Web Semantics: Science, Services and Agents on the World Wide Web 6 (2008) 250–256

Contents lists available at ScienceDirect

Web Semantics: Science, Services and Agents on the World Wide Web

journal homepage: www.elsevier.com/locate/websem

# FOAFing the music: Bridging the semantic gap in music recommendation☆

Òscar Celma*, Xavier Serra

*Universitat Pompeu Fabra, Barcelona, Spain*

ARTICLE INFO

*Article history:*
Received 13 May 2008
Received in revised form 6 September 2008
Accepted 15 September 2008
Available online 23 October 2008

*Keywords:*
Music recommendation
Music 2.0
Semantic Web
Hybrid recommender
FOAF
Long tail

ABSTRACT

In this paper we give an overview of the *Foafing the Music* system. The system uses the *Friend of a Friend* (FOAF) and *RDF Site Summary* (RSS) vocabularies for recommending music to a user, depending on the user's musical tastes and listening habits. Music information (new album releases and reviews, podcast sessions, audio from MP3 blogs, related artists' news, and upcoming gigs) is gathered from thousands of RSS feeds. The presented system provides music discovery by means of: user profiling (defined in the user's FOAF description), context-based information (extracted from music related RSS feeds) and content-based descriptions (extracted from the audio itself), based on a common ontology (OWL DL) that describes the music recommendation domain. The system is available at: http://foafing-the-music.iua.upf.edu.



## 1. Introduction

The World Wide Web has become the host and distribution channel for a broad variety of digital multimedia assets. Although the Internet infrastructure allows simple straightforward acquisition, the value of these resources lacks powerful content management, retrieval and visualisation tools. Music content is no exception; although there is a sizeable amount of text-based information related to music (album reviews, artist biographies, etc.) this information is hardly ever associated with the objects it refers to, that being the music files themselves (MIDI and/or audio). Moreover, music is an important vehicle for communicating to other people something relevant about our personality, history, etc.

In the context of the Semantic Web, there is a clear interest to create a Web of machine-readable homepages describing people, the links among them, and the things they create and do. The FOAF (*Friend Of A Friend*) project[1] provides conventions and a language to describe homepage-like content and social networks. FOAF is based on the RDF/XML[2] vocabulary. We can foresee that with the user's FOAF profile, a system would get a better representation of the user's musical needs. On the other hand, the RSS vocabulary[3] allows systems one to syndicate Web content on the Internet. Syndicated content includes data such as news, event listings, headlines, project updates, as well as music related information, such as new music releases, album reviews, podcast sessions, upcoming gigs, etc.

## 2. Background

In this section we present the concepts and technologies that are the basis of the proposed system. In Section 2.1 we introduce the music information plane concept, as well as the existing semantic gap in the music domain. Then, Section 2.2 introduces the basic concepts of music recommendation, and presents the most common approaches — that is collaborative filtering, and content-based recommendations. Section 2.3 provides and overview of some of the existing music recommendation systems. Finally, we present in Section 2.4 the current attempts to model user profiles and user preferences, and how to adapt them to take into account music related information.

### 2.1. The music information plane

In the past 20 years, the signal processing and computer music communities have developed a wealth of techniques and

☆ This work was partially funded by the SIMAC IST-FP6-507142 European project.

* Corresponding author. Tel.: +34 935 422 199.
*E-mail addresses:* oscar.celma@upf.edu, ocelma@iua.upf.edu (Ò. Celma), xavier.serra@upf.edu (X. Serra).

[1] http://www.foaf-project.org.

[2] http://www.w3.org/RDF.

[3] http://web.resource.org/rss/1.0/.

1570-8268/$ – see front matter © 2008 Elsevier B.V. All rights reserved.
doi:10.1016/j.websem.2008.09.004

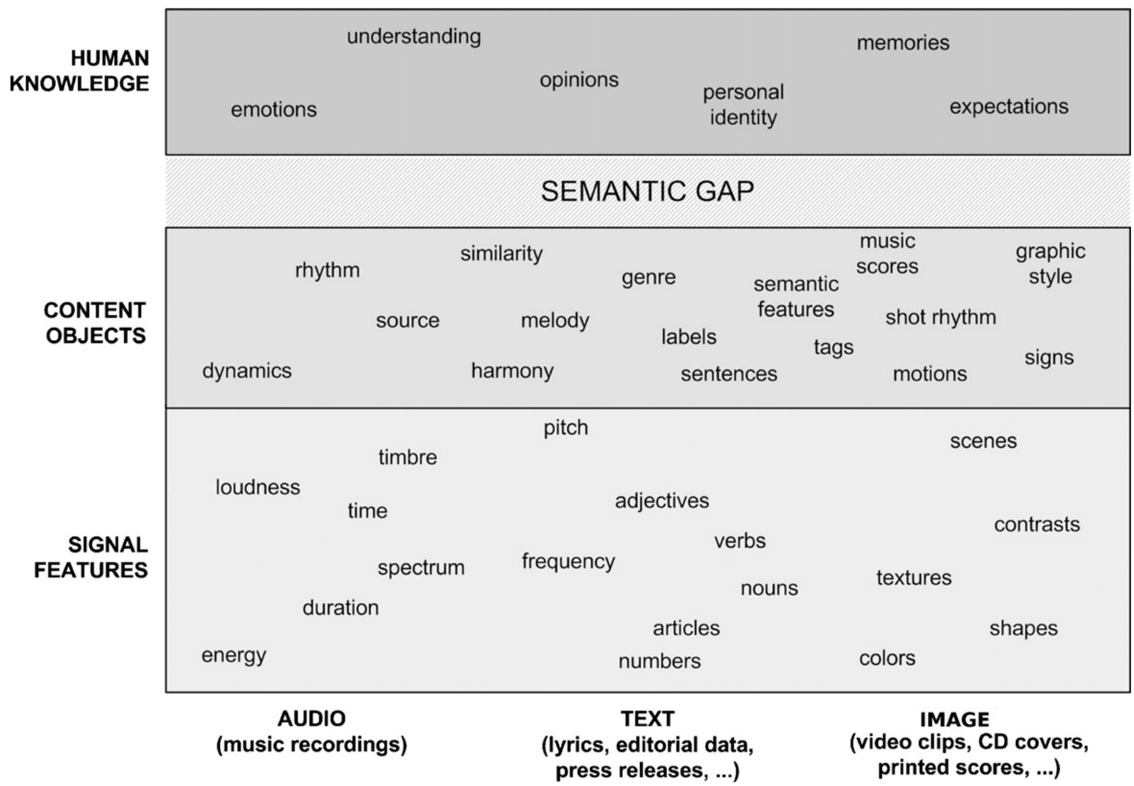


**Fig. 1.** The music information plane and its semantic gap between users (human understanding) and content object descriptions.

technologies to describe audio and music content at the lowest (or close-to-signal) level of representation. However, the gap between these low-level descriptors and the concepts that music listeners use to relate with music collections (the so-called "semantic gap") is still, to a large extent, waiting to be bridged.

Due to the inherent complexity needed to describe multimedia objects, a layered approach with different levels of granularity is needed when designing an ontology for a particular domain. Depending on the requirements, one might choose the appropriate level of abstraction. In the multimedia field and, specially, in the music field we foresee three levels of abstraction: low-level basic features, mid-level semantic features, and human understanding. The first level includes physical features of the objects, such as the sampling rate of an audio file, as well as some basic features like the spectral centroid of an audio frame, or even the predominant chord in a sequential list of frames. A high-level of abstraction aims to describe concepts such as a guitar solo, or tonality information (e.g. key and mode) of a music title. Finally, the higher level should use reasoning methods and semantic rules to retrieve, for instance, several audio files with "similar" guitar solos over the same key.

Similarly, we describe the music information plane in two dimensions (see Fig. 1). One dimension takes into account the different media types that serve as input data. The other dimension is the level of abstraction in the information extraction process of this data. The input media types include data coming from: audio (music recordings), text (lyrics, editorial text, press releases, etc.) and image (video clips, CD covers, printed scores, etc.). On the other hand, for each media type there are different levels of information extraction. The lowest level is located at the signal features. This level lies far from what an end-user might find meaningful. Regardless, it is the basis that allows one to describe the content and to produce more elaborate descriptions of the media objects. This level includes basic audio features, such as: energy, frequency, and mel frequency cepstral coefficients to describe the timbral properties of the sound, and also basic natural language processing for the textual information. At the mid-level (the content objects), the information extraction process and the elements described are closer to the end-user. This level includes a description of musical concepts (e.g. rhythm, harmony, melody), or named entity recognition for text information. Finally, the higher-level includes information tightly related with the users who are interacting with music knowledge (opinions, emotions, memories, etc.). It is clear, then, that the semantic gap is located in this area: the descriptions that can be extracted from the sources (audio, text, image) are not yet close enough to the end-user. This limits the users' interaction with music and audio content. Fig. 1 depicts the music information plane.

In this paper, we focus on leveraging the semantic gap in the context of music recommendation. To achieve this, we propose a music recommendation ontology that allows one to describe the music content (songs, artists, etc.). We use the FOAF ontology to model user profiles and user preferences.

### 2.2. *Music recommendation*

The main goal of a music recommendation system is to propose to the end-user interesting and unknown music artists (and their available tracks, if possible), based on user's musical taste. But musical taste and music preferences are affected by several factors, even demographic and personality traits. Thus, the combination of music preferences and personal aspects — such as: age, gender, origin, occupation, musical education, etc. — could improve music recommendations [12].

Moreover, a desirable property of a music recommendation system should be the ability to dynamically retrieve new music related information, as it should recommend new items to the user. In this sense, there is frequently more freely available (in terms of licensing) music on Internet, performed by "unknown" artists that can perfectly suit for new recommendations. Nowadays, music websites are informing the user about new releases or artist's related news, mostly in the form of RSS feeds. For instance, iTunes Music Store provides an RSS (version 2.0) feed generator,[4] updated once a week, that publishes new releases of artists' albums. A music recommendation system should take advantage of these publishing services, as well as integrating them into the system, in order to filter and recommend new music to the user.

#### 2.2.1. Music recommendation algorithms

The most commonly used music recommendation methods are collaborative-filtering, and audio content-based analysis. The collaborative filtering method consists of making use of feedback from users to improve the quality of recommended material presented to users. Obtaining feedback can be explicit or implicit. Explicit feedback comes in the form of user ratings or annotations, whereas implicit feedback can be extracted from the user's listening habits. The main caveats of this approach are the following: the *cold-start* problem, the novelty detection problem, the item popularity bias, and the enormous amount of data (i.e. users and items) needed to get some reasonable results [6]. There are some examples that succeed based on this approach. For instance, *Last.fm* [5] and *Amazon* [7] are two illustrative systems.

Audio content-based filtering tries to extract useful information from the music collection, that could be useful to represent a user's musical taste. This approach solves the limitation of collaborative filtering as it can recommend new items (even before the system knows anything about that item), by comparing the actual set of user items and calculating a distance with a relevant similarity measure. In the music field, extracting musical semantics from the raw audio and computing similarities between music pieces is a challenging problem. In [9], Pachet proposes a classification of musical metadata, and describes how this classification affects music content management, as well as the problems faced when defining a ground truth reference for music similarity (both in collaborative and content-based filtering).

### 2.3. Related systems

Most of the current music recommenders are based on a collaborative filtering approach. Examples of such systems are: Last.fm, MyStrands,[6] MusicMobs,[7] Goombah Emergent Music,[8] iRate,[9] and inDiscover,[10] based on the Racofi collaborative filtering system [1]. The basic idea of a music recommender system based on collaborative filtering is

(1) To keep track of which artists (and songs) a user listens to — through iTunes, WinAmp, Amarok, XMMS, etc. plugins,
(2) To search for other users with similar tastes, and
(3) To recommend artists (or songs) to the user, according to these similar listeners' taste.

Likewise, the most notable system using (manually annotated) content-based descriptions to recommend music is *Pandora*.[11] The main problem of the system is the scalability, because the music annotation process is very time consuming.

Contrastingly, the *Foafing the Music* approach focuses on how to recommend, discover and explore music content; from context-based information (extracted from music related RSS feeds), and content-based descriptions (automatically extracted from the audio itself [2]), while being based on a common ontology that describes the musical domain. To our knowledge, nowadays does not exist any system that recommends items to a user based on FOAF profiles. Yet, there is the *FilmTrust* system.[12] It is part of a research study aimed at understanding how social preferences might help web sites to present information in a more useful way. The system collects user reviews and ratings about movies, and stores them into the user's FOAF profile.

### 2.4. User profiling

Music recommendation is highly dependant on the user. The user modelling process is a crucial step in understanding user preferences. However, in the music field, there have been few attempts to explicitly extend user profiles by adding music related information. Yet, it is an interesting way to communicate with other people, and to express music preferences (it is very common to embed in a webpage a simple widget that displays the most recent tracks a user has played). Music is an important vehicle for conveying to others something relevant about our personality, history, etc. The music information to be added in the profile should be related with the user's interests and habits.

The most relevant proposals in this context are: the User modelling for Information Retrieval Language, the related description schemas defined by the MPEG-7 standard, and the Friend of a Friend (FOAF) initiative — hosted by the Semantic Web community. The complexity in terms of semantics increases with each proposal. The following sections present these approaches.

#### 2.4.1. User modelling for information retrieval (UMIRL)

The UMIRL language, proposed by Chai and Vercoe [3], allows one to describe perceptual and qualitative features of the music. It is specially designed for music information retrieval systems. The profile can contain both demographic information and direct information about the music objects: favourite bands, styles, songs, etc. Moreover, a user can add his definition of a perceptual feature, and his meaning, using music descriptions. For instance: "a *romantic piece* has a slow tempo, lyrics are related with *love*, and has a soft intensity, and the context to use this feature is while having a special dinner with user's girlfriend".

The representation they proposed uses the XML syntax, without any associated schema or document type definition to validate the profiles. Listing 1 shows a possible user profile.

[4] http://phobos.apple.com/WebObjects/MZSearch.woa/wo/0.1.
[5] http://www.last.fm.
[6] http://www.mystrands.com.
[7] http://www.musicmobs.com.
[8] http://goombah.emergentmusic.com/.
[9] http://irate.sourceforge.net.
[10] http://www.indiscover.net/.
[11] http://www.pandora.com/.
[12] http://trust.mindswap.org/FilmTrust.

```
<user>
 <generalbackground>
   <name>Joan Blanc</name>
   <education>MsC</education>
   <citizen>Catalan</citizen>
 </generalbackground>
 <musicbackground>
   <education>none</education>
   <instrument>guitar</instrument>
 </musicbackground>
 <musicpreferences>
   <genre>rock</genre>
   <album>
     <title>To bring you my love</title>
     <artist>P.J. Harvey</artist>
   </album>
 </musicpreferences>
</user>
```

Listing 1. Example of a user profile in UMIRL.

This proposal is one of the first attempts in the Music Information Retrieval community. The main goal was to propose a representation format, as a way to interchange profiles among systems, though, it lacks formal semantics to describe the meaning of their descriptors and attributes. To cope with this limitation, the following section presents an approach by using the descriptors defined in the MPEG-7 standard.

#### 2.4.2. MPEG-7 user preferences

MPEG-7, formally named Multimedia Content Description Interface, is an ISO/IEC standard developed by the Moving Picture Experts Group (MPEG). The main goal of the MPEG-7 standard is to provide structural and semantic description mechanisms for multimedia content. The standard provides a set of description schemes (DS) to describe multimedia assets. In this paper, we only focus on the descriptors that describes user preferences of multimedia content, while a concise description of the whole standard is presented in [8].

User preferences in MPEG-7 include content filtering, searching and browsing preferences. The usage history, which represents the user history of interaction with multimedia items, can be denoted too. Filtering and searching preferences include the user preferences regarding classification (i.e. country of origin, language, available reviews and ratings, reviewers, etc.) and creation preferences. The creation preferences describe the creators of the content (e.g. favourite singer, guitar player, composer, and music bands). Also, it allows one to define a set of keywords, location and a period of time. Using a preference value attribute, the user can express positive (likes) and negative (dislikes) preferences for each descriptor. The following example shows a hypothetical user profile definition, stating that she likes the album *"To bring you my love"* from *P.J. Harvey*:

```
<UserPreferences>
 <UserIdentifier protected="true">
  <Name xml:lang="ca">Joan Blanc</Name>
 </UserIdentifier>
 <FilteringAndSearchPreferences>
  <CreationPreferences>
   <Title preferencValue="8">To bring you my love</Title>
   <Creator>
    <Role>
     <Name>Singer</Name>
    </Role>
    <Agent xsi:type="PersonType">
     <Name>
      <GivenName>Polly Jean</GivenName>
      <FamilyName>Harvey</FamilyName>
     </Name>
    </Agent>
   </Creator>
   <Keyword>dramatic</Keyword>
   <Keyword>fiery</Keyword>
   <DatePeriod>
    <TimePoint>1995-01-01</TimePoint>
    <Duration>P1825D</Duration>
   </DatePeriod>
  </CreationPreferences>
 </FilteringAndSearchPreferences>
</UserPreferences>
```

Listing 2. Example of a user profile in MPEG-7.

MPEG-7 usage history is defined following the usage history description scheme. *UsageHistory DS* contains a history of user actions. It contains a list of actions (play, play-stream, record, etc.), with an associated observation period. The action has a program identifier (an identifier of the multimedia content for which the action took place) and, optionally, a list of related links or resources.

In [11], Tsinaraki and Christodoulakis present a way to overcome some of the limitations of describing user preferences in MPEG-7. They argue that there is still a lack of semantics when defining user preferences, as the whole MPEG-7 standard is based on XML Schemas. For example, filtering and search preferences allow one to specify a list of textual keywords, without being related to any taxonomy nor ontology. Their implementation is integrated into a framework based on an upper ontology that covers the MPEG-7 multimedia description schemes. That upper ontology uses the OWL notation, so it does the next proposal, based on the FOAF initiative.

#### 2.4.3. FOAF: user profiling in the Semantic Web

The FOAF (Friend Of A Friend) project provides conventions and a language "to tell" a machine the type of things a user says about himself in his homepage. FOAF is based on the RDF/XML vocabulary. As we noted before, the knowledge held by a community of "peers" about music is also a source of valuable metadata. FOAF nicely allows one to easily relate and connect people.

FOAF profiles include demographic information (name, gender, age, sex, nickname, homepage, depiction, web accounts, etc.) geographic (city and country, geographic latitude and longitude), social information (relationship with other persons), pyschographic (i.e. user's interests) and behavioural (usage patterns). There are some approaches that allow modelling music taste in a FOAF profile. The Semantic Web approach facilitates the integration of different ontologies. Listing 3 shows how to express that a user likes an artist, using the general *Music Ontology* proposed by Giasson and Raimond in [5].

```
<foaf:interest>
 <mo:MusicArtist rdf:about="http://zitgist.com/music/
   artist/ca37-...fc">
  <mo:discogs rdf:resource="http://www.discogs.com/artist
    /PJ+Harvey"/>
  <foaf:img rdf:resource="http://ec2.images-amazon.com/

    images/P/B00852Q....jpg"/>
  <foaf:homepage rdf:resource="http://pjharvey.net/"/>
  <foaf:name>P.J. Harvey</foaf:name>
  <mo:wikipedia rdf:resource="http://en.wikipedia.org/
    wiki/PJ_Harvey"/>
 </mo:MusicArtist>
</foaf:interest>
```

Listing 3. FOAF example of an artist description that a user is interested in, using the general Music Ontology.

## 3. System overview

The overview of the *Foafing the Music* system is depicted in Fig. 2. The system is divided in two main components, that is (i) how to gather data from external third party sources (presented in Section 3.1), and (ii) how to recommend music to the user based on the crawled data, and the semantic description of the music titles (Section 3.2).

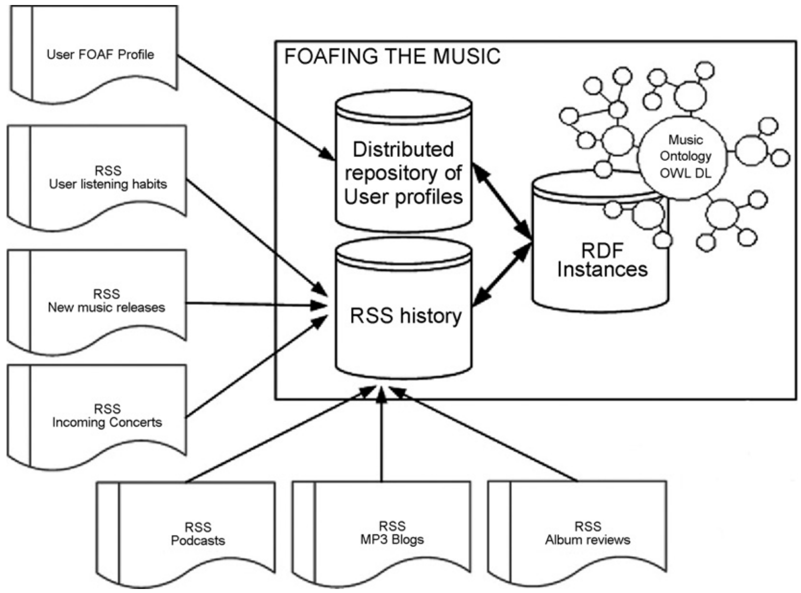


**Fig. 2.** Architecture of the *Foafing the Music* system.

### 3.1. Gathering music related information

Personalised services can raise privacy concerns due to the acquisition, storage and application of sensitive personal information [10]. In our system, information about the user is not stored in the system in any way. Instead, the system has only a link pointing to the user's FOAF profile (in many cases a link to a *Livejournal* account). Thus, the sensitivity of this data is up to the user, not to the system. Users' profiles in *Foafing the Music* are distributed over the net.

Regarding music related information, our system exploits the mashup approach. The system uses a set of public available APIs and web services sourced from third party websites. This information can come in any of the different RSS formats (v2.0, v1.0, v0.92 and Yahoo! Media RSS), as well as in the Atom format. Thus, the system has to deal with syntactically and structurally heterogeneous data. Moreover, the system keeps track of all the new items that are published in the feeds, and stores the new incoming data in a historic relational database. Input data of the system is based on the following information sources:

- *User listening habits*. To keep track of the user's listening habits, the system uses the services provided by *last.fm*. This system offers a list of RSS feeds that provide the most recent tracks a user has played. Each item feed includes the artist name, the song title, and a timestamp — indicating when the user has listened to the track.
- *New music releases*. The system uses a set of RSS feeds that gathers new music releases from *iTunes*, *Amazon*, *Yahoo! Shopping* and *Rhapsody*.
- *Upcoming concerts*. The system uses a set of RSS feeds that syndicates music related events. The websites are: *Eventful.com*, *Upcoming.org*, and *Sub Pop* record label.[13] Once the system has gathered the new items, it queries the Google Maps API to get the geographic location of the venues.
- *Podcast sessions*. The system gathers information from a list of RSS feeds that publish podcasts sessions.

**Table 1**
Information gathered from music related RSS feeds is stored into a relational database.

| Source | # RSS seed feeds | # Items stored |
|---|---|---|
| New releases | 44 | 426,839 |
| MP3 blogs | 127 | 600,838 |
| Podcasts | 830 | 146,922 |
| Album reviews | 12 | 127,367 |
| Upcoming concerts | 14 | 292,526 |

Based on the user's FOAF profile, the system filters this information, and presents the most relevant items according to her musical taste.

- *MP3 blogs*. The system gathers information from a list of MP3 blogs that talk about artists and new music releases.
- *Album reviews*. Information about album reviews are crawled from the RSS feeds published by *Rateyourmusic.com*, *Pitchforkmedia.com*, online magazines *Rolling Stone*,[14] *BBC*,[15] *New York Times*,[16] and *75 or less records*.[17]

Table 1 shows some basic statistics of the data that has been gathered since mid April, 2005 until the first week of May, 2008. These numbers show that the system has to deal with daily incoming data.

On the other hand, we have defined a simple music recommendation OWL DL ontology (http://foafing-the-music.iua.upf.edu/music-ontology#) that describes some basic properties of the artists and music titles, as well as some descriptors automatically extracted from the audio files (e.g. tonality, rhythm, moods, music intensity, etc.). In [4] we propose a way to map our ontology and the *Musicbrainz* ontology, onto the MPEG-7 standard, which acts as an upper-ontology for multimedia description. This way we can link our dataset with the *Musicbrainz* information in a straightforward manner.

A focused web crawler has been implemented in order to add instances to our music ontology. The crawler extracts metadata of

[13] http://www.subpop.com/.

[14] http://www.rollingstone.com/.
[15] http://www.bbc.co.uk/.
[16] http://www.nytimes.com/.
[17] http://www.75orless.com/.

artists and songs, and the relationships between artists (such as: “related with”, “influenced by”, “followers of”, etc.), and converts it to RDF/XML notation. The seed sites to start the crawling process are music metadata providers, such as *MP3.com*, *Yahoo! Music*, and *RockDetector*, as well as independent music labels (*Magnatune*, *CDBaby*, *GarageBand*, etc.). Based on our lightweight music recommendation ontology, Listing 4 shows the RDF/XML description of an artist from *GarageBand*.

```
<rdf:Description rdf:about="http://www.garageband.com/
   artist/randycoleman">
 <rdf:type rdf:resource="&music;Artist"/>
 <foaf:name>Randy Coleman</foaf:name>
 <music:decade>1990</music:decade>
 <music:decade>2000</music:decade>
 <music:genre>Pop</music:genre>
 <foaf:based_near
   rdf:resource="http://sws.geonames.org/5368361/"/>
 <music:influencedBy
   rdf:resource="http://www.coldplay.com"/>
 <music:influencedBy
   rdf:resource="http://www.jeffbuckley.com"/>
 <music:influencedBy
   rdf:resource="http://www.radiohead.com"/>
</rdf:Description>
```

Listing 4. RDF example of an artist individual

Listing 5 shows the description of an individual track of the above artist, including basic editorial metadata, and some features extracted automatically from the audio file.

```
<rdf:Description rdf:about="http://www.garageband.com/
   song?|pe1|S8LTM0LdsaSkaFeyYG0">
 <rdf:type rdf:resource="&music;Track"/>
 <music:title>Last Salutation</music:title>
 <music:playedBy rdf:resource="http://www.garageband.com/
    artist/randycoleman"/>
 <music:duration>247</music:duration>
 <music:intensity>Energetic</music:intensity>
 <music:key>D</music:key>
 <music:keyMode>Major</music:keyMode>
 <music:tonalness>0.84</music:tonalness>
 <music:tempo>72</music:tempo>
</rdf:Description>
```

Listing 5. Example of a track individual

These individuals are used in the recommendation process, to retrieve artists and songs related with the user’s musical taste.

### 3.2. Music recommendation process

This section explains the music recommendation process, based on all the information that has continuously been gathered from the RSS feeds and the crawler. Music recommendations, in the *Foafing the Music* system, are generated according to the following steps:

(1) Get music related information from user’s FOAF interests, and listening habits from *last.fm*,
(2) Detect artists and bands,
(3) Compute similar artists, and
(4) Rate the results by relevance, according to the user’s profile.

In order to gather music related information from a FOAF profile, the system extracts the information from the FOAF interest property (if `dc:title` is given then it gets its value, otherwise it gathers the text from the `<title>` tag of the HTML resource).

```
<foaf:interest
  rdf:resource="http://www.tylaandthedogsdamour.com/"
  dc:title="The␣Dogs␣d'Amour" />
```

Listing 6. Example of a FOAF interest with a given `dc:title`.

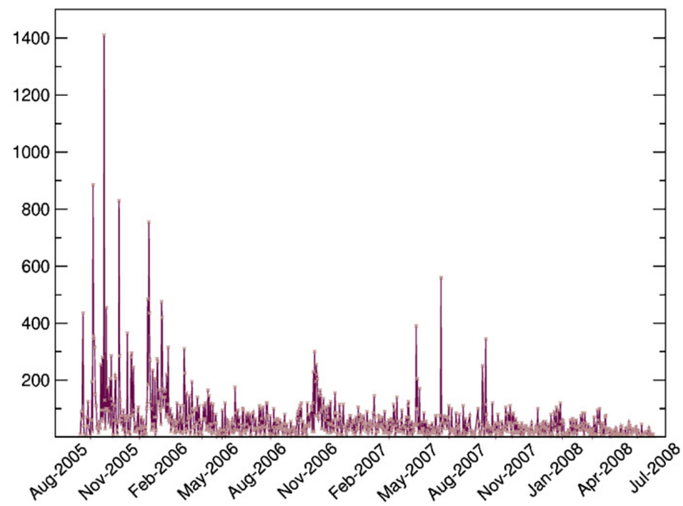


**Fig. 3.** Daily accesses to *Foafing the Music*. The system has an average of 60 daily unique accesses, from more than 4,000 users.

The system can also extract information from a user’s FOAF interest that includes the artist description based on the general Music Ontology [5](see Section 2.4).

Based on the music related information gathered from the user’s profile and listening habits, the system detects the artists and bands that the user is interested in, by doing a SPARQL query to the artist RDF repository. Once the user’s artists have been detected, artist similarity is computed. This process is achieved by exploiting the RDF graph of artists’ relationships (e.g. *influenced by*, *followers of*, *worked with*, etc.), as shown in Listing 4. The system offers two ways of recommending music information. *Static* recommendations are based on the favourite artists encountered in the FOAF profile. We assume that a FOAF profile would be rarely updated or modified. On the other hand, *dynamic* recommendations are based on user’s listening habits, which are updated much more often than the user’s profile. With this approach the user can discover a wide range of new music and artists on a daily basis.

Once the recommended artists have been computed, *Foafing the Music* filters music related information coming from the gathered music information (see Section 3.1) in order to:

- Get new music releases from iTunes, Amazon, Yahoo Shopping, etc.
- Download (or stream) audio from MP3-blogs and Podcast sessions,
- Create, automatically, XSPF[18] playlists based on audio similarity,
- View upcoming gigs happening near to the user’s location, and
- Read album reviews.

Syndication of the website content is done via an RSS 1.0 feed. For most of the above mentioned functionalities, there is a user’s feed subscription option to get the results — in the RSS format, that includes the instances (i.e. artists) from our music recommendation ontology.

### 3.3. Usage data

Since its inception in August 2005, the system has an average of 60 daily unique accesses, from more than 4000 active users. More than half of the users automatically created an account using an

[18] http://www.xspf.org/. XSPF is a playlist format based on XML syntax.

external FOAF profile (most of the times, around 70%, the profile came from their *Livejournal* FOAF account). Also, more than 65% of the users add her *last.fm* account, so we can use their listening habits from *last.fm*. Fig. 3 shows the number of logins over time, since August 2005 till July 2008. The peaks are clearly correlated with related news about the project (e.g. local TV and radio interviews, and reviews on the web).

## 4. Conclusions

Describing music assets is a crucial task for any music recommender system. The *success* of a music recommender can depend on the accuracy, and level of detail of this semantic information. Furthermore, formalising some musical concepts into an ontology — that allows us to describe the musical assets involved in the recommendation process — and linking them with the user profile, eases the recommendation process.

In this sense, we have proposed a system that filters music related information, based on a given user's FOAF profile, and her listening habits. A system based on FOAF profiles and user's listening habits allows one to "understand" a user in two complementary ways; psychological factors — personality, demographic preferences, social relationships — and explicit musical preferences. In the music field, we expect that filtering information about new music releases, artists' interviews, album reviews, and so on, can improve user satisfaction. It is clear that high-level musical descriptors facilitate more accurately content retrieval, and personalised recommendations. Thus, going one step beyond, it would be desirable to combine mid-level acoustic metadata with as much editorial and cultural metadata as possible. From this combination, more sophisticated inferences and semantic rules would be possible. These rules could derive hidden high-level metadata that could be, then, easily understood by the end-user, and thus enhancing their profiles. Yet, user (or domain expert) intervention would be needed, at least to validate that the induced rules make sense. We foresee that exploiting the general Music Ontology [5], as well as using all the linked information available in the Web of Data, we follow the right path to achieve a truly semantically-enhanced music recommender.

*Foafing the Music* is accessible through http://foafing-the-music.iua.upf.edu.

## References

[1] M. Anderson, M. Ball, H. Boley, S. Greene, N. Howse, D. Lemire, S. McGrath, Racofi: a rule-applying collaborative filtering system. In: Proceedings of COLA'03. IEEE/WIC, October 2003.
[2] O. Celma, P. Cano, P. Herrera, Search sounds: an audio crawler focused on weblogs, in: Proceedings of 7th International Conference on Music Information Retrieval, Victoria, Canada, 2006.
[3] W. Chai, B. Vercoe, Using user models in music information retrieval systems, in: Proceedings of 1st International Conference on Music Information Retrieval, 2000.
[4] R. Garcia, O. Celma, Semantic integration and retrieval of multimedia metadata, in: Proceedings of 4rd International Semantic Web Conference, Knowledge Markup and Semantic Annotation Workshop, Galway, Ireland, 2005.
[5] F. Giasson, Y. Raimond, Music ontology specification, working draft, February 2007.
[6] J.L. Herlocker, J.A. Konstan, L.G. Terveen, J.T. Riedl, Evaluating collaborative filtering recommender systems, ACM Trans. Inf. Syst. 22 (1) (2004) 5–53.
[7] G. Linden, B. Smith, J. York, Amazon. com recommendations: item-to-item collaborative filtering, IEEE Internet Comput. 4 (1) (2003).
[8] B.S. Manjunath, P. Salembier, T. Sikora, Introduction to MPEG 7: Multimedia Content Description Language, Wiley, 2002.
[9] F. Pachet, Knowledge Management and Musical Metadata, Idea Group, 2005.
[10] E. Perik, B. de Ruyter, P. Markopoulos, B. Eggen, The sensitivities of user profile information in music recommender systems, in: Proceedings of Private, Security, Trust, 2004.
[11] C. Tsinaraki, S. Christodoulakis, Semantic user preference descriptions in mpeg-7/21, 2005.
[12] A. Uitdenbogerd, R. van Schnydel, A review of factors affecting music recommender success, in: Proceedings of 3rd International Conference on Music Information Retrieval, Paris, France, 2002.