A scalable framework for multimedia knowledge management

Yves Raimond, Samer Abdallah, Mark Sandler, Mounia Lalmas

Centre for Digital Music, Queen Mary, University of London
Department of Computer Science, Queen Mary, University of London

December 7, 2006

Introduction - Understanding of MM materials

Multimedia information retrieval systems need to provide an *understanding* of the multimedia materials using either:

Manual annotations "There is a person in the middle of this picture, and this person is my father"

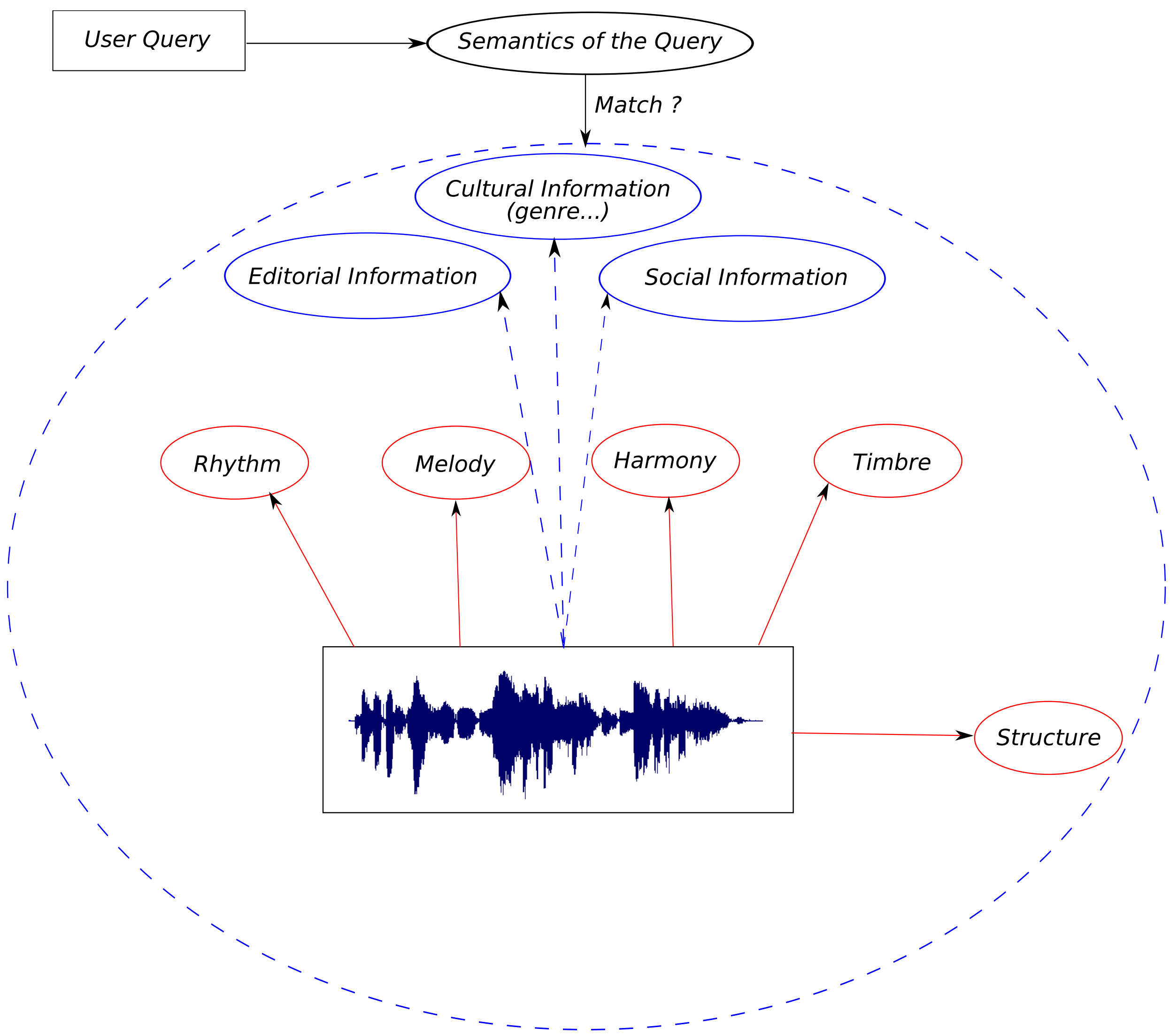
Automatic understanding "I am a really clever algorithm, and if I look deeply into this signal, I can see a pattern which looks similar to what I know is the father of this guy"

Cultural information "Most of the people associate this track with what they call *garage rock*, I may call this information *genre*"

Social information "I like these songs, you like almost the same ones, if you like another song, I may like it as well!"

Introduction - MM information retrieval

In the case of *music information retrieval* [Pachet, 2005]:



Problem 1 - Shared knowledge environment

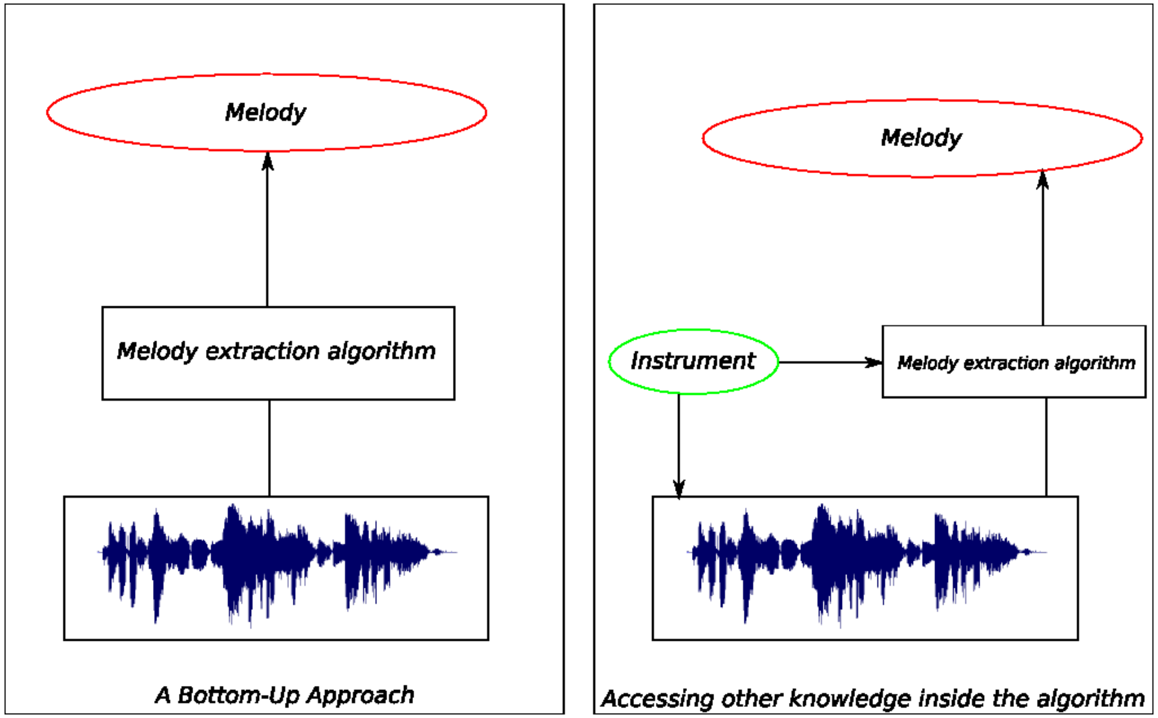
Several *actors* in the knowledge assertion process:

- Databases (Musicbrainz [Swartz, 2002], FreeDB...)
- Automatic concept detectors
- Humans...

We need a shared knowledge environment!

Problem 2 - Accessing knowledge *inside* algorithms

Manual annotations are quite accurate, because people have *culture*. **What about algorithms??**



We will try to design a framework where algorithms can access this shared knowledge environment...

Context sensitivity of processes: Knowledge + Data Analysis = New Knowledge

Overview

- **1 - The ‘knowledge machine’ framework**
  - a) Knowledge representation for data analysis
  - b) Evaluation engines
  - c) Function tabling
- **2 - Interaction with a shared knowledge environment**
  - a) A semantic-web knowledge environment
  - b) Interpreting semantic-web knowledge inside the ‘knowledge machines’
  - c) Exporting knowledge to the semantic-web environment
  - d) Implementation
- **3 - Two use cases**
  - a) Enhanced workspace for multimedia processing researchers
  - b) End-user information access
- **Conclusion and further work**

The ‘knowledge machine’ framework

- a) Knowledge representation for data analysis
- b) Evaluation engines
- c) Function tabling

Knowledge representation for data analysis

- **Dictionary approach**
  - Key/value pairs
  - Matlab workspace
  - File system, hierarchical structure of keys

- **Semantics of results?**
  We need knowledge about which function was used, which result, what parameters...

- **Towards predicate calculus for such knowledge representation [Abdallah et al., 2006]**
  Facts or *composite formulæ* involving the logical connectives *if*, $\exists$ (*exists*), $\forall$ (*for all*), $\vee$ (*or*), $\wedge$ (*and*), $\neg$ (*not*) and $\equiv$ (*equivalent to*)

  '*this* spectrogram was computed from *this* signal using *these* parameters' could be represented as:

  spectogram(DigitalSignal,FrameSize,HopSize,Spectrogram)

Evaluation engines

Such predicates, used in a given *mode*, may hide calls external *evaluation engines*

- Matlab
- C/C++ libraries
- ...

This can be done either directly, or through an external interpreter (such as Matlab).

- *is* operator in standard prolog
- Operator === evaluates terms representing Maltab expressions
- A matrix multiplication:

mtimes(A,B,C) _if_ C===A * B

Function tabling

- **We consider *tabling* [Sagonas et al., 1994] of resulting *facts***
  - 0) I have one object x
  - 1) I may call an evaluation engine to evaluate *mypredicate*(x,B)
  - 2) *mypredicate*(x,y) is tabled
  - 3) *mypredicate*(x,B) will bind B to y without any other evaluation
  - 4) *mypredicate*(A,y) will bind A to x: **Reverse-engineering of objects**

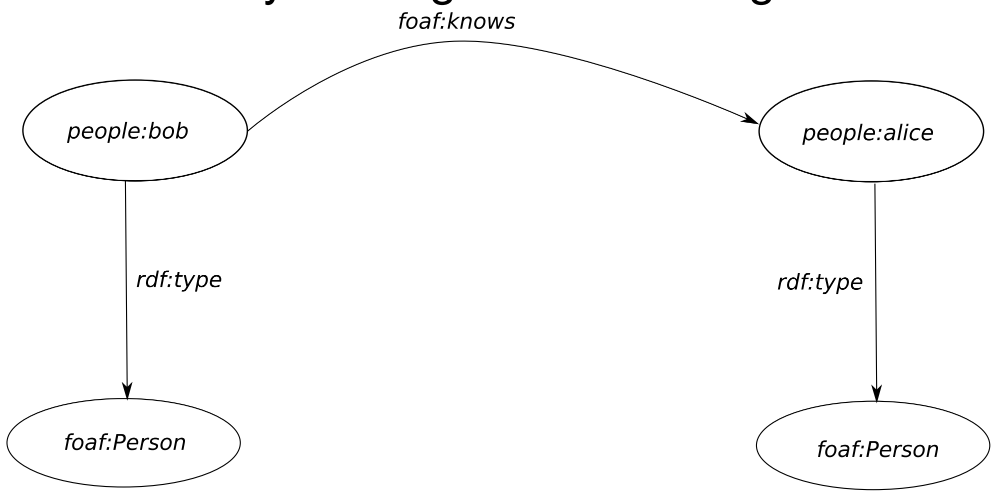
- And now...
  - Never compute twice the same thing
  - Being able to reverse-engineer every objects in our workspace

Interaction with a shared knowledge environment

- a) A semantic-web knowledge environment
- b) Interpreting semantic-web knowledge inside the ‘knowledge machines’
- c) Exporting knowledge to the semantic-web environment
- d) Implementation

A semantic-web knowledge environment

- **A *web of data* (*RDF*)** [Lassila and Swick, 1998]...
  - Considering every objects as first-class entities in a relational data structure
  - Assigning URIs to these entities, and to the properties binding them together
  - Let every actor contribute by making new knowledge available



- **... which *means* something (*OWL*)** [McGuinness and van Harmelen, 2003]
  - Understanding this data by linking it to real world objects
  - Domain ontologies: identify important concepts and relations in a given domain

Interpreting semantic-web knowledge inside the 'knowledge machines'

- **Building an *interpretation* of the theory**
  Interpreting available knowledge as predicates

- **Example: Creating a predicate linking an audiofile and an instrument**

```
PREFIX mu: <http://purl.org/NET/c4dm/music.owl>
SELECT ?a ?i WHERE {
    ?a rdf:type mu:AudioFile. ?a mu:encodesSignal ?dts.
    ?dts mu:sampledVersionOf ?cts. ?rec event:hasProduct ?cts.
    ?rec event:hasFactor ?snd. ?perf event:hasProduct ?snd.
    ?perf mu:hasFactor ?i. ?i rdf:type mu:Instrument }
```

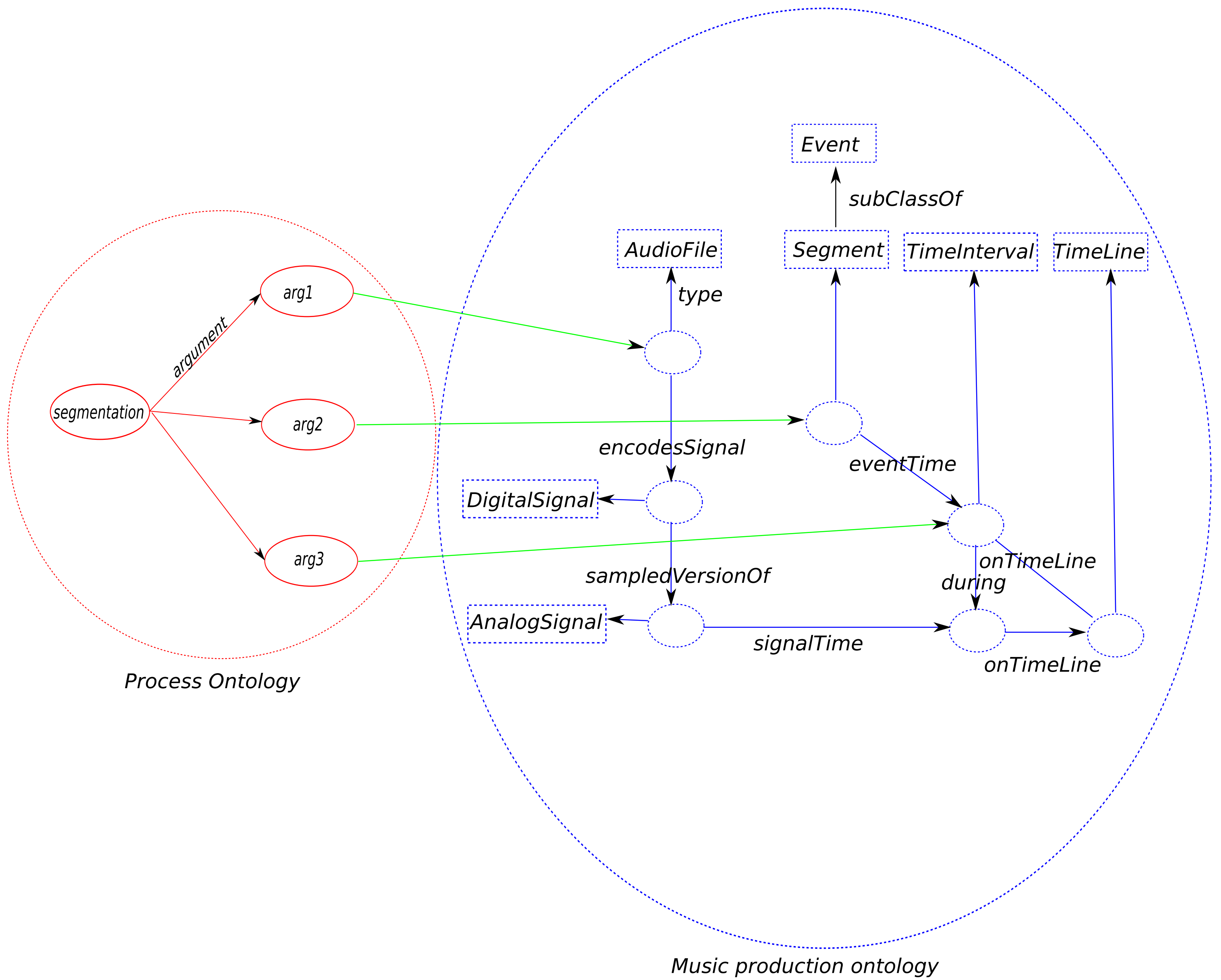
  could be associated with the following predicate:
  ***audiofile_ instrument*(AudioFile,Instrument)**
  now available when building composite formulæ

Exportingknowledge from the ‘knowledge machines’ to the semantic-web environment

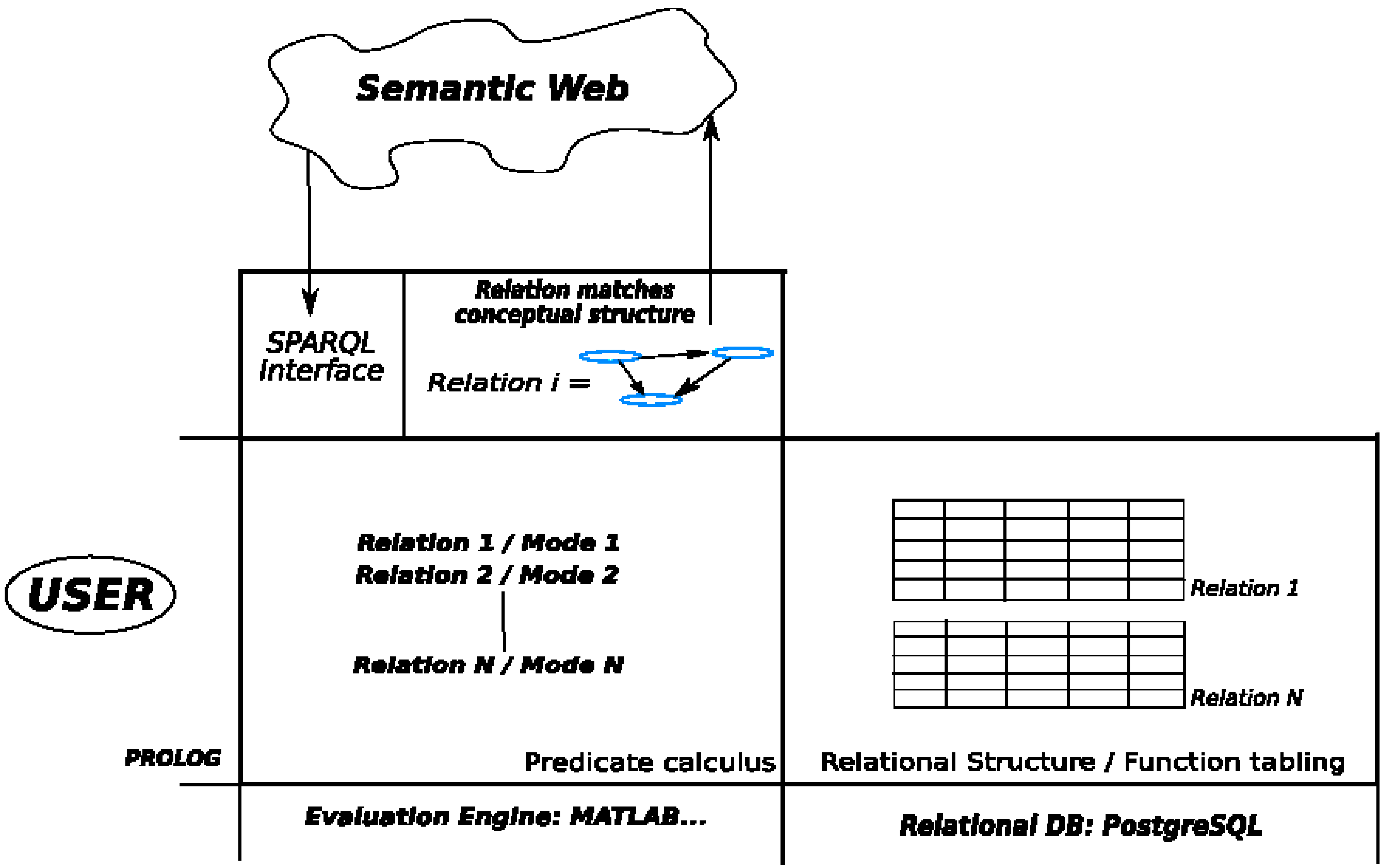
- **Once a predicate is considered relevant according to the available domain ontologies we have access to**
  - Express what the predicate *means* through what we call a *semantic match*
  - Express this match in terms of RDF/OWL

- **Exporting knowledge**
  - When this predicate holds new knowledge, export it to the SW
  - A planning component interprets *semantic matches*

Relating a segmentation predicate to its effects



Implementation



Two use cases

- a) Enhanced workspace for multimedia processing researchers
- b) End-user information access

Both sides of the stack...

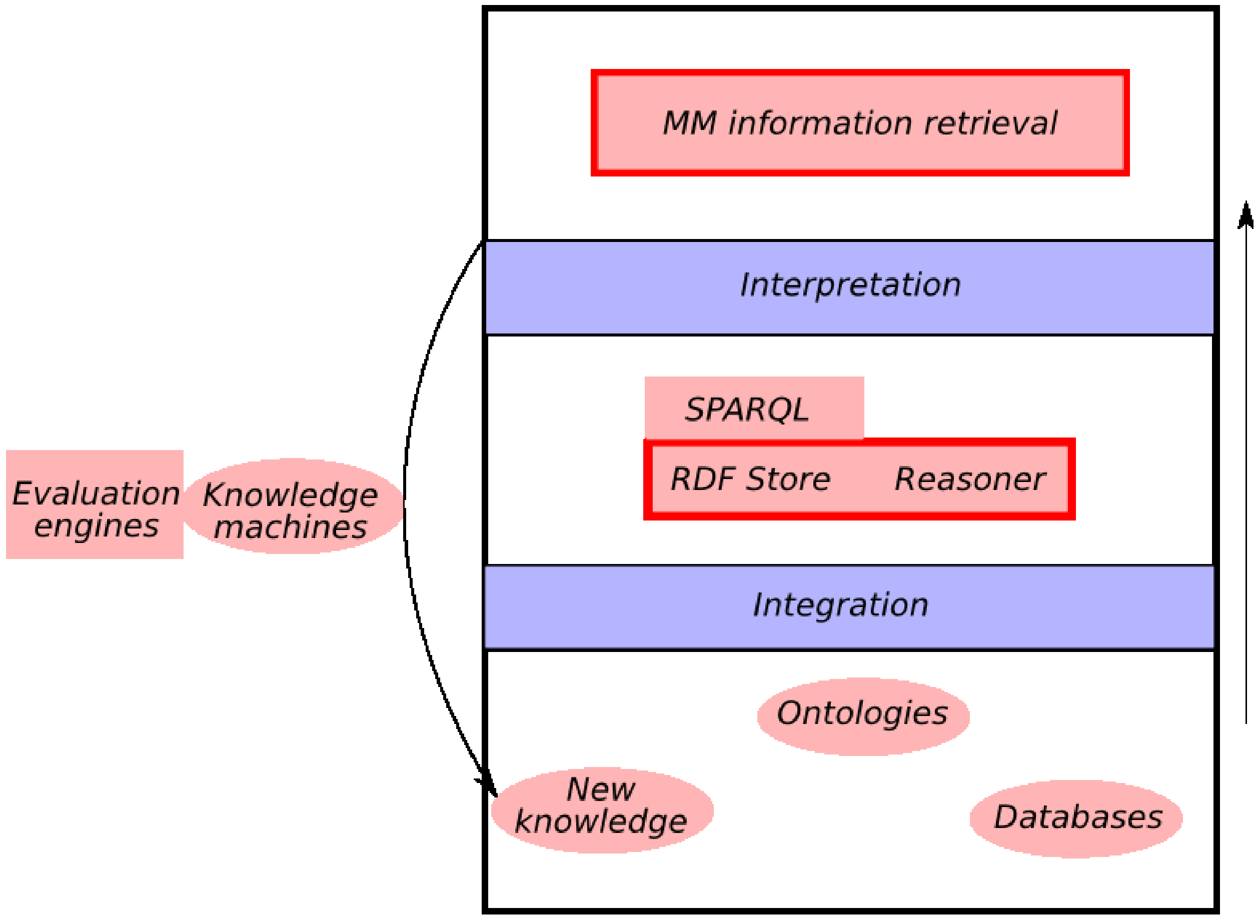
Enhanced workspace for multimedia processing researchers

- **A *semantic workspace*...**
  - Every object in the workspace is part of the same logical structure
  - Every object in the workspace can be *reverse-engineered*
  - Never do twice the same computation!
- **... aware of an open-knowledge environment**
  - Access the knowledge environment *inside* the concept detector
  - Access to an ever evolving knowledge environment
  - Export knowledge by specifying that a newly created predicate actually *do* something relevant
- **Example - Melody extraction algorithm**

  I can state that a particular sub-algorithm is to be used if the audio signal was created by a particular instrument.
  This could lead to the use of an instrument classification algorithm, exported by another knowledge machine.

Information access

On top of the shared knowledge environment, information retrieval tools can be built, by providing an *interpretation layer*.



Conclusion

- ***Knowledge Machines***
  allowing to wrap and work on multimedia analysis algorithms in a *semantic workspace*

- **... aware of an open knowledge environment**
  by being able to access this knowledge inside predicates, thus available for re-use

- **... and exporting knowledge**
  by specifying a *match* between a predicate and a RDF graph pattern

- **Adaptable to a large range of data analysis problems**

A network of Knowledge Machines could bring a distributed, approximate and artificial *cognition* for multimedia materials, against a *culture* which is defined by the different available ontologies.

Further work

- Quantifying accuracy of statements?
  A computer-generated concept detection has not the *same* accuracy as an human-generated one

- Refining the planning component?
  Better interpretation of 'semantic matches'

- Statistical analysis *inside* the knowledge machine framework
  Judging whether a predicate has successfully captured a given concept

Questions?

References

[Abdallah et al., 2006] Abdallah, S., Raimond, Y., and Sandler, M. (2006). An ontology-based approach to information management for music analysis systems. In *Proceedings of 120th AES convention*.

[Lassila and Swick, 1998] Lassila, O. and Swick, R. (1998). Resource description framework (RDF) model and syntax specification.

[McGuinness and van Harmelen, 2003] McGuinness, D. L. and van Harmelen, F. (2003). Owl web ontology language: Overview. Working draft, World Wide Web Consortium.

[Pachet, 2005] Pachet, F. (2005). *Knowledge Management and Musical Metadata*. Idea Group.

[Sagonas et al., 1994] Sagonas, K., Swift, T., and Warren, D. S. (1994). XSB as an efficient deductive database engine. In *SIGMOD '94: Proceedings of the 1994 ACM SIGMOD international conference on Management of data*, pages 442–453, New York, NY, USA. ACM Press.

[Swartz, 2002] Swartz, A. (2002). Musicbrainz: A semantic web service. *IEEE Intelligent Systems*, 17(1):76–77.