into a single resultant weight that can be used to filter out undesired matches.

In parallel to WP4 work focused on Lupedia, similarly to recommendation services WP1 carried out internal R&D activities about enrichment basing on the assumption that the multilingual aspect of Wikipedia can enhance semantic annotations: By generating a larger annotation set than from the English pages only, by giving more precise annotations than “disambiguation” URIs in some cases and by generating a network of relationships between the annotation URIs.

Wikipedia is a multilingual user-contributed resource, meant for human browsing. The pages in different languages are not translations of each other, but truly different pages on the same subject. Some overlap can be found, but different authors bring different light on the subject. RDF triples can be generated from these pages, providing machine-readable information. This information can be used for semantic search or recommender systems, to cite only a few applications. The DBpedia project generates RDF triples from the info boxes of Wikipedia pages in English, and adds some multilingual information: labels and comments in different languages if available. VUA claims that other sections of the unstructured plain text of Wikipedia pages can bring useful triples, particularly in terms of content description. The main added value of DBpedia so far is indeed on the metadata level: triples about the birth date of an author, the geographic location of a place and the list of people who were born there, etc. For search and recommendation of TV programs, movies or operas, triples representing the plot of the resource are crucial, and can also be extracted with Natural Language Processing techniques. VUA research proposes a (latin) language agnostic approach and architecture that generates URI annotations from multilingual texts, and apply it to Wikipedia pages in different languages. An internal use-case for this research activity is the semantic annotation of movies’ plot, with URIs from the Linked Open Data cloud, as one input for a semantic recommender system.
4.4.3.2. Use Cases Connection

The initial version of Lupedia was tested in the frame of the BBC’s use case scenario 7.c. The main objective of this use case is to follow users’ behaviour on various social networks and recommend content from the BBC programmes based on extracted user preferences. The main purpose of applying semantic enrichment services to this use case is to enrich the original program description. Lupedia was used to process the synopsis (as seen on the BBC website) of BBC programmes. The synopses were annotated in the background and the extracted entities were stored in a local semantic repository. They were made available to other partners at http://sparql-notube.ontotext.com/openrdf-workbench/repositories/notube-test/query.

Subsequently, the other use case leaders started using Lupedia and expressed requests for extensions. All new extensions and options are a direct result of such requests. Examples of such extensions are the options single greedy match only and case insensitive. They are particularly useful for looking up individual movie titles or other similar entities unconnected to running text, i.e. a task different from enrichment of programme descriptions as described above for the BBC use case.

4.4.3.3. Data Sets Connection

DBPedia\(^38\) is the main source of data for enrichment services. It’s a comprehensive multilingual resource and it serves as the central hub of the Linked Open Data (LOD) cloud. As such, it is well-linked to various other resources and was the logical choice for the central dictionary of Lupedia.

\(^{38}\) http://dbpedia.org

![Figure 18 – Linked Open Data Cloud Snapshot](http://dbpedia.org)
The data is loaded from FactForge\textsuperscript{39}, Ontotext's own LOD repository. The core technology used to build and search in the dictionary is the Large Knowledge Base Gazetteer (LKBG). More details on the technology and its implementation can be found in deliverable D4.4.

The London Classification (LonClass) is another potential dictionary source maintained by the BBC. The main challenge with LonClass is its complexity and representation, as it allows for complex semantic expressions but is not an RDF-based resource. The initial work for LonClass integration has started at the time of reporting.

With respect to the proposed enrichment research from VUA, all the terms found for plot description have been published internally\textsuperscript{40} as well as the enriched terms with DBPedia URL for plot descriptions\textsuperscript{41}. This service is still under development, but it will be connected to the enrichment services of the EPG Dataware house (already mentioned in section 4.4.2.1 about WP3 recommendation activities).

4.4.3.4. Integration

Lupedia is implemented as a RESTful services and it can produce output in four different formats:

- HTML (meant for humans)
- RDFa
- JSON
- XML

This has proven very flexible and it has met the requirements of all project partners.

Concerning WP1 research related to enrichment, the soon-to-be-released service will be TV-Anytime compliant. To enable a smooth integration VUA already worked together with IRT in order to define a TV-A format for the EPG data, as part of the metadata conversion task of WP2.

4.5. Content Services

4.5.1. Video Processing Services

4.5.1.1. Overview

TGV provides services for the automatic video reframing and for the ad insertion in NoTube. The automatic video reframing service automatically repurposes the video content, i.e. to adapt the image size to the screen size in a specific way, in order to provide a better viewing comfort. It takes in input a multimedia essence file (i.e. audio, video) and returns a multimedia essence file with the video that has been cropped.

The ad insertion service is used to insert advertising clips into a video sequence, with corresponds to the user’s interest and to insert it at the privileged moment and place of the video. It takes in input a movie file and an advertising file and returns the movie file with the ad inserted.

\textsuperscript{39} http://factforge.net, previously known as Linked Data Semantic Repository or LDSR

\textsuperscript{40} http://eculture2.cs.vu.nl/mlwiki/inspect.php

\textsuperscript{41} http://eculture2.cs.vu.nl/mlwiki/enrich.php
4.5.1.2. Use cases connection
The automatic video reframing Web Service will be integrated in use case 7.a. The automatic video reframing will analyse News video item, find the region of interest in the video content and crop the video to adapt it to the terminal capabilities.
The ad insertion Web Service, on the other hand, will be integrated in use case 7.b. iFanzy server requests the TGV platform to perform the ad insertion, through the Web Service, by giving in input, the movie file and the advertising file.

4.5.1.3. Data Sets Connection
The automatic video reframing service needs to have access to the corresponding content repository, to obtain the media file and to store the resulting, processed content.
The ad insertion service needs to have access to the corresponding content repository in use case 7.b, to obtain the movie file and the advertising file and to store the resulting file.

4.5.1.4. Integration
The format of all the video/audio files, used in automatic video reframing service and ad insertion service, shall be a MPEG2 transport stream with MPEG2 video elementary stream.

4.5.2. Audio Processing Services

4.5.2.1. Overview
IRT provides services for the analysis of loudness of media in NoTube. The service analyses multimedia essence files (i.e. audio, video) and returns the loudness of the media as metadata following new loudness harmonisation standards from ITU and EBU. In the further course of the project, the metadata can be used either to control the volume of playback devices or to pre-process the audio of the media file to realise a harmonic loudness for the listener/viewer.
In principle, it is possible to either adjust the loudness when the content is put into the system (i.e. as part of the ingest process), but it would also be possible to do it in the playback environment making use of the loudness metadata. In fact, this would probably be the easiest way by automatically adjusting the playback volume in the video player. However, this heavily depends on the video playback component individually implemented in the considered use case.

4.5.2.2. Use cases connection
Loudness harmonisation has been introduced as an opportunity to mitigate the audio differences between several multimedia contents (i.e.: TV programmes, ads) when merged together when played back by the user. This can be an improvement to the user experience harmonising, as the name of the feature suggests, heterogeneous audiovisual contents.
It has not been integrated in the NoTube scenarios due to the recent module availability. From an early evaluation it seems possible to adopt it in the scope of use cases where multimedia files consumed at end-user side play a central role. For instance News Items that are re-assembled in UC 7.a or ads in UC 7.b could potentially originate from various channels and broadcasters with different loudness levels (e.g. RAI, BBC, but also CNN, al-Djazeera etc.).

4.5.2.3. Data Sets Connection
The loudness analyser needs to have access to the corresponding content repository to obtain the media files which are to be analysed. The resulting loudness metadata needs to be added to the metadata sets corresponding to the media essence.

4.5.2.4. Integration
The loudness analyser service itself only supports audio in WAV format. Media essence in formats other than that need to be converted using an additional format conversion service.
A/V analysis are time-consuming processes. Using these analysis as services may require complex asynchronous message handling.

4.5.3. **Personal Video Recorder and Media Storage services**

4.5.3.1. **Overview**
TXT developed a Personal Video Recorder service for DVB-T broadcast. This service is fully programmable, it can be driven through a RESTful interface and it is also integrated with the service providing the EPG. It allows the user as well as automated services to schedule DVB-T programmes recording. The PVR service is complemented by a Media Storage service that stores the recorded programmes for future playback, and indexes them leveraging on NoTube metadata services.

4.5.3.2. **Use cases connection**
An instance of the Personal Video Recorder and Media Storage services is being used by the Personalised Semantic News Prototype WP7.a. The PVR is deployed in the Home Ambient module. The application logic of the Home Ambient monitors available broadcasts through EPG services, retrieves metadata from the Broadcaster through WP2 services, and matches them against the user profile through WP3 recommendation services. Whenever a relevant or recommended programme is scheduled for broadcast, the application logic sets the PVR accordingly in order to record the programme. The recorded video is then managed by Media Storage services and made available for playback as part of a playlist when selected from the front end.

4.5.3.3. **Data Sets Connection**
The PVR service uses services to retrieve detailed EPG entries for the scheduler. The Media Storage services accesses services on the side of the Service Provider to retrieve additional metadata for the recorded programme.

4.5.3.4. **Integration**
The PVR service needs to be connected to a hardware DVB-T tuner that allows video recording. It can record in MPEG2 DVR-MS and DVR-TS and it uses the AVI format as container.
5. Evaluation Planning

The evaluation of the three use case prototypes will be done according to the individual use cases functional requirements and thus will be planned and performed respectively in WP7.a, WPp7.b and WP7.c.

From the WP6 perspective the foreseen evaluation is not at application level, but rather at integration. In particular we plan evaluation for:

- Interface with legacy CMS
- Security and privacy preservation

5.1. Interface with legacy CMS

5.1.1. Goal

The main goal for this bullet is to understand how easy is to connect legacy CMS to the NoTube platform. This is an indicator of the work that shall be performed at the side of a TV broadcaster in order to connect its content assets to the NoTube platform.

This in turn involves some non-functional attributes described earlier on in D6.1.

We foresee two potential issues in this respect:

1. Legacy CMS are usually proprietary, closed software that the broadcaster is not likely to modify without a proper motivation.
2. At the technical level, the main obstacle is represented by proprietary contents (data and metadata) formats to be managed by the platform.

5.1.2. What, Where and How

In order to achieve the abovementioned goal the main actors involved are:

- The Broadcasters (WP7)
- Metadata Conversion Services (WP2)
- Advanced Audio/Video content Services (WP4)

The idea is to leverage on the NoTube applications development experience through the collection of feedback, by means of technical interviews, from broadcasters on one side and technicians on the other, about the two main issues we pointed out at the beginning of this section:

- The first one will be focused on the broadcaster internal environment, in particular its constraints in light of an integration with an external platform like NoTube
- The second one will target the technical developers in order to measure the effort required to implement ad-hoc converters/adapters to enable the data exchange between legacy systems and NoTube.

The acquired data will be useful to evaluate:

- Interoperability - refers to the ability of a collection of communicating entities (components) to share specific information and operate on it according to a commonly agreed operational semantics.
- Scalability - the ability to maintain performance when the system is changed in size or in volume in order to meet different needs.
- Extensibility - the ease with which the considered environment's features can be extended without affecting other parts of the system.
- Adaptability - the capability of a system to withstand and easily adapt to changes in its environment, requirements and implementation technologies.
5.2. **Security and privacy preservation**

5.2.1. **Goal**
The main aim is to ensure the privacy protection of the user by measuring potential weak points in the platform as well as the technical mechanisms adopted to prevent security flaws.

5.2.2. **What, Where and How**
The main actors are:
- The User Profile Management module
- NoTube services and applications

The evaluation will focus on collecting technical feedback about:
- Storage modalities for the user’s profile
- Storage modalities for the user’s activities
- Connection to external source (i.e.: Social Networks)
- Back-end interface of the User Profile Management module
- Techniques adopted to preserve security and privacy
  - Access to personal user details
  - Data exchange

The modality of the evaluation will be based on the simple low-level software “attacks”, basically trying to retrieve user profile details without providing credentials or trying to bypass the OAuth mechanism.

5.3. **Roadmap**
The evaluation of interfaces with legacy CMSs as well as security and privacy preservation will take place basing on the second integrated prototypes availability. WP7 internal plans foresees to freeze the major developments by the end of January 2011, reserving the three subsequent months (till the end of March 2011) to perform refinements, bug fixing and finalisation.

The final round prototypes are due M33, together with D6.4 “NoTube Integrated System 3rd Prototype”.

For the above reasons the WP6 evaluation activities could be following the modalities described in sections 5.1 and 5.2 as follows:
- M23 – Evaluation Planning (included in this document)
- M26 – 2nd Integrated Prototypes completion
- M25-M27 – First round of evaluation (focus on the 2nd Integrated Prototype)
- M33 – 3rd Integrated Prototypes completion
- M32-M34 – Second round of evaluation (focus on the 3rd Integrated Prototype)

The evaluation will start before completion of prototypes development towards the finalisation phases in order to collect updated feedback from the involved partners.
6. Conclusions

This document represents a companion to the second release of the NoTube integrated services platform developed in WP6 and also integrates the description of the integrated prototypes developed in WP7 and due M23. Its purpose is to provide an overview of the integrated services platform including the real-world integration approach, as well as individual achievements in terms of prototypes/services integration for the three use cases.

This deliverable provides also a high level view of the developed and exploited services for the main categories envisaged in NoTube, in each prototype, and according to the previously occurred design phase. The internal evaluation plan in terms of actors, goals, target groups and methodology has been presented as well focusing on field-specific issues like legacy systems and data/privacy protection, as part of the very next steps in the WP6 activities.

The NoTube services platform is the result of a design and implementation work that accommodates requirements originating from several different stakeholders (e.g. service developers, use case designers, broadcasters, end users) and integrates heterogeneous technologies into a unified reference framework. This process required an effort from both a technical design and coordination perspective.

From a technical perspective the main research challenges that we faced, described in the document, were the integration of heterogeneous software modules involving different technologies, data repositories and, in turn, potentially different data formats. This also includes the procedures for interfacing legacy Content Management Systems with NoTube semantic services, a fundamental issue in the adoption of NoTube.

From a coordination perspective, during our work we closely monitored activities both in the R&D work packages (WP1-5) and the use cases work packages (WP7a/b/c). This continuous communication and coordination activity facilitated the individual prototypes implementation as well as the development of a set of platform services in line with both the project goals and the connection with application scenarios. Tight cooperation with activities in WP5 was also instrumental in creating a service base that supports brokerage, and, in turn, a use cases design that is decoupled from low level services thanks to the Service Broker.

Similarly to the previous released document (D6.2), the presented content is subject to be revised in the final issue due at the end of the project reflecting the final evolution of the services base and the three prototypes.