DIP
Data, Information and Process Integration with Semantic Web Services
FP6 - 507483

Deliverable

WP 10: Case study eBanking

D10.4
WSMO descriptions of Application

Silvestre Losada
Oscar Corcho
Jesús Contreras
Mónica Martínez Montes
José Luis Bas
Sergio Bellido
Richard Benjamins
Jordi Ribas

June 29th, 2005

12/07/2005
EXECUTIVE SUMMARY

This deliverable contains the WSMO descriptions of the Semantic Web Services that will be deployed in the context of the mortgage comparison application specified in deliverable D10.2.

The deliverable contributes to the goals of DIP by providing a use case for the use of WSMO in the description of the SWS involved in an application, and for the use of the SWS architecture defined in the context of this project.

This deliverable is relevant for all the DIP technical workpackages (1, 2, 3, 4a, 4b, 5, and 6), since it provides constraints about the type of SWS that will have to be represented in the context of eBanking applications. It also provides relevant information about how this case study will use the functionalities provided by the DIP infrastructure, namely discovery, mediation and execution.

The target audience of this deliverable is as follows: the partners who are developing tools to be used for the description of SWS, the partners who are developing the DIP infrastructure according to the DIP architecture (either the underlying middleware or the individual components), and external readers who are interested in finding information about a use case for SWS.

Disclaimer: The DIP Consortium is proprietary. There is no warranty for the accuracy or completeness of the information, text, graphics, links or other items contained within this material. This document represents the common view of the consortium and does not necessarily reflect the view of the individual partners.
WSMO descriptions of Application

Document Information

<table>
<thead>
<tr>
<th>IST Project Number</th>
<th>FP6 – 507483</th>
<th>Acronym</th>
<th>DIP</th>
</tr>
</thead>
</table>

| Full title         | Data, Information, and Process Integration with Semantic Web Services |
| Project URL        | [http://dip.semanticweb.org](http://dip.semanticweb.org) |
| Document URL       | Kai Tullius |

<table>
<thead>
<tr>
<th>Deliverable Number</th>
<th>10.4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Title</td>
<td>WSMO descriptions of the application</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Work package Number</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Title</td>
<td>Case study eBanking</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Date of delivery</th>
<th>Contractual</th>
<th>Actual</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>M 18</td>
<td>29-June-2005</td>
</tr>
<tr>
<td>Status</td>
<td>version 0.1</td>
<td>final</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Nature</th>
<th>Prototype □</th>
<th>Report □</th>
<th>Dissemination □</th>
<th>Ontology □</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Dissemination Level</th>
<th>Public □</th>
<th>Consortium □</th>
</tr>
</thead>
</table>

| Authors (Partner) | Silvestre Losada, Oscar Corcho, Jesús Contreras, Richard Benjamins, Jordi Ribas (iSOCO), Mónica Martínez Montes, José Luis Bas, Sergio Bellido (Bankinter), |
| Responsible Author | Silvestre Losada |
| Email              | slosada@isoco.com |
| Partner            | iSOCO |
| Phone              | +34913349797 |

Abstract (for dissemination) This deliverable contains the WSMO descriptions of the SWS that will be deployed in the context of the mortgage comparison application specified in deliverable D10.2.

| Keywords | WSMO, financial services |

Version Log

<table>
<thead>
<tr>
<th>Issue Date</th>
<th>Rev No.</th>
<th>Author</th>
<th>Change</th>
</tr>
</thead>
<tbody>
<tr>
<td>2-04-05</td>
<td>001</td>
<td>Silvestre Losada</td>
<td>Creates initial version of the Document</td>
</tr>
<tr>
<td>20-04-05</td>
<td>002</td>
<td>Oscar Corcho</td>
<td>Executive summary</td>
</tr>
<tr>
<td>2-05-05</td>
<td>003</td>
<td>Jesús</td>
<td>Changes in sections 2 &amp; 3. Conceptual architecture</td>
</tr>
</tbody>
</table>

Deliverable 10.4
<table>
<thead>
<tr>
<th>Date</th>
<th>No.</th>
<th>Author(s)</th>
<th>Changes</th>
</tr>
</thead>
<tbody>
<tr>
<td>10-05-05</td>
<td>004</td>
<td>Contreras</td>
<td>WSMO descriptions</td>
</tr>
<tr>
<td>13-05-05</td>
<td>005</td>
<td>Silvestre Losada</td>
<td>Changes in introducción</td>
</tr>
<tr>
<td>15-05-05</td>
<td>006</td>
<td>JL Bas Sergio Bellido</td>
<td>Descriptions added.</td>
</tr>
<tr>
<td>20-05-05</td>
<td>007</td>
<td>Jordi Ribas</td>
<td>Version for internal review.</td>
</tr>
<tr>
<td>16-06-05</td>
<td>007</td>
<td>Richard Benjamins</td>
<td>Implement recommendations QA review.</td>
</tr>
</tbody>
</table>
### Project Consortium Information

<table>
<thead>
<tr>
<th>Partner</th>
<th>Acronym</th>
<th>Contact</th>
</tr>
</thead>
</table>
| National University of Ireland Galway | NUIG | Prof. Dr. Christoph Bussler  
Digital Enterprise Research Institute (DERI)  
National University of Ireland, Galway  
Galway  
Ireland  
Email: chris.bussler@deri.org  
Tel: +353 91 512460 |
| Fundacion De La Innovacion.Bankinter | Bankinter | Monica Martinez Montes  
Fundacion de la Innovation. Bankinter  
Paseo Castellana, 29  
28046 Madrid,  
Spain  
Email: mmntnez@bankinter.es  
Tel: 916234238 |
| Berlecon Research GmbH | Berlecon | Dr. Thorsten Wichmann  
Berlecon Research GmbH  
Oranienburger Str. 32  
10117 Berlin,  
Germany  
Email: tw@berlecon.de  
Tel: +49 30 2852960 |
| British Telecommunications Plc. | BT | Dr John Davies  
BT Exact (Orion Floor 5 pp12)  
Adastral Park Martlesham  
Ipswich IP5 3RE,  
United Kingdom  
Email: john.nj.davies@bt.com  
Tel: +44 1473 609583 |
| Swiss Federal Institute of Technology, Lausanne | EPFL | Prof. Karl Aberer  
Distributed Information Systems Laboratory  
École Polytechnique Fédérale de Lausanne  
Bât. PSE-A  
1015 Lausanne, Switzerland  
Email: Karl.Aberer@epfl.ch  
Tel: +41 21 693 4679 |
| Essex County Council | Essex | Mary Rowatt,  
Essex County Council  
PO Box 11, County Hall, Duke Street  
Chelmsford, Essex, CM1 1LX  
United Kingdom.  
Email: maryr@essexcc.gov.uk  
Tel: +44 (0)1245 436524 |
| Forschungszentrum Informatik | FZI | Andreas Abecker  
Forschungszenrum Informatik  
Haid-und-Neu Strasse 10-14  
76131 Karlsruhe  
Germany  
Email: abecker@fzi.de  
Tel: +49 721 9654 0 |

Deliverable 10.4
<table>
<thead>
<tr>
<th>Partner</th>
<th>Acronym</th>
<th>Contact</th>
</tr>
</thead>
</table>
| Institut für Informatik, Leopold-Franzens Universität Innsbruck        | UIBK    | Prof. Dieter Fensel  
Institute of computer science  
University of Innsbruck  
Technikerstr. 25  
A-6020 Innsbruck, Austria  
Email: dieter.fensel@deri.org  
Tel: +43 512 5076485 |
| ILOG SA                                                                | ILOG    | Christian de Sainte Marie  
9 Rue de Verdun, 94253  
Gentilly, France  
Email: csma@ilog.fr  
Tel: +33 1 49082981 |
| inubit AG                                                              | Inubit  | Torsten Schmale  
inubit AG  
Lützowstraße 105-106  
D-10785 Berlin  
Germany  
Email: te@inubit.com  
Tel: +49 30726112 0 |
| Intelligent Software Components, S.A.                                  | iSOCO   | Dr. V. Richard Benjamins, Director R&D  
Intelligent Software Components, S.A.  
Pedro de Valdivia 10  
28006 Madrid, Spain  
Email: rbenjamin@isoco.com  
Tel. +34 913 349 797 |
| NIWA WEB Solutions                                                      | NIWA    | Alexander Wahler  
NIWA WEB Solutions  
Niederacher & Wahler OEG  
Kirchengasse 13/1a  
A-1070 Wien  
Email: wahler@niwa.at  
Tel: +43(0)1 3195843-11 |
| The Open University                                                     | OU      | Dr. John Domingue  
Knowledge Media Institute  
The Open University, Walton Hall  
Milton Keynes, MK7 6AA  
United Kingdom  
Email: j.b.domingue@open.ac.uk  
Tel.: +44 1908 655014 |
| SAP AG                                                                 | SAP     | Dr. Elmar Dornier  
SAP Research, CEC Karlsruhe  
SAP AG  
Vincenz-Priessnitz-Str. 1  
76131 Karlsruhe, Germany  
Email: elmar.dornier@sap.com  
Tel: +49 721 6902 31 |
WSMO descriptions of Application

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>OWL</td>
<td>Web Ontology Language</td>
</tr>
<tr>
<td>SOAP</td>
<td>Simple Object Access Protocol</td>
</tr>
<tr>
<td>SWS</td>
<td>Semantic Web Services</td>
</tr>
<tr>
<td>UDDI</td>
<td>Universal Discovery, Description and Integration</td>
</tr>
<tr>
<td>WSDL</td>
<td>Web Service Description Language</td>
</tr>
<tr>
<td>WSML</td>
<td>Web Service Modeling Language</td>
</tr>
<tr>
<td>WSMO</td>
<td>Web Service Modeling Ontology</td>
</tr>
<tr>
<td>WSMX</td>
<td>Web Service Execution Environment</td>
</tr>
<tr>
<td>XML</td>
<td>eXtensible Markup Language</td>
</tr>
</tbody>
</table>

Deliverable 10.4
TABLE OF CONTENTS

EXECUTIVE SUMMARY ...................................................................................................... I
LIST OF KEY WORDS/ABBREVIATIONS ........................................................................... VI
TABLE OF CONTENTS ..................................................................................................... VII
1. INTRODUCTION ......................................................................................................... 1
2. USE CASE OVERVIEW .............................................................................................. 1
   2.1. INTRODUCTION ............................................................................................... 1
   2.2. CONCEPTUAL ARCHITECTURE ................................................................. 3
   2.3. ACTORS ROLES AND GOALS ....................................................................... 4
   2.4. DIP COMPONENTS AND SYSTEM ARCHITECTURE ................................ 5
3. WSMO USE CASE MODELS ....................................................................................... 7
   3.1. Ontologies ........................................................................................................ 17
       3.1.1. Bankinter Domain Ontology ................................................................. 17
       3.1.2. Bankinter Knowledge Base ................................................................. 21
       3.1.3. Bank1 Domain Ontology ................................................................. 22
       3.1.4. Bank1 knowledge base ................................................................. 25
   3.2. Goals ............................................................................................................... 25
   3.3. Web Services ................................................................................................... 29
       3.3.1. Bankinter Services ................................................................. 29
       3.3.2. Bank1 services ............................................................................... 32
       3.3.3. Bank2 services ............................................................................... 35
       3.3.4. Bank3 services ............................................................................... 38
   3.4. Mediators ........................................................................................................... 41
4. CONCLUSIONS ............................................................................................................ 43
REFERENCES ..................................................................................................................... 43
APENDIX A ......................................................................................................................... 45

LIST OF FIGURES
Figure 1. Conceptual Architecture ................................................................................ 3
Figure 2 Mortgage comparison service overview. ........................................................ 4

Deliverable 10.4
LIST OF TABLES
Table 1: “Financial Ontology” ......................................................................................... 7
Table 2: “Bankinter knowledge Base” ............................................................................... 7
Table 3. “Bank1 domain Ontology” .................................................................................... 8
Table 4. “Bank1 knowledge base” ...................................................................................... 8
Table 5. “Concrete Goal for find mortgage simulator” ..................................................... 8
Table 6. “Concrete goal to find mortgage.” ....................................................................... 9
Table 7. “Concrete goal to find mortgage” ........................................................................ 9
Table 8. "Bankinter Mortgage simulator Web Service" .................................................. 10
Table 9. “Bankinter Mortgage simulator Web Service” .................................................. 10
Table 10. “Bankinter Mortgage simulator Web Service” ................................................. 11
Table 11. “Bank1 Mortgage simulator Web Service” .................................................... 11
Table 12. “Bank1 Mortgage simulator Web Service” .................................................... 11
Table 13. “Bank1 Mortgage simulator Web Service” .................................................... 12
Table 14 “Bank2 Mortgage simulator Web Service” ................................................... 12
Table 15. “Bank2 Mortgage simulator Web Service” ................................................... 12
Table 16. “Bank2 Mortgage simulator Web Service” ................................................... 13
Table 17. “Bank3 Mortgage simulator Web Service” ................................................... 13
Table 18. “Bank3 Mortgage simulator Web Service” ................................................... 14
Table 19. “Bank3 Mortgage simulator Web Service” ................................................... 14
Table 20. “OWL currency mediator” ............................................................................. 14
Table 21. “OWL currency mediator” ............................................................................. 15
Table 22. “OWL mediator, imports ontology bank2 into wsml format” ..................... 15
Table 23. “OWL mediator, imports ontology bank3 into wsml format” ..................... 15
Table 24. “OOmediator, Transforms ontology bank1 in Financial ontology” ............ 16
Table 25. “OOmediator, Transforms ontology bank2 in Financial ontology” ............ 16
Table 26. “OOmediator, Transforms ontology bank3 in financial ontology” ............ 16
1. **INTRODUCTION**

This document introduces the WSMO formal description of eBanking Case Study Web Services. The objective of this document is twofold: on the one hand it provides a useful requirement set for DIP software infrastructure and on the other hand it contributes to the construction of basic eBanking SWS infrastructure.

The eBanking case study covers almost all typical scenarios of application construction and integration using SWS technology. The application automates the process of collecting mortgage data from several banks, taking into account that the data can be accessed by executing SWS from different banks. Then it provides this aggregated information to users, according to the data that they have filled-in in appropriate query forms.

From the business point of view it helps the bank customer to find out the best mortgage product. Even if the functionality is not new (many banks already include it in their service offering) the underlying technology is revolutionary. The usage of DIP technology allows for low cost and quick application construction and adaptation to the market evolution and needs. In a short period of time and with less effort than traditional software development, the bank is able to deploy new business processes to their customers.

This document is organized as follows: section 2 defines the general organization of the Use Case; Section 3 presents examples of specific WSMO descriptions. Section 4 concludes the document.

2. **USE CASE OVERVIEW**

This section provides a description of the setting of this use case.

2.1. **INTRODUCTION**

Comparador.com is a web portal that provides mortgage information offered by financial organizations to its users. At this moment this process is not automatic and requires human interaction; since a worker has to update the information bases whenever there is a change in the mortgages, the maintenance cost is elevated. Using SWS this process can be automated, reducing the costs and increasing the quality of the information since the data are collected in real time.

Actually these kinds of services are fed with data obtained with screen-scraping techniques or filled in by humans. These data are used internally to resolve the required mathematic formulae. Thus, the information should not be quite accurate, since slightly different formulae could be used by each Bank, as is actually the case.

In contrast with these kinds of services, in our Case Study data is to be calculated by each Bank provider, using its own formulae and, therefore, certifying the results in real time. It is then possible to apply this case study to other industrial sectors. For example, it can be used to create Simulator/Comparator applications to the Travel, Real State or Insurance sectors.
The mandatory parameters to calculate the mortgage formula are:

- **Monthly payment**: The amount of money it is paid every month.
- **Number of payments** during the whole life of the mortgage.
- **Total mortgage amount**.
- **Interest rate** to be applied to the mortgage.
- **Type of interest**: Fixed type, variable type or mixed. Banks usually have different rates for each type of loan.

The specific interest rate for a given mortgage is supposed to be embedded in each bank Web Service, so it will not take part of the input parameters. The type of interest has to be provided by the user. From the other three remaining parameters (monthly payment, number of payments, total mortgage amount) only two have to be provided by the user: the other one will be calculated by the system, being the output data.

Therefore we have three combinations of possible input data:

<table>
<thead>
<tr>
<th>Input</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Monthly payment + Number of Payments + Type of Interest</td>
<td>Mortgage Amount</td>
</tr>
<tr>
<td>Monthly payment + Mortgage Amount + Type of Interest</td>
<td>Number of Payments</td>
</tr>
<tr>
<td>Number of Payments + Mortgage Amount + Type of Interest</td>
<td>Monthly payment</td>
</tr>
</tbody>
</table>

For the sake of simplicity the following assumptions are considered by each bank:

- All the monthly payments are equal (the Web Service will provide the first payment amount).
- All the monthly payments include the amortization of the principal (not only-interest payments are involved).
- The operation commissions are not considered.

Also a set of optional output data can be considered; hence they may be provided or not by some SWS.

<table>
<thead>
<tr>
<th>Optional Output data</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Starting interest rate</td>
<td>The mortgage starting interest rate</td>
</tr>
<tr>
<td>TAE</td>
<td>The interest rate including commissions (mandatory in Spain)</td>
</tr>
</tbody>
</table>
2.2. CONCEPTUAL ARCHITECTURE

In this use case, the comparator is the central point of interaction between the customer and other service providers. Our Comparator aggregates Web Services of different financial entities that provide mortgage comparator services. In short, it provides the following functionality: A customer uses the Comparator service as the entry point for his requests. The end-user services are aggregated by the Comparator by invoking Web Services offered by several financial entities service providers. Figure 1 depicts this process.

![Figure 1. Conceptual Architecture.](image)

The conceptual architecture illustrates the structure of the comparator, the services offered by the providers (banks) and the end user interfaces. Each time a client wants to know the mortgage market proposals, the application provides him/her with the actual simulations made ON-LINE in each bank WS-based Simulator. The results are given back to the user in a human-readable interface so that he is able comparing them. Some further filtering on the outputs provided to the user can be done to improve the system usability. The data obtained could be order by end user using some criteria like interest, amount etc.

Figure 2 shows the complete sequence of interactions that are carried out between the different components of the system.
WSMO descriptions of application.

When a bank creates a new SWS register description in a common repository. These descriptions are used by discovery component in order to offer to the customers suitable services according with his preferences.

1. When a customer wants to know mortgage offers he she connects to mortgage comparison service, and introduces necessary data to find mortgage offers.
2. Mortgage comparison service builds a formal goal in WSML according to the goal template using customer’s data inputs.
3. Once a goal has been built, is sent to the discovery to find suitable services that match with a customer preferences expressed in the goal.
4. Discovery component returns a set of services that fulfil customer preferences.
5. Each service is invoked by the mortgage comparison service in order to obtain all mortgage offers offered by banks.
6. Mortgages offers obtained from banks are given back in a human readable interface so that the customer is able to compare them.

2.3. ACTORS ROLES AND GOALS

In this use case there are 3 actors. This section defines why they participate in this use case (goal) and the particular interactions they are involved in (roles).

1. **Customer** the end-user that requests a service provided by the comparator
   - *Goal*: find Mortgage Offers.
WSMO descriptions of application.

- **Role:** end-user, interacts with Comparator for service usage.

2. **Comparator:** the intermediate between the Customer and the Banks. It provides mortgage market proposals to customers by aggregating the separate offers provided by the single Service Providers (Banks).
   - **Goal:** Provide to the end users the greater number of existing mortgages offers in the market, facilitating to them the selection of best mortgage, in the other side the financial organizations are able to show more easily mortgage offers to customers
   - **Role:** interaction with the customer via user interface (e.g.: web-based for direct human, via Web Services for machine-users or whatever), usage Web Services offered by Service Providers, centrally holding all functionalities for handling SWS (mechanisms for discovery, composition, execution, etc.)

3. **Banks:** commercial companies that provides mortgages offers.
   - **Goal:** Give publicity to its mortgage loans so that the final clients contract them.
   - **Role:** Provide information about his mortgages offers, also provide the semantic descriptions of its services.

### 2.4. DIP COMPONENTS AND SYSTEM ARCHITECTURE

In this use case, the comparator is the central point of interaction between the Customer and other Web Services. Regarding the technological requirements, it is obvious that the Web Services offered by the Service Providers (Banks) have to carry sufficient descriptive information to support automated Web Service usage, and the comparator has to interact with DIP components to handle Semantic Web Services. The basic architecture is shown in Figure 2. The essential functionalities are:

1) It has to provide a user interface for customer interaction: web interface, Web Service Interface, mobile devices interface, etc
2) It has to provide the appropriate Web Services for mortgage simulation.
3) It has to discover suitable Web Services for a certain user request.
4) It has to invoke external Semantic Web Services
5) It has to provide a Web Service Execution Environment with control functions, error handling, and support of optional user interaction (wsmx).
6) It has to deal properly with heterogeneous resources, thus providing the suitable mediation facilities (Banks ontologies are heterogeneous).
7) It has to register the providers’ Semantic Descriptions.
WSMO descriptions of application.

Figure 3. System architecture.

**Discovery**

In DIP, semantic discovery of web services means discovery of abstract services represented by formal service capabilities, which are part of the semantic description of the web service published in the DIP registry by the service providers (banks in this particular case). The comparator creates a goal description to describe their service requirements. To create these descriptions a goal template predefined is used by comparator to create a goal description more easily.

**Invocation**

The DIP execution environment expects the WSDL description available with the semantic web service description, and executes it.

**Mediation**

Mediation is required in the case when one component is not capable to interpret the content of a message sent by another component. The basic functionality of a data mediation module is to transform messages from source format to a target one, which could require both syntactic and semantic transformation.
Service Registry

It provides an API that facilitates the publishing of DIP elements (ontologies, mediators, goals, DIP web services etc). In figure 2 service registry is depicted as service discovery.

Semantic Repository

The semantic repository (SEMR) is an ontology server which allows the storage, retrieval, and querying of ontologies and other data (semantic web service descriptions, instance data, etc.) The banks are required to storage their ontologies in this component

3. WSMO USE CASE MODELS

This section exemplifies the specification of this use case in the “Web Service Modeling Ontology” (WSMO). These specifications comply with WSMO final version 1.1 and they have been validated with the WSMO Online Validation Service.

The following tables provide an overview of the resources specified in this use case. As no transactions nor changes in the real world occur within this use case, no assumptions and effects are needed in the service descriptions. In this case we consider currently four financial entities, Bankinter (real) Bank1, Bank2, Bank3 (simulated). However other banks can be added at any moment.

Bankinter Services

<table>
<thead>
<tr>
<th>WSMO component type</th>
<th>Ontology</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>Financial Ontology</td>
</tr>
<tr>
<td>description</td>
<td>defines ontology constructs for the bankinter domain</td>
</tr>
<tr>
<td>imported ontologies</td>
<td>OWL currency mediator.</td>
</tr>
<tr>
<td>main constructs</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>WSMO component type</th>
<th>Ontology</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>Bankinter knowledge base.</td>
</tr>
<tr>
<td>description</td>
<td>Defines ontology constructs for the bankinter domain.</td>
</tr>
</tbody>
</table>
imported ontologies |
--- | |
used mediators | OWL currency mediator. |

main constructs |
--- | |

WSML model |
--- | |
Listing 2 Bankinter KnowledgeBase |

Table 3. “Bank1 domain Ontology”

<table>
<thead>
<tr>
<th>WSMO component type</th>
<th>Ontology</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>BankOntology</td>
</tr>
<tr>
<td>Description</td>
<td>Defines ontology constructs for the bank 1 domain.</td>
</tr>
</tbody>
</table>

imported ontologies |
--- | |
used mediators | Date and Time Ontology\(^1\). |

main constructs |
--- | |
OWL currency mediator. |

WSML model |
--- | |
Listing 3. Bank1 domain ontology. |

Table 4. “Bank1 knowledge base”

<table>
<thead>
<tr>
<th>WSMO component type</th>
<th>Ontology</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Bank1 Knowledge Base</td>
</tr>
<tr>
<td>Description</td>
<td>holds all pre-defined instance data needed within the use case</td>
</tr>
</tbody>
</table>

imported ontologies |
--- | |
used mediators | - BankOntology |

main constructs |
--- | |
instances for : |
Some kinds of ontologies |

WSML model |
--- | |
Listing 4. Bank1 Knowledge Base. |

Table 5. “Concrete Goal for find mortgage simulator”

<table>
<thead>
<tr>
<th>WSMO component type</th>
<th>Goal</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>Mortgage Simulator</td>
</tr>
<tr>
<td>description</td>
<td>Concrete goal example to find out mortgage offers.</td>
</tr>
</tbody>
</table>

\(^1\) Available at http://www.wsmo.org/ontologies/dateTime/
### WSMO descriptions of application.

<table>
<thead>
<tr>
<th>imported ontologies used mediators</th>
<th>- FinancialOntology</th>
</tr>
</thead>
<tbody>
<tr>
<td>main constructs</td>
<td></td>
</tr>
<tr>
<td></td>
<td><em>precondition:</em> Mortgage simulator with know loan capital, interest rate an monthly payment, other restrictions like max value for.</td>
</tr>
<tr>
<td></td>
<td><em>postcondition:</em> Number of payments for a mortgage that has an opening commission of 0.7 and its necessary to contract life insurance for this mortgage.</td>
</tr>
</tbody>
</table>

**WSML model**

**Table 5.** “Concrete Goal for find mortgage simulator”

<table>
<thead>
<tr>
<th>WSMO component type</th>
<th>Goal</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>Mortgage simulator</td>
</tr>
<tr>
<td>description</td>
<td>A concrete goal that would be send to discovery component to find mortgage simulator services that matches with restrictions expresed in a goal</td>
</tr>
<tr>
<td>imported ontologies used mediators</td>
<td>- FinancialOntology</td>
</tr>
<tr>
<td>main constructs</td>
<td><em>precondition:</em> It’s necessary to know the end date of the payment of the mortgage, the interest rate type and the total amount of mortgage loan. <em>postcondition:</em> Monthly payment of mortgage that has opening</td>
</tr>
</tbody>
</table>

**WSML model**

**Listing 6.** Concrete goal example to find a mortgage simulator knowing the numbers of payment, the interest rate type and the total amount of the mortgage loan.

<table>
<thead>
<tr>
<th>WSMO component type</th>
<th>Goal</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>Mortgage Simulator</td>
</tr>
<tr>
<td>description</td>
<td>A concrete goal that would be send to discovery component to find mortgage simulator services that matches with restrictions expresed in a goal</td>
</tr>
<tr>
<td>imported ontologies used mediators</td>
<td>- FinancialOntology</td>
</tr>
</tbody>
</table>

**Table 6.** “Concrete goal to find mortgage.”

**Table 7.** “Concrete goal to find mortgage”
| main constructs | precondition: A monthly payment, an interest rate type and the number of payments. Mortgage must be interest rate variable along the time.  

postcondition: Total amount of the mortgage that has an opening commission of 0.25 and its subrogation commission is 0.2 |
| WSML model | Listing 7. Concrete goal exampe to find mortgage simulator knowing the monthly payment, the interest rate type and the number of payments. |

---

**Table 8. "Bankinter Mortgage simulator Web Service"**

<table>
<thead>
<tr>
<th>WSMO component type</th>
<th>Web Service</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Bankinter Mortgage simulator Web Service</td>
</tr>
<tr>
<td>description</td>
<td>Web service for simulate mortgage loans</td>
</tr>
<tr>
<td>imported ontologies used mediators</td>
<td>Financial Ontology.</td>
</tr>
</tbody>
</table>

| main constructs | precondition: It is necessary to know monthly payment, interest rate type and number of payments.  

postcondition: Total amount of mortgage loan that has opening commission 0.7 and its necessary contract life insurance for this mortgage. |

---

**Table 9. “Bankinter Mortgage simulator Web Service”**

<table>
<thead>
<tr>
<th>WSMO component type</th>
<th>Web Service</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Bankinter Mortgage simulator Web Service</td>
</tr>
<tr>
<td>description</td>
<td>Web service to simulate mortgage loans</td>
</tr>
<tr>
<td>imported ontologies used mediators</td>
<td>Financial Ontology.</td>
</tr>
</tbody>
</table>

| main constructs | precondition: It’s necessary to know the monthly payment, the interest rate type and the total amount of the mortgage loan.  

postcondition: Number of payments of the mortgage that have an opening commission of 0.7 and it’s necessary contract life insurance for this mortgage. |

- 10 -
WSMO descriptions of application.

<table>
<thead>
<tr>
<th>WSMO component type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>Web Service</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Name</th>
<th>Bankinter Mortgage simulator Web Service</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>Web service for simulate mortgage loans</td>
</tr>
<tr>
<td>Imported ontologies/used mediators</td>
<td>Financial Ontology.</td>
</tr>
</tbody>
</table>

Main constructs:

**Precondition:** It’s necessary to know the number of payments, the interest rate type and the total amount of mortgage loan.

**Postcondition:** Monthly payment of mortgage that has an opening commission of 0.7 and it's necessary contract life insurance for this mortgage.

|------------|-----------------------------------------------------|

---

Table 10. “Bankinter Mortgage simulator Web Service”

<table>
<thead>
<tr>
<th>WSMO component type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>Web Service</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Name</th>
<th>Bank1 Mortgage simulator Web Service</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>Web service for simulate mortgage loans</td>
</tr>
<tr>
<td>Imported ontologies/used mediators</td>
<td>BankOntology</td>
</tr>
</tbody>
</table>

Main constructs:

**precondition:** It’s necessary to know a monthly payment, the interest rate type and the number of payments.

**postcondition:** Total amount of mortgage loan that has an opening commission of 0.7 and a cancel commission 0.1.

|------------|-----------------------------------------------------|

---

Table 11. “Bank1 Mortgage simulator Web Service”

<table>
<thead>
<tr>
<th>WSMO component type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>Web Service</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Name</th>
<th>Bank1 Mortgage simulator Web Service</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>Web service for simulate mortgage loans</td>
</tr>
<tr>
<td>Imported ontologies/used mediators</td>
<td>BankOntology</td>
</tr>
</tbody>
</table>

Main constructs:

**precondition:** It’s necessary to know a monthly payment, the interest rate type and the number of payments.

**postcondition:** Total amount of mortgage loan that has an opening commission of 0.7 and a cancel commission 0.1.

|------------|--------------------------------------------------|
WSMO descriptions of application.

imported ontologies used mediators | BankOntology.
---|---

**main constructs**

| precondition: | It’s necessary to know a monthly payment, interest rate type and Total amount of mortgage loan.
| postcondition: | End date of payment for mortgage of mortgage that has opening commission 0.7 and cancel commission 0.1 |

**WSML model** | Listing 12 **Bank1 mortgage simulator Web Service.**

---

**Table 13. “Bank1 Mortgage simulator Web Service”**

<table>
<thead>
<tr>
<th>WSMO component type</th>
<th>Web Service.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Bank 1 Mortgage simulator Web Service</td>
</tr>
<tr>
<td>Description</td>
<td>Web service for simulate mortgage loans</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>imported ontologies used mediators</th>
<th>BankOntology.</th>
</tr>
</thead>
</table>

**main constructs**

| precondition: | It’s necessary to know the end date of the payment of the mortgage, the interest rate type and the total amount of the mortgage loan. 
| postcondition: | Monthly payment of the mortgage that has an opening commission of 0.7 and a cancel commission of 0.1. |

**WSML model** | Listing 13 **Bank1 mortgage simulator Web Service.**

---

**Bank 2 Services**

**Table 14 “Bank2 Mortgage simulator Web Service”**

<table>
<thead>
<tr>
<th>WSMO component type</th>
<th>Web Service</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Mortgage Simulator</td>
</tr>
<tr>
<td>Description</td>
<td>Mortgage simulator Web Service.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>imported ontologies used mediators</th>
<th>OWL 2 WSML mediator.</th>
</tr>
</thead>
</table>

**main constructs**

| precondition: | Mortgage simulator with known monthly payments, interest rate and number of payments. 
| postcondition: | Mortgage capital of the mortgage that has an opening commission of 0.7, an interest delay commission of 0.1 and its necessary to contract a life insurance for this mortgage |

**WSML model** | Listing 14. **Bank2 mortgage simulator Web Service**

---

**Table 15. “Bank2 Mortgage simulator Web Service”**

---
**Table 15. Bank2 mortgage simulator Web Service.**

<table>
<thead>
<tr>
<th>WSMO component type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Mortgage simulator Web Service</td>
</tr>
<tr>
<td>Description</td>
<td>Web service for simulate mortgage loans</td>
</tr>
<tr>
<td>imported ontologies</td>
<td>- OWL 2 WSML mediator.</td>
</tr>
<tr>
<td>used mediators</td>
<td></td>
</tr>
<tr>
<td>main constructs</td>
<td><strong>precondition:</strong> It’s necessary to know the monthly payment, the</td>
</tr>
<tr>
<td></td>
<td>interest rate type and the total amount of the mortgage loan.</td>
</tr>
<tr>
<td></td>
<td><strong>postcondition:</strong> Number of payments of the mortgage that has an</td>
</tr>
<tr>
<td></td>
<td>opening commission of 0.7, an interest delay commission of 0.1 and it’s</td>
</tr>
<tr>
<td></td>
<td>necessary contract a life insurance for this mortgage.</td>
</tr>
<tr>
<td>WSML model</td>
<td>Listing 15. <strong>Bank2 mortgage simulator Web Service.</strong></td>
</tr>
</tbody>
</table>

**Bank 3 Services**

**Table 16. “Bank3 Mortgage simulator Web Service”**

<table>
<thead>
<tr>
<th>WSMO component type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Mortgage simulator Web Service</td>
</tr>
<tr>
<td>Description</td>
<td>Web service for simulate mortgage loans</td>
</tr>
<tr>
<td>imported ontologies</td>
<td>- OWL 2 WSML mediator.</td>
</tr>
<tr>
<td>used mediators</td>
<td></td>
</tr>
<tr>
<td>main constructs</td>
<td><strong>precondition:</strong> A monthly payment, interest rate type and</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>WSML model</td>
<td>Listing 16. <strong>Bank2 mortgage simulator Web Service.</strong></td>
</tr>
</tbody>
</table>
WSMO descriptions of application.

number of payments. Mortgage must be interest rate variable along the time.

postcondition: Total amount of mortgage which opening commission is 0.25 and subrogation commission is 0.2.

WSML model Listing 17. Bank3 mortgage simulator Web Service.

Table 18. “Bank3 Mortgage simulator Web Service”

<table>
<thead>
<tr>
<th>WSMO component type</th>
<th>Web Service</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Mortgage simulator Web Service</td>
</tr>
<tr>
<td>Description</td>
<td>Web service for simulate mortgage loans</td>
</tr>
<tr>
<td>imported ontologies used mediators</td>
<td>- OWL 2 WSML mediator.</td>
</tr>
<tr>
<td>main constructs</td>
<td>precondition: It’s necessary know the monthly payment, the interest rate type and the total amount of the mortgage loan. Mortgage must be interest rate variable along the time. postcondition: Number of payments of the mortgage which opening commission is 0.25 and subrogation commission is 02.</td>
</tr>
</tbody>
</table>

Table 19. “Bank3 Mortgage simulator Web Service”

<table>
<thead>
<tr>
<th>WSMO component type</th>
<th>Web Service</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Mortgage simulator Web Service</td>
</tr>
<tr>
<td>Description</td>
<td>Web service for simulate mortgage loans</td>
</tr>
<tr>
<td>imported ontologies used mediators</td>
<td>- OWL 2 WSML mediator.</td>
</tr>
<tr>
<td>main constructs</td>
<td>precondition: A number of payments, interest rate type and total amount of mortgage loan. Mortgage must be interest rate variable along the time. postcondition: Monthly payment of a mortgage which opening commission is 0.25 and subrogation commission is 0.2.</td>
</tr>
</tbody>
</table>

Table 20. “OWL currency mediator”

<table>
<thead>
<tr>
<th>WSMO component type</th>
<th>ooMediator</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>OwlCurrencyMediator</td>
</tr>
<tr>
<td>description</td>
<td>importing the Currency DAML ontology to WSML</td>
</tr>
</tbody>
</table>
### Table 20. OWL currency mediator Bankinter.

<table>
<thead>
<tr>
<th>WSMO component type</th>
<th>ooMediator</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>OwlCurrencyMediator</td>
</tr>
<tr>
<td>Description</td>
<td>Mediator to import the Currency DAML ontology to WSML</td>
</tr>
</tbody>
</table>

### Table 21. OWL currency mediator.

<table>
<thead>
<tr>
<th>WSMO component type</th>
<th>ooMediator</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>OwlCurrencyMediator</td>
</tr>
<tr>
<td>Description</td>
<td>Mediator to import the Currency DAML ontology to WSML</td>
</tr>
</tbody>
</table>

### Table 22. OWL mediator, imports ontology bank2 into wsml format.

<table>
<thead>
<tr>
<th>WSMO component type</th>
<th>ooMediator</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>OWL2WSMLOntologyBank2</td>
</tr>
<tr>
<td>Description</td>
<td>Mediator to import the OWL OntologyBank2 ontology to WSML</td>
</tr>
</tbody>
</table>

### Table 23. OWL mediator, imports ontology bank3 into wsml format.

<table>
<thead>
<tr>
<th>WSMO component type</th>
<th>ooMediator</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>importing the OWL OntologyBank3 ontology to WSML</td>
</tr>
<tr>
<td>description</td>
<td>Defines ontology constructs for the bankinter domain.</td>
</tr>
</tbody>
</table>

WSMO descriptions of application.
WSMO descriptions of application.

<table>
<thead>
<tr>
<th>used mediators</th>
</tr>
</thead>
</table>
| main constructs | *source component*: a OntologyBank3 ontology (in OWL)  
| | *target component*: OntologyBank3 (in WSML)  
| | *mediation service*: not specified |
| WSML model | Listing 23. **OWL2WSML mediator** |

Table 24. “OOmediator, Transforms ontology bank1 in Financial ontology”.

<table>
<thead>
<tr>
<th>WSMO component type</th>
<th>ooMediator</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>Bank1toFinancialOntology</td>
</tr>
<tr>
<td>description</td>
<td>Transforms OntologyBank1 in wsml in Financial ontology</td>
</tr>
<tr>
<td>imported ontologies</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>
| main constructs | *source component*: a OntologyBank3 ontology (in OWL)  
| | *target component*: OntologyBank3 (in WSML)  
| | *mediation service*: not specified |
| WSML model | Table 24. **“OOmediator, Transforms ontology bank1 in Financial ontology”**. |

Table 25. “OOmediator, Transforms ontology bank2 in Financial ontology”.

<table>
<thead>
<tr>
<th>WSMO component type</th>
<th>ooMediator</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>Bank2toFinancialOntology</td>
</tr>
<tr>
<td>description</td>
<td>Transforms OntologyBank2 in wsml in Financial ontology</td>
</tr>
<tr>
<td>imported ontologies</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>
| main constructs | *source component*: a OntologyBank3 ontology (in OWL)  
| | *target component*: OntologyBank3 (in WSML)  
| | *mediation service*: not specified |
| WSML model | Table 25. **“OOmediator, Transforms ontology bank2 in Financial ontology”**. |

Table 26. “OOmediator, Transforms ontology bank3 in financial ontology”.

<table>
<thead>
<tr>
<th>WSMO component type</th>
<th>ooMediator</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>Bank3toFinancialOntology</td>
</tr>
<tr>
<td>description</td>
<td>Transforms OntologyBank3 in wsml in Financial ontology</td>
</tr>
</tbody>
</table>
Table 26. “OOmediator, Transforms ontology bank3 in financial ontology”.

3.1. Ontologies

With regard to modularized ontologies as a basic design principle of WSMO, we define domain ontologies as the terminology definitions.

In the banking domain standardisation efforts are very slow. Several examples exist of unsuccessful attempts, for instance, Mobipay in Spain, a micro-payments standardisation initiative, where different interests were involved and no effective market action was finally made. If standardisation efforts in bank sector are very slow each bank will use a different domain ontology[1] For this reason in the scope of Use Case each bank has different domain ontologies.

Bankinter domain ontology has been translated into WSML [4] format. Ontology Bank1 has been described using WSML format. Bank2 and Bank3 uses OWL format as formalism, these ontologies can be found in the Appendix A.

3.1.1. Bankinter Domain Ontology

The financial ontology is used to describe the financial domain in Bankinter. The ontology includes concepts like services and products; channels, users, currencies etc. All of them are described in D 10.3 [1] and translated into WSML format. This ontology does not aim at covering the whole financial domain but focuses mainly on modelling conceptually the mortgage domain for this Use Case. To model this ontology a set of 12 mortgage simulators have been used as a knowledge source. These simulators are mainly offered by financial institutions, although other non-financial Web sites are also available with the same kind of information.

The ontology has been designed to be modular enough to allow refinements in the context of the current domain and extensions to other domains in the financial area.


```xml
namespace _"http://users.isoco.net/~slosada/ontologies/bankinter/FinancialOntology.wsml#",
dc _"http://purl.org/dc/elements/11#",
foaf _"http://xmlns.com/foaf/01#",
xsd _"http://www.w3c.org/2001/XMLSchema#",
wsml _"http://www.wsmo.org/2004/wsml#",
dt _"http://www.wsmo.org/ontologies/dateTime/#",
oo _"http://www.wsmo.org/2004/d3/d3.2/v0.1/20040628/resources/owlCurrencyMediator.wsml#"

ontology _"http://users.isoco.net/~slosada/ontologies/bankinter/FinancialOntology.wsml#"

nfp
```
WSMO descriptions of application.

concept Service
   nfp
dc#description hasValue "Service is the type of things that can be done but that do not require contract (example: a transference order). Service also is any product non financier who tributes a bank"
endnfp

concept AddedValue subConceptOf Service
   nfp
dc#description hasValue "The sales revenue from selling a product less the cost of the materials purchased used in those products. It is an indicator of relative efficiency within between firms, although in the latter case it is open to distortion where mark-up varies between standard premium-priced segments of a market"
endnfp

concept Quota
   nfp
dc#description hasValue "The sales revenue from selling a product less the cost of the materials purchased used in those products. It is an indicator of relative efficiency within between firms, although in the latter case it is open to distortion where mark-up varies between standard premium-priced segments of a market"
endnfp

concept Customer
   nfp
dc#description hasValue "Bank client, who usually has a contractual relationship with the bank"
endnfp

concept Corporative subConceptOf Department
   nfp
dc#description hasValue "Company with a specific set of characteristics that require a personalised commercial treatment. Depending on each bank, the set of characteristics may change, although they normally refer to number of employees annual turnover. It includes company groups multinational companies, etc."
endnfp

concept ProductRateApplicationFixed subConceptOf ProductRateApplication
   nfp
dc#description hasValue "Fixed interest rate. It never varies during the mortgage life"
endnfp

concept SavingAccount
   nfp
dc#description hasValue "Account without a chequebook normally with a low interest rate"
endnfp

concept Collection subConceptOf Service
   nfp
dc#description hasValue "Deposit in a saving account"
endnfp

concept InvestmentFund subConceptOf Asset
   nfp
dc#description hasValue "Investment club where a set of customers put their money so that the bank performs an investment on behalf of them."
endnfp

concept Individuals subConceptOf Department
   nfp
dc#description hasValue "Department that deals with physical persons"
endnfp

concept ProductRateApplicationMixed
   nfp
dc#description hasValue "Mixed interest rate, usually composed of fixed variable interest rates"
endnfp
rateVariable ofType ProductRateApplicationVariable
termRateFixed ofType xsd:string
concept InvoicePayment subConceptOf Payment
  nfp
dc#description hasValue "An itemized statement given to a buyer by a seller usually specifying the price of goods services the terms of sale"
endnfp
concept Loan subConceptOf Asset
  nfp
dc#description hasValue "Money let out at interest"
endnfp
concept SME subConceptOf Department
  nfp
dc#description hasValue "Small Medium Enterprise"
endnfp
concept Administrative subConceptOf EmployeeArea
  nfp
dc#description hasValue "Employee with administrative functions"
endnfp
concept Payment subConceptOf Service
  nfp
dc#description hasValue "Money given to pay for something"
endnfp
concept EmployeeArea subConceptOf User
  nfp
dc#description hasValue ""
endnfp
concept User
  nfp
dc#description hasValue ""
endnfp
login ofType xsd:string
password ofType xsd:string
concept ProductRateApplicationVariable subConceptOf ProductRateApplication
  nfp
dc#description hasValue "Variable interest rate. It may vary during the mortgage life"
endnfp
concept MortgageLoan subConceptOf Loan
  nfp
dc#description hasValue "A long-term loan backed by real estate valuable property, usually the item purchased with the loan. The creditor can claim that property if all payments are not made by the borrower when they are due"
endnfp
loanCapital ofType xsd:float
revisionTermNext ofType xsd:string
concept Person
  nfp
dc#description hasValue "Bank client that represents a single person (physical juridical)"
endnfp
CustomerNIF ofType string
concept Commercial subConceptOf EmployeeArea
  nfp
dc#description hasValue "Employee with commercial functions"
endnfp
concept Liability subConceptOf Product
  nfp
dc#description hasValue "The amount that is owed by an individual company, whether money, products, services, to others."
endnfp
concept Branch subConceptOf Channel
  nfp
dc#description hasValue "Communication means used in the relationship between the bank its customers, including branches, phone, Internet, virtual banking, etc."
endnfp
concept vBanking subConceptOf Channel
  nfp
dc#description hasValue "Virtual Banking. Banking without human intervention"
endnfp
concept InvestmentAccount subConceptOf Asset
  nfp
dc#description hasValue "Account setup to perform an investment, such as a fixed term deposit"
concept Staff subConceptOf EmployeeArea
  nfp
dc#description hasValue "Central Services of a company, such as Human Resources, Management, Innovation, etc."
endnfp
concept Company subConceptOf Customer
  nfp
dc#description hasValue "A number of people grouped together as a business enterprise. Types of companies include public limited companies, partnerships, joint ventures, proprietorships, branches of foreign companies"
endnfp
CIF ofType (1) xsd#string
concept CurrentMortgageLoan subConceptOf MortgageLoan
  nfp
dc#description hasValue "Current mortgage loan that the user is willing to change."
  cancelationCommission ofType (1) Quota
  saleCostProperty ofType (1) xsd#string
  interestNextRevision ofType (1) xsd#float
  handlingCapital ofType (1) xsd#float
endnfp
concept SOHO 
  nfp
dc#description hasValue "Small Office, Home Office. It usually refers to professionals who work in their own offices"
endnfp
CIF ofType (1) string
concept Asset subConceptOf Product
  nfp
dc#description hasValue "The land property of a company individual, payments due from bills, investments, anything else owned that can be turned into cash"
  channel ofType (1) Channel
  service ofType (1) Service
endnfp
concept Department subConceptOf User
  nfp
dc#description hasValue "Internal area of the bank that performs a specific function"
endnfp
concept FutureMortgageLoan subConceptOf MortgageLoan
  nfp
dc#description hasValue "Future mortgage loan that the user will use instead of the current one"
  periodicityCuota ofType (1) dt#date
  initialPeriod ofType (1) dt#date
  term ofType (1) Term
  subrogationCommission ofType (1) xsd#float
  revisionTerm ofType (1) xsd#duration
  buyCostProperty ofType (1) xsd#double
  openingCommission ofType (1) xsd#float
  quotasAfterRevision ofType (1) xsd#float
  initialQuota ofType (1) xsd#float
  delayInterestRate ofType (1) xsd#float
  interesDelay ofType (1) xsd#float
  lifeInsurance ofType (1) xsd#boolean
  homeInsurance ofType (1) xsd#boolean
  mortgageTaxation ofType (1) xsd#float
endnfp
concept Term
  totalTerm ofType (1) xsd#integer,
  typeTerm hasValue (1) xsd#string
endnfp
concept Product
  nfp
dc#description hasValue "Bank product that requires the signature of a contract between the customer and the bank"
  interestRateType ofType (1) ProductRateApplication
  expirationDate ofType (1) dt#date
  signalDateContract ofType (1) dt#date
  APR ofType (1) xsd#float
  currencyProduct ofType (1) cu#currency
endnfp
3.1.2. Bankinter Knowledge Base.

Bankinter Knowledge Base holds all instance data that are needed within the use case descriptions. The knowledge base is defined as WSMO ontology that holds instances.

Listing 2 Bankinter KnowledgeBase

```xml
namespace _"http://users.isoco.net/~slosada/ontologies/bankinter/FinancialOntology.wsml#",
  dc _"http://purl.org/dc/elements/11#",
  foaf _"http://xmlns.com/foaf/01/",
  xsd _"http://www.w3c.org/2001/XMLSchema#",
  wsml _"http://www.wsmo.org/2004/wsml#",
  loc _"http://www.wsmo.org/ontologies/dateTime#",
  cu _"http://www.wsmo.org/2004/d3/d3.2/v0.1/20040628/resources/owlCurrencyMediator.wsml#",
  fin _"http://users.isoco.net/~slosada/ontologies/bankinter/FinancialOntology.wsml#"

ontology _"http://users.isoco.net/~slosada/ontologies/bankinter/KBBankinter.wsml"

nfp
dc#title hasValue "Bankinter knowledge base"
dc#subject hasValue "Financial" dc#description hasValue "Describes the pre-defined instances for Bankinter Mortgage Simulator"
dc#subject hasValue { "Simulator", "Mortgage", "Financial", "Product"} dc#publisher hasValue (ISOCO, BANKINTER) dc#date hasValue _date("2005-03-14") dc#format hasValue "text/html" dc#language hasValue "en-US" dc#rights hasValue _http://www.isoco.com/privacy.html" wsml#version hasValue "$Revision: 1.0 $"
endnfp

usesMediator ( _"http://www.wsmo.org/2004/d3/d3.2/v0.1/20040628/resources/owlCurrencyMediator.wsml#", 
  _"http://users.isoco.net/~slosada/ontologies/bankinter/FinancialOntology.wsml#")

instance mortgageRateVariable memberOf fin#MortgageLoan

  periodicityQuota hasValue periodicityQuotaTime
  initialPeriod hasValue periodTime
term hasValue termInstance
subrogationCommission hasValue 0.3
revisionTerm hasValue P1Y
buyCostProperty hasValue _#
openingCommission hasValue 1.0
initialQuota hasValue _#
delayInInterestRate hasValue 1.0
lifeInsurance hasValue true
homeInsurance hasValue false
mortgageTaxation hasValue _#
interestRateType hasValue rateVariable
expirationDate hasValue _#
signalDateContract hasValue _#
APR hasValue _#
currencyProduct hasValue euro
interestRateValue hasValue 3.8
name hasValue "HIPOTECA INTERES VARIABLE BANKINTER"

instance rateVariable memberOf fin#ProductRateApplicationVariable
interestRateValue hasValue 0.7
interestReference hasValue EURIBOR

instance euro memberOf cu#currency
```
3.1.3. Bank1 Domain Ontology.

This ontology models the domain of financial entity that has been simulated in the scope of the Use Case

Listing 3. Bank1 domain ontology.

```xml

ontology _"http://users.isoco.net/~slosada/ontologies/bank1/Bank1Ontology.wsml#"

nfp
dc#title hasValue "Financial Ontology"
dc#subject hasValue "Financial"
dc#description hasValue ""
dc#subject hasValue ( "Simulator", "Mortgage", "Financial", "Product")
dc#format hasValue "text/html"
dc#language hasValue "en-US"
dc#rights hasValue _"http://www.isoco.com/privacy.html"
wsml#version hasValue "$Revision: 1.0 $"
endnfp

usesMediator { _"http://www.wsmo.org/2004/d3/d3.2/v0.1/20040628/resources/owlCurrencyMediator.wsml" }

concept Details

nfp
dc#description hasValue "Detalles de anotaciones en cuentas de ahorro, fondos de inversión tarjetas de crédito etc"
endnfp
currency ofType (1) cu#currency
amount ofType (1) xsd#float

concept InvestmentAccountsValues subConceptOf Details

nfp
dc#description hasValue "Detalles de en cuentas de inversiónes "
endnfp

concept PhoneConsumeDetails subConceptOf Details

nfp
dc#description hasValue "Detalles de consumo telefónico "
endnfp
date ofType (1) dt#date
destiny ofType (1) xsd#string
destinyPhone ofType (1) xsd#string
destinyPhone ofType (1) xsd#string
duration ofType (1) xsd#integer
phone ofType (1) xsd#string

concept SavingAccountsDetails subConceptOf Details

nfp
dc#description hasValue "Detalles de movimientos cuentas corrientes, movimientos de debe, movimientos haber "
endnfp
code ofType (1) xsd#string
date ofType (1) xsd#date
```
WSMO descriptions of application.

```xml
<concept InvoiceDetails subConceptOf Details>
  <nfp>
    <dc#description hasValue>"Detalle de pago de facturas"
  </nfp>
</concept>

<concept Transactions>
  <nfp>
    <dc#description hasValue>"Especially a business agreement exchange"
  </nfp>
</concept>

<concept AccountTransactions subConceptOf Transactions>
  <nfp>
    <dc#description hasValue>"Transacciones asociadas a una cuenta"
  </nfp>
</concept>

<concept CreditCardTransactions subConceptOf Transactions>
  <nfp>
    <dc#description hasValue>"Transacciones asociadas a una tarjeta de credito"
  </nfp>
</concept>

<concept Product>
  <nfp>
    <dc#description hasValue>"The land property of a company individual, payments due from bills, investments, anything else owned that can be turned into cash"
  </nfp>
  <holder ofType (1) User>
  <interest_rate ofType (1) xsd#float>
  <product_id ofType (1) xsd#string>
  <currency ofType (1) cu#currency>
</concept>

<concept InvestmentAccounts subConceptOf Product>
  <nfp>
    <dc#description hasValue>"Account setup to perform an investment, such as a fixed term deposit"
  </nfp>
  <date ofType (1) dt#date>
  <number ofType (1) xsd#string>
  <description ofType (1) xsd#string>
</concept>

<concept InvestmentFounds subConceptOf Product>
  <nfp>
    <dc#description hasValue>"Investment club where a set of customers put their money so that the bank performs an investment on behalf of them."
  </nfp>
  <buyValue ofType (1) xsd#float>
  <date ofType (1) dt#date>
  <name ofType (1) xsd#string>
  <number ofType (1) xsd#string>
  <unitPrice ofType (1) xsd#float>
  <units ofType (1) xsd#integer>
  <yield ofType (1) xsd#float>
</concept>

<concept Loan subConceptOf Product>
  <nfp>
    <dc#description hasValue>"Money let out at interest"
  </nfp>
  <l_currency ofType (1) cu#currency>
  <l_endDate ofType (1) dt#date>
  <l_startDate ofType (1) dt#date>
  <l_PayAmountTerm ofType (1) xsd#float>
  <l_startBalance ofType (1) xsd#float>
</concept>

<concept MortgageLoan subConceptOf Loan>
  <nfp>
    <dc#description hasValue>"A long-term loan backed by real estate valuable property, usually the item purchased with the loan. The creditor can claim that property if all payments are not made by the borrower when they are due"
  </nfp>
  <ml_capital ofType (1) xsd#float>
</concept>
```
WSMO descriptions of application.

```xml
ml_interestRateType ofType (1) xsd#string
ml_openingComission ofType (1) xsd#float
ml_subrogationComission ofType (1) xsd#float
ml_totalCancelComission ofType (1) xsd#float

concept SavingAccounts subConceptOf Product
  nfp
dc#description hasValue "Account without a chequebook normally with a low interest rate"
endnfp
amount ofType (1) xsd#float
currency ofType (1) cu#currency
transactions ofType (0 *) AccountTransactions
accountNumber ofType (0 *) xsd#string
associateInvoices ofType (0 *) InvoicesPayment

concept Services
  nfp
dc#description hasValue "Service is the type of things that can be done but that do not require contract (example: a transference order). Service also is any product nonfinancier who distributes a bank"
endnfp
holder ofType (1) User
service_id ofType (1) xsd#string

concept CreditCard subConceptOf Services
  nfp
dc#description hasValue "A plastic card having a magnetic strip, issued by a bank business authorizing the holder to buy goods services on credit. Also called charge card."
endnfp
ledgerBalance ofType (1) xsd#float
nextCloseDate ofType (1) dt#date
number ofType (1) xsd#string
transactions ofType (0 *) CreditCardTransactions
nextPayDate ofType (1) dt#date

concept InvoicesPayment subConceptOf Services
  nfp
dc#description hasValue "An itemized statement given to a buyer by a seller usually specifying the price of goods services the terms of sale"
endnfp
associatedAccount ofType (0 *) SavingAccount
details ofType (0 *) InvoicesDetails
invoiceAumount ofType (1) xsd#float
invoiceContractNumber ofType (1) xsd#string
currency ofType (1) cu#currency
invoiceNumber ofType (1) xsd#string
invoiceType ofType (1) xsd#string

concept PhoneConsume subConceptOf Services
  nfp
dc#description hasValue "Service that allows to consult the phone consume"
endnfp
currency ofType (1) cu#currency
amount ofType (1) xsd#string
details ofType (1) PhoneConsumeDetails
phone ofType (1) xsd#string
startDate ofType (1) dt#date
endDate ofType (1) dt#date

concept User
  nfp
dc#description hasValue "Entity that have been contracted some type of product service in the bank"
endnfp
login ofType (1) xsd#string
NIF ofType (1) xsd#string
password ofType (1) xsd#string
products ofType (0 *) Product
services ofType (0 *) Services
```
3.1.4. Bank1 knowledge base.

Bank1 Knowledge Base holds all the instance data needed within the use case descriptions. The knowledge base is defined as a WSMO ontology that holds instances.

Listing 4. Bank1 Knowledge Base.

```xml
namespace _"http://users.isoco.net/~slosada/ontologies/bank1/KBBank1.wsml#",
dc _"http://purl.org/dc/elements/11#",
foaf _"http://xmlns.com/foaf/01#",
xsd _"http://www.w3c.org/2001/XMLSchema#",
wsml _"http://www.wsmo.org/2004/wsml#",
cu _"http://www.wsmo.org/2004/d3/d3.2/v0.1/20040628/resources/owlCurrencyMediator.wsml#",
fin _"http://users.isoco.net/~slosada/ontologies/Bank1/Bank1Ontology.wsml#"

ontology _"http://users.isoco.net/~slosada/ontologies/bank1/KBBank1.wsml"

nfp
dc#title hasValue "Bankinter knowledge base"
dc#subject hasValue "Financial"
dc#description hasValue "Describes the pre-defined instances for Bankinter Mortgage Simulator"
dc#subject hasValue {"Simulator", "Mortgage", "Financial", "Product"}
dc#publisher hasValue (SOCCO)
dc#date hasValue _date("2005-03-18")
dc#format hasValue "text/html"
dc#language hasValue "en-US"
dc#rights hasValue _http://www.isoco.com/privacy.html"
wsml#version hasValue "$Revision: 1.0 $"
endnfp

importsOntology { _"http://users.isoco.net/~slosada/ontologies/bank1/Bank1Ontology.wsml#"}

instance mortgobeB1 memberOf fin#MortgageLoan
interest_rate hasValue 0.4
product_id hasValue hipotecaVariable
currency hasValue euro
ml_capital hasValue 190000
ml_interestRateType hasValue 0.48
ml_openningComission hasValue 0.1
ml_subrogationComission hasValue 0.2
ml_totalCancelComission hasValue 0
l_endDate hasValue _#
l_startDate hasValue _#
l_PayAmountTerm hasValue 900

instance euro memberOf cu#currency
  name hasValue "Euro"
  code hasValue "EUR"
```

3.2. Goals.

Goals denote what a user wants as the result of the Web Service. By modeling the goal, WSMO describes the information elements the user wants to get from the service (preconditions, posconditions) and the state of the world (assumptions, effects). In case of the mortgage simulators there are no changes in the real world, for this reason goals do not have assumptions and effects.

Listing 5 Goal to find mortgage simulator with know loan capital, interest rate an monthly payment.

**precondition**: Mortgage simulator with know loan capital, interest rate an monthly payment, other restrictions like max value for.
**postcondition**: Number of payments for a mortgage that has an opening commission of 0.7 and it's necessary to contract life insurance for this mortgage.

```
namespace (_"http://users.isoco.net/~slosada/ontologies/bankinter/GoalGetNumberOfPayments.wsml#",
  dc "http://purl.org/dc/elements/11#",
  foaf "http://xmlns.com/foaf/01",
  xsd "http://www.w3c.org/2001/XMLSchema",
  wsml "http://www.wsmo.org/2004/wsml#",
  fin "http://users.isoco.net/~slosada/ontologies/bankinter/FinancialOntology.wsml#")

goal _"http://users.isoco.net/~slosada/ontologies/bankinter/GoalGetNumberOfPayments.wsml"

  nfp
dc#title hasValue "Goal to find mortgage simulator with value restrictions"
dc#type hasValue _"http://www.wsmo.org/2004/d2#goals"
dc#description hasValue ""
dc#subject hasValue _"Simulator", "Mortgage", "Financial", "Product"
dc#date hasValue _date("2005-03-07")
dc#format hasValue "text/html"
dc#language hasValue "en-US"
dc#rights hasValue _"http://www.isoco.com/privacy.html"
wsml#version hasValue _$Revision: 10 $"
endnfp

importsOntology _"http://users.isoco.net/~slosada/ontologies/bankinter/FinancialOntology.wsml#"

capability
  sharedVariables ?mortgageLoan

precondition
  nfp
dc#description hasValue "The input has number of payments, type of interest and mortgage amount to simulate mortgage result is a monthly payment."
endnfp

definedBy
  ?mortgageLoan memberOf fin#MortgageLoan
  [ loanCapital hasValue ?capital,
    initialQuota hasValue ?quota,
    interestRateType hasValue ?interest
    and ?capital < 200000
    and ?quota < 900
    and ?interest memberOf productRateApplicationVariable[interestRateValue hasValue 0.5,
    referenceType hasValue _# ].

postcondition
  nfp
dc#description hasValue "Result of webService is a list of mortgage. The person who wants contract this mortgage must have life insurance."
endnfp

definedBy
  ?mortgageLoan memberOf fin#MortgageLoan
  and ?mortgageLoan [ term hasValue _#,
    openingCommission hasValue ?opCommission,
    lifeInsurance hasValue true,
    homeInsurance hasValue false ]
  and ?opCommission < 0.7.
```

**Listing 6**. Concrete goa example to find a mortgage simulator knowing the numbers of payment, the interest rate type and the total amount of the mortgage loan.

**precondition**: It’s necessary to know the end date of the payment of the mortgage, the interest rate type and the total amount of mortgage loan.
**postcondition:** Monthly payment of mortgage that has opening

```xml
capability
  sharedVariables ?mortgageLoan

precondition
  nfp
dc#description hasValue "The input has number of payments, type of interest and 
mortgage amount to simulate mortgage result is a monthly payment."
endnfp

definedBy
  ?mortgageLoan memberOf fin#MortgageLoan
  [ loanCapital hasValue ?capital,
  term hasValue ?term,
  interestRateType hasValue ?interest]
  and ?capital < 200000
  and ?term[ totalTerm hasValue ?totalTerm,
  typeTerm hasValue MONTH ]
  and ?totalTerm < 300
  and ?interest memberOf productRateApplicationVariable[interestRateValue hasValue 0.5,
  referenceType hasValue _# ].

postcondition
  nfp
dc#description hasValue "Result of webService is a list of mortgage. The person who wants 
contract this mortgage must have life insurance. "
endnfp

definedBy
  ?mortgageLoan memberOf fin#MortgageLoan
  and ?mortgageLoan [ 
  initialQuota hasValue _#,
  openingCommission hasValue ?opCommission,
  lifeInsurance hasValue true,
  homeInsurance hasValue false ]
  and ?opCommission < 0.7.
```

Listing 7. Concrete goal example to find mortgage simulator knowing the monthly payment, the interest rate type and the number of payments.
**precondition:** A monthly payment, an interest rate type and the number of payments. Mortgage must be interest rate variable along the time.

**postcondition:** Total amount of the mortgage that has an opening commission of 0.25 and its subrogation commission is 0.2

```xml
namespace { "http://users.isoco.net/~slosada/ontologies/bank3/GoalGetMortgageCapital.wsml#",
  dc _"http://purl.org/dc/elements/11#",
  foaf _"http://xmlns.com/foaf/01#",
  xsd _"http://www.w3c.org/2001/XMLSchema#",
  wsml _"http://www.wsmo.org/2004/wsml#",
  oo _"http://users.isoco.net/~slosada/ontologies/bank3/OWL2WSMLOntologyBank3.wsml#"}

goal _"http://users.isoco.net/~slosada/ontologies/bank3/GoalGetMortgageCapital.wsml"

  nfp
dc#title hasValue "Goal to find mortgage simulator with value restrictions"
dc#type hasValue _"http://www.wsmo.org/2004/d2#goals"
dc#description hasValue ""
dc#subject hasValue { "Simulator", "Mortgage", "Financial", "Product"}
dc#date hasValue _date("2005-03-30")
dc#format hasValue "text/html"
dc#language hasValue "en-US"
dc#rights hasValue _"http://www.isoco.com/privacy.html"
wsml#version hasValue "$Revision: 1.0 $"
endnfp

importsOntology { _"http://users.isoco.net/~slosada/ontologies/bankinter/FinancialOntology.wsml#"}

capability
sharedVariables ?mortgageLoan

  precondition
  nfp
dc#description hasValue "The input has number of payments, type of interest and monthly payment to simulate mortgage result is a mortgage amount."
endnfp

definedBy
?mortgageLoan memberOf oo#MortgageLoan
  and ?mortgageLoan [ term hasValue ?term, initialQuota hasValue ?quota, interestRateType hasValue ?interest ]
  and ?term memberOf Term
  and ?term[ totalTerm hasValue ?totalTerm, typeTerm hasValue MONTH ]
  and ?totalTerm < 298
  and ?interest memberOf productRateApplicationVariable [interestRateValue hasValue

  ?interestValue] and ?interestValue < 0.5
  and ?quota < 900.

  postcondition
  nfp
dc#description hasValue "The output is a list of mortgage amount."
endnfp

definedBy
?mortgageLoan memberOf MortgageLoan
  and ?mortgageLoan [ capital hasValue _#, openingCommission hasValue 0.25,
  subrogationCommission hasValue 0.2 ].
```
3.3. Web Services

We define the Web Services in this use case as end-user services (which means that the user interacts with this services) to simulate mortgages. This Web Services can be composed of other Web Services. The modeling for the WSMO Web Service Interface will be added in a later version. A Web Service Capability in WSMO is described by pre- and post-conditions, assumptions and effects, as defined in [5]. In this particular case when a mortgage is simulated, changes in the real world do not take place; for this reason the web services described above do not contain assumptions and effects.

3.3.1. Bankinter Services.


**precondition**: It is necessary to know monthly payment, interest rate type and number of payments.

**postcondition**: Total amount of mortgage loan that has opening commission 0.7 and its necessary contract life insurance for this mortgage.

```xml

webService _"http://users.isoco.net/~slosada/ontologies/bankinter/WSGetMortgageCapital.wsml"

  nfp
    dc#title hasValue "Web Service that is a mortgage simulator"
    dc#type hasValue (_"http://www.wsmo.org/2004/d2#webservice"
    dc#description hasValue ""
    dc#subject hasValue (_ Simulator, "Mortgage", "Financial", "Product")
    dc#date hasValue _date("2005-03-30")
    dc#format hasValue "text/html"
    dc#language hasValue "en-US"
    dc#rights hasValue (_"http://www.isoco.com/privacy.html"
  endnfp

importsOntology (_"http://users.isoco.net/~slosada/ontologies/bankinter/FinancialOntology.wsml#")

capability
  sharedVariables ?mortgageLoan

precondition
  nfp
    dc#description hasValue "The input has number of payments, type of interest and mortgage amount to simulate mortgage result is a monthly payment."
  endnfp

definedBy
  ?mortgageLoan memberOf fin#MortgageLoan
  [ term hasValue ?term,
    initialQuota hasValue ?quota,
    interestRateType hasValue ?interest]
  and ?term[ totalTerm hasValue ?totalTerm,
    typeTerm hasValue MONTH ]
  and ?totalTerm < 300
  and ?quota > 600
  and ?interest memberOf productRateApplicationVariable
```

- 29 -
WSMO descriptions of application.


**precondition**: It’s necessary to know the monthly payment, the interest rate type and the total amount of the mortgage loan.

**postcondition**: Number of payments of the mortgage that have an opening commission of 0.7 and it’s necessary contract life insurance for this mortgage

```xml
namespace { "http://users.isoco.net/~slosada/ontologies/bankinter/WSGetNumberOfPayment.wsml#"
, dc "http://purl.org/dc/elements/11#",
  foaf "http://xmlns.com/foaf/01/",
  xsd "http://www.w3c.org/2001/XMLSchema#",
  wsml "http://www.wsmo.org/2004/wsml#"
, fin "http://users.isoco.net/~slosada/ontologies/bankinter/FinancialOntology.wsml#"
}

webService _"http://users.isoco.net/~slosada/ontologies/bankinter/WSGetNumberOfPayment..wsml"

  nfp
dc#title hasValue "Web Service that is a mortgage simulator"
dc#type hasValue { "http://www.wsmo.org/2004/d2#webservice" }
dc#description hasValue ""
dc#subject hasValue { "Simulator", "Mortgage", "Financial", "Product"}
dc#date hasValue _date("2005-03-30")
dc#format hasValue "text/html"
dc#language hasValue "en-US"
dc#rights hasValue {_"http://www.isoco.com/privacy.html"}
wsml#version hasValue "$Revision: 1.0 $"
endnfp

importsOntology { _"http://users.isoco.net/~slosada/ontologies/bankinter/FinancialOntology.wsml#"
}

capability
  sharedVariables ?mortgageLoan
  precondition
    nfp
dc#description hasValue "The input has, monthly payment type of interest and mortgage amount to simulate mortgage result is a number of payments."
endnfp
  definedBy
?mortgageLoan memberOf fin#MortgageLoan
  [ LoanCapital hasValue ?capital,
    initialQuota hasValue ?quota,
    interestRateType hasValue ?interest]
and ?capital < 200000
and ?quota < 900
and ?interest memberOf productRateApplicationVariable]
```

[interestRateValue hasValue 0.5, referenceType hasValue _#].

**postcondition**

nfp
dc#description hasValue “Result of webService is a list of mortgage. The person who wants contract this mortgage must have life insurance. ”
endnfp

definedBy
?mortgageLoan memberOf fin#MortgageLoan
  and ?mortgageLoan
    loanCapital hasValue _#,
    openingCommission hasValue ?opCommission,
    lifeInsurance hasValue true,
    homeInsurance hasValue false ]
and ?opCommission < 0.7.


**precondition**: It’s necessary to know the monthly payment, the interest rate type and the total amount of the mortgage loan.

**postcondition**: Number of payments of the mortgage that have an opening commission of 0.7 and it’s necessary contract life insurance for this mortgage

```xml
namespace { "http://users.isoco.net/~slosada/ontologies/bankinter/WSGetNumberOfPayment.wsml#",
  dc "http://purl.org/dc/elements/11#",
  foaf "http://xmlns.com/foaf/01/",
  xsd "http://www.w3c.org/2001/XMLSchema#",
  wsml "http://www.wsmo.org/2004/wsml#"
, fin "http://users.isoco.net/~slosada/ontologies/bankinter/FinancialOntology.wsml#"
}

webService _"http://users.isoco.net/~slosada/ontologies/bankinter/WSGetNumberOfPayment..wsml"

  nfp
dc#title hasValue "Web Service that is a mortgage simulator"
dc#type hasValue { "http://www.wsmo.org/2004/d2#webservice" }
dc#description hasValue ""
dc#subject hasValue { "Simulator", "Mortgage", "Financial", "Product"}
dc#date hasValue _date("2005-03-30")
dc#format hasValue "text/html"
dc#language hasValue "en-US"
dc#rights hasValue {_"http://www.isoco.com/privacy.html"}
wsml#version hasValue "$Revision: 1.0 $"
endnfp

importsOntology { _"http://users.isoco.net/~slosada/ontologies/bankinter/FinancialOntology.wsml#"
}

capability
  sharedVariables ?mortgageLoan
  precondition
    nfp
dc#description hasValue "The input has, monthly payment type of interest and mortgage amount to simulate mortgage result is a number of payments."
endnfp
  definedBy
?mortgageLoan memberOf fin#MortgageLoan
  [ LoanCapital hasValue ?capital,
    initialQuota hasValue ?quota,
    interestRateType hasValue ?interest]
and ?capital < 200000
and ?quota < 900
and ?interest memberOf productRateApplicationVariable]
```

**precondition**: It’s necessary to know the number of payments, the interest rate type and the total amount of mortgage loan.

**postcondition**: Monthly payment of mortgage that has an opening commission of 0.7 and it’s necessary contract life insurance for this mortgage.
3.3.2. Bank1 services


**precondition**: It’s necessary to know a monthly payment, the interest rate type and the number of payments.

**postcondition**: Total amount of mortgage loan that has ab opening commission of 0.7 and a cancel commission 0.1.

```xml
namespace { _"http://users.isoco.net/~slosada/ontologies/bank1/WSGetMortgageCapitalBank1.wsml#",
  dc  _"http://purl.org/dc/elements/11#",
  foaf _"http://xmlns.com/foaf/01/",
  xsd _"http://www.w3c.org/2001/XMLSchema#",
  wsml _"http://www.wsmo.org/2004/wsml#",
  bank _"http://users.isoco.net/~slosada/ontologies/bank1/Bank1Ontology.wsml"
}

webService _"http://users.isoco.net/~slosada/ontologies/bank1/WSGetMortgageCapitalBank1.wsml"

  nfp
dc#title hasValue "Goal to find mortgage simulator with value restrictions"
dc#type hasValue _"http://www.wsmo.org/2004/d2#goals"
dc#description hasValue "Web Service for simulate mortgage"
dc#subject hasValue { "Simulator", "Mortgage", "Financial", "Product" }
dc#date hasValue _date("2005-03-07")
dc#format hasValue "text/html"
dc#language hasValue "en-US"
dc#rights hasValue _"http://www.isoco.com/privacy.html"
wsml#version hasValue "$Revision: 10 "$
endnfp

importsOntology { _"http://users.isoco.net/~slosada/ontologies/bank1/Bank1Ontology.wsml"}

capability
  sharedVariables ?mortgageLoan
  precondition
    nfp
dc#description hasValue "The input has number of payments, type of interest and mortgage amount to simulate mortgage result is a monthly payment."
endnfp
  definedBy
    ?mortgageLoan memberOf fin#MortgageLoan
    
    [ term hasValue ?term,
      initialQuota hasValue ?quota,
      interestRateType hasValue ?interest]```
WSMO descriptions of application.

Listing 12 Bank1 mortgage simulator Web Service.

**precondition:** It’s necessary to know a monthly payment, interest rate type and Total amount of mortgage loan.

**postcondition:** Number of payments of mortgage that has opening commission 0.7 and must contract life insurance.

```xml
namespace { "http://users.isoco.net/~slosada/ontologies/bank1/WSGetEndOfPaymentsBank1.wsml#",
dc _"http://purl.org/dc/elements/11#",
foaf _"http://xmlns.com/foaf/01/",
xsd _"http://www.w3c.org/2001/XMLSchema#",
wsml _"http://www.wsmo.org/2004/wsml#",
bank _"http://users.isoco.net/~slosada/ontologies/bank1/FinancialBank1.wsml"
}

webService _"http://users.isoco.net/~slosada/ontologies/bank1/WSGetEndOfPaymentsBank1.wsml"

nfp
dc#title hasValue "Web Service that is a mortgage simulator"
dc#type hasValue _"http://www.wsmo.org/2004/d2#webservice"
dc#description hasValue "",
dc#subject hasValue { "Simulator", "Mortgage", "Financial", "Product"}
dc#date hasValue _date("2005-03-14")
dc#format hasValue "text/html"
dc#language hasValue "en-US"
dc#rights hasValue _http://www.isoco.com/privacy.html"
wsml#version hasValue "$Revision: 1.0 $"
endnfp

importsOntology { _"http://users.isoco.net/~slosada/ontologies/bank1/Bank1Ontology.wsml"

capability
  sharedVariables ?mortgageLoan
precondition
  nfp
dc#description hasValue "The input has, monthly payment type of interest and mortgage amount to simulate mortgage result is a number of payments."
endnfp
definedBy
?mortgageLoan memberOf fin#MortgageLoan
  loanCapital hasValue ?capital,
  initialQuota hasValue ?quota,
  interestRateType hasValue ?interest
```
Listing 13 Bank1 mortgage simulator Web Service.

**precondition:** It’s necessary to know the end date of the payment of the mortgage, the interest rate type and the total amount of the mortgage loan.

**postcondition:** Monthly payment of the mortgage that has an opening commission of 0.7 and must contract life insurance

```xml
namespace _{"http://users.isoco.net/~slosada/ontologies/bank1/WSGetMonthlyPaymentBank1.wsml#",
  dc _"http://purl.org/dc/elements/11#",
  foaf _"http://xmlns.com/foaf/01/",
  xsd _"http://www.w3c.org/2001/XMLSchema#",
  wsml _"http://www.wsmo.org/2004/wsml#",
  bank _"http://users.isoco.net/~slosada/ontologies/bank1/Bank1Ontology.wsml"}

webService _"http://users.isoco.net/~slosada/ontologies/bank1/WSGetMonthlyPaymentBank1.wsml"

  nfp
    dc#title hasValue "Web Service that is a mortgage simulator"
    dc#description hasValue "The input has number of payments, type of interest and mortgage amount to simulate mortgage result is a monthly payment."
  endnfp

  importsOntology _{"http://users.isoco.net/~slosada/ontologies/bank1/Bank1Ontology.wsml"}

  capability
    sharedVariables ?mortgageLoan
    precondition
      nfp
dc#description hasValue "The input has number of payments, type of interest and mortgage amount to simulate mortgage result is a monthly payment."
  endnfp

  definedBy
    ?mortgageLoan memberOf fin#MortgageLoan
      term hasValue ?term,
        loanCapital hasValue ?capital,
        interestRateType hasValue ?interest
      and ?capital < 200000
        and ?term[ totalTerm hasValue ?totalTerm,
          typeTerm hasValue MONTH ]
        and ?totalTerm < 298
      and ?interest memberOf productRateApplicationVariable
        interestRateValue hasValue ?value,
        referenceType hasValue _#
      and ?value > 0.3.
  endnfp
```

```xml
postcondition
nfp
dc#description hasValue "Result of webService is a list of mortgage. The person who wants to contract this mortgage must have life insurance."
endnfp
```

```xml
definedBy
  ?mortgageLoan memberOf fin#MortgageLoan
    and ?mortgageLoan [ term hasValue _#, openingCommission hasValue ?opCommission,
      lifeInsurance hasValue true,
      homeInsurance hasValue false ]
    and ?opCommission < 0.7.
endnfp
```
3.3.3. Bank2 services


**precondition:** Mortgage simulator with known monthly payments, interest rate and number of payments.

**postcondition:** Mortgage capital of the mortgage that has an opening commission of 0.7, an interest delay commission of 0.1 and it is necessary to contract a life insurance for this mortgage.

```xml
namespace {"http://users.isoco.net/~slosada/ontologies/bank2/WSGetMortgageCapitalBank2.wsml#",
  dc      "http://purl.org/dc/elements/11#",
  foaf    "http://xmlns.com/foaf/01/",
  xsd     "http://www.w3c.org/2001/XMLSchema#",
  wsml    "http://www.wsmo.org/2004/wsml#",
  oo      "http://users.isoco.net/~slosada/ontologies/Bank2/OWL2WSMOntologyBank2.wsml#"}

webService _"http://users.isoco.net/~slosada/ontologies/bank2/WSGetMortgageCapitalBank2.wsml"

  nfp
  dc#title hasValue "Web Service that is a mortgage simulator"
  dc#type hasValue {"http://www.wsmo.org/2004/d2#service"}
  dc#description hasValue """"
  dc#subject hasValue {"Simulator", "Mortgage", "Financial", "Product"}
  dc#date hasValue _date("2005-03-30")
  dc#format hasValue "text/html"
  dc#language hasValue "en-US"
  dc#rights hasValue {"http://www.isoco.com/privacy.html"
  wsml#version hasValue "$Revision: 10 $"
endnfp

usesMediator _"http://users.isoco.net/~slosada/ontologies/Bank3/OWL2WSMOntologyBank2.wsml"

capability
  sharedVariables ?mortgageLoan

precondition
  nfp
  dc#description hasValue "The input has number of payments, type of interest and monthly payment to simulate mortgage result is a mortgage amount."
endnfp

definedBy
  ?mortgageLoan memberOf doi#MortgageLoan
  and ?mortgageLoan [ term hasValue ?term, initialQuota hasValue ?quota , interestRateType hasValue ?interest]
  and ?term[totalTerm hasValue ?totalTerm,}

**precondition:** It’s necessary to know the monthly payment, the interest rate type and the total amount of the mortgage loan.

**postcondition:** Number of payments of the mortgage that has an opening commission of 0.1, an interest delay commission of 0.1 and it’s necessary contract a life insurance for this mortgage.
precondition: It's necessary to know the number of payments, the interest rate type and the total amount of the mortgage loan.

postcondition: Monthly payment. of the mortgage that has an opening commission of 0.7, an interest delay commission of 0.1 and it’s necessary to contract a life insurance for this mortgage.


namespace {"http://users.isoco.net/~slosada/ontologies/bank2/WSGetMonthlyPaymentBank2.wsml#",
            dc  "http://purl.org/dc/elements/11#",
            foaf _"http://xmlns.com/foaf/01/",
            xsd _"http://www.w3c.org/2001/XMLSchema#",
            wsml _"http://www.wsmo.org/2004/wsml#",
            oo _"http://users.isoco.net/~slosada/ontologies/bank2/OWL2WSMLmediatorOntologyBank2.wsml#"}
webService _"http://users.isoco.net/~slosada/ontologies/bank2/WSGetMonthlyPaymentBank2.wsml"

nfp
dc#title hasValue "Web Service that is a mortgage simulator"
dc#type hasValue (_"http://www.wsmo.org/2004/d2#webservice")
dc#description hasValue ""
dc#subject hasValue { "Simulator", "Mortgage", "Financial", "Product"}
dc#date hasValue _date("2005-03-30")
dc#format hasValue "text/html"
dc#language hasValue "en-US"
dc#rights hasValue (_"http://www.isoco.com/privacy.html")
wsml#version hasValue "$Revision: 10 $"
endnfp

usesMediator { _"http://users.isoco.net/~slosada/ontologies/bank3/OWL2WSMLmediatorOntologyBank2.wsml" }
capability
  sharedVariables ?mortgageLoan

precondition
nfp
dc#description hasValue "The input has number of payments, type of interest and mortgage amount to simulate mortgage result is a monthly payment."
endnfp

definedBy

?mortgageLoan memberOf oo#MortgageLoan
    and ?mortgageLoan [ term hasValue ?term ,
    loanCapital hasValue ?capital ,
    interestRateType hasValue ?interest]
    and ?term[ totalTerm hasValue ?totalTerm ,
    typeTerm hasValue MONTH ]
    and ?totalTerm < 300
    and ?interest memberOf productRateApplicationVariable[interestRateValue hasValue 0.5 ,
    referenceType hasValue _# ]
    and ?capital > 230000.

postcondition
nfp
dc#description hasValue "The output is a list of mortgage amount."
endnfp
WSMO descriptions of application.

3.3.4. Bank3 services

Listing 17. Bank3 mortgage simulator Web Service.

**precondition:** A monthly payment, interest rate type and number of payments. Mortgage must be interest rate variable along the time.

**postcondition:** Total amount of mortgage which opening commission is 0.25 and subrogation commission is 0.2.

```xml
  dc "http://purl.org/dc/elements/11#",
  foaf "http://xmlns.com/foaf/01",
  xsd "http://www.w3c.org/2001/XMLSchema#",
  oo "http://users.isoco.net/~slosada/ontologies/Bank3/OWL2WSMLOntologyBank3.wsml#")


  nfp
dc#title hasValue "Web Service that is a mortgage simulator"
da#subject hasValue ("Simulator", "Mortgage", "Financial", "Product")
dc#description hasValue ""
dc#type hasValue ("_http://www.wsmo.org/2004/d2#webservice")
dc#format hasValue "text/html"
dc#language hasValue "en-US"
dc#rights hasValue ("_http://www.isoco.com/privacy.html")
wsml#version hasValue "$Revision: 10 $"
endnfp

usesMediator ("_http://users.isoco.net/~slosada/ontologies/Bank3/OWL2WSMLOntologyBank3.wsml")

capability

sharedVariables ?mortgageLoan

precondition

nfp
dc#description hasValue "The input has number of payments, type of interest and monthly payment to simulate mortgage result is a mortgage amount."
endnfp

definedBy

?mortgageLoan memberOf oo#MortgageLoan and ?mortgageLoan [ term hasValue ?term, initialQuota hasValue ?quota, interestRateType hasValue ?interest ] and ?term memberOf Term and ?term[ totalTerm hasValue ?totalTerm, typeTerm hasValue MONTH ] and ?totalTerm < 298 and ?interest memberOf productRateApplicationVariable [interestRateValue hasValue ?interestValue] and ?interestValue < 0.5 and ?quota < 900 .

postcondition

nfp
```
WSMO descriptions of application.


precondition: It’s necessary know the monthly payment, the interest rate type and the total amount of the mortgage loan. Mortgage must be interest rate variable along the time.

postcondition: Number of payments of the mortgage which opening commission is 0.25 and subrogation commission is 02

```xml
namespace {
  "http://users.isoco.net/~slosada/ontologies/bank3/WSGetNumberOfPaymentsBank3.wsml#",
  dc _"http://purl.org/dc/elements/11#",
  foaf _"http://xmlns.com/foaf/01/",
  xsd _"http://www.w3c.org/2001/XMLSchema#",
  wsml _"http://www.wsmo.org/2004/wsml#",
  oo _"http://users.isoco.net/~slosada/ontologies/Bank3/OWL2WSMLOntologyBank3.wsml#"
}

```

```xml
webService _"http://users.isoco.net/~slosada/ontologies/bank3/WSGetNumberOfPaymentsBank3.wsml"

nfp
dc#title hasValue "Web Service that is a mortgage simulator"
dc#type hasValue (_"http://www.wsmo.org/2004/d2#webservice")
dc#description hasValue ""
dc#subject hasValue ("Simulator", "Mortgage", "Financial", "Product")
dc#date hasValue _date("2005-03-30")
dc#format hasValue "text/html"
dc#language hasValue "en-US"
dc#rights hasValue (_"http://www.isoco.com/privacy.html")
wsml#version hasValue "$Revision: 10 $"
endnfp

usesMediator (_"http://users.isoco.net/~slosada/ontologies/Bank3/OWL2WSMLOntologyBank3.wsml")

capability
sharedVariables ?mortgageLoan

precondition
nfp
dc#description hasValue "The input has, monthly payment type of interest and mortgage amount to simulate mortgage result is a number of payments."
endnfp
definedBy

?mortgageLoan memberOf oo#MortgageLoan
and ?mortgageLoan [ loanCapital hasValue _#, openingCommission hasValue 0.25, subrogationCommission hasValue 0.2 ].

postcondition
nfp
dc#description hasValue "The output is a list of mortgage amount."
endnfp
definedBy

?mortgageLoan memberOf MortgageLoan
and ?mortgageLoan [ loanCapital hasValue _#, openingCommission hasValue 0.25, subrogationCommission hasValue 0.2 ].
```

- 39 -

**precondition**: A number of payments, interest rate type and total amount of mortgage loan. Mortgage must be interest rate variable along the time.

**postcondition**: Monthly payment of a mortgage which opening commission is 0.25 and subrogation commission is 0.2.

```xml
  dc       "http://purl.org/dc/elements/11#",
  foaf     "http://xmlns.com/foaf/01",
  xsd      "http://www.w3c.org/2001/XMLSchema#",
  wsml     "http://www.wsmo.org/2004/wsml#",
  oo       "http://users.isoco.net/~slosada/ontologies/bank3/OWL2WSMLOntologyBank3.wsml#"}


  usesMediator {"http://users.isoco.net/~slosada/ontologies/bank3/OWL2WSMLOntologyBank3.wsml#"}

  capability
    sharedVariables ?mortgageLoan

    precondition
      nfp
        dc#description hasValue "The input has number of payments, type of interest and mortgage amount to simulate mortgage result is a monthly payment." 
      endnfp

    definedBy
      ?mortgageLoan memberOf oo#MortgageLoan
      and ?mortgageLoan [term hasValue ?term, loanCapital hasValue ?capital, interestRateType hasValue ?interest ]
      and ?term memberOf Term
      and ?term [totalTerm hasValue ?totalTerm, typeTerm hasValue MONTH ]
      and ?totalTerm < 298
      and ?interest memberOf productRateApplicationVariable [interestRateValue hasValue ?interestValue]
      and ?interestValue < 0.5
      and ?capital < 200000.

    postcondition
      nfp
        dc#description hasValue "The output is a list of mortgage amount." 
      endnfp

    definedBy
      ?mortgageLoan memberOf MortgageLoan
      and ?mortgageLoan [ initialQuota hasValue _#, openingCommission hasValue 0.25, subrogationCommission hasValue 0.2].
```

WSMO descriptions of application.
3.4. Mediators.

Listing 20. OWL currency mediator Bankinter.
This mediator transforms currency ontology in OWL format into WSML format

```xml
namespace {'http://users.isoco.net/~slosada/ontologies/bankinter/owlCurrencyMediator.wsml#'
  dc _http://purl.org/dc/elements/11#",
  foaf _http://xmlns.com/foaf/01",
  xsd _http://www.w3c.org/2001/XMLSchema#,
  wsml _http://www.wsmo.org/2004/wsml#}

ooMediator _http://users.isoco.net/~slosada/ontologies/bankinter/owlCurrencyMediator.wsml
  nonFunctionalProperties
    dc#description hasValue "This ooMediator imports the owl currency
description of the financial ontology to wsml ontology."
    dc#type hasValue _http://www.wsmo.org/2004/d2/#ooMediator
    wsml#version hasValue "$Revision: 1.0 $"
  endNonFunctionalProperties

source _http://www.daml.ecs.soton.ac.uk/ont/currency.daml
target _http://users.isoco.net/~slosada/ontologies/bankinter/FinancialOntology.wsml
```

This mediator transforms currency ontology in OWL format into WSML format

```xml
namespace {'http://users.isoco.net/~slosada/ontologies/Bank1/OwlCurrencyMediator.wsml#'
  dc _http://purl.org/dc/elements/11#",
  foaf _http://xmlns.com/foaf/01",
  xsd _http://www.w3c.org/2001/XMLSchema#,
  wsml _http://www.wsmo.org/2004/wsml#}

ooMediator _http://users.isoco.net/~slosada/ontologies/Bank1/OwlCurrencyMediator.wsml
  nonFunctionalProperties
    dc#description hasValue "This ooMediator translates the owl
description of the financial ontology to wsml ontology."
    dc#type hasValue _http://www.wsmo.org/2004/d2/#ooMediator
    wsml#version hasValue "$Revision: 1.0 $"
  endNonFunctionalProperties

source _http://www.daml.ecs.soton.ac.uk/ont/currency.daml
target _http://users.isoco.net/~slosada/ontologies/bank1/Bank1Ontology.wsml
```

Listing 22. Bank2 OWL2WSML mediator.
This mediator transforms Bank2 ontology in OWL format into WSML format

```xml
namespace {'http://users.isoco.net/~slosada/ontologies/bank2/OWL2WSMLOntologyBank2.wsml#'
  dc _http://purl.org/dc/elements/11#",
  foaf _http://xmlns.com/foaf/01",
  xsd _http://www.w3c.org/2001/XMLSchema#,
  wsml _http://www.wsmo.org/2004/wsml#}

ooMediator _http://users.isoco.net/~slosada/ontologies/bank2/OWL2WSMLOntologyBank2.wsml
  nonFunctionalProperties
    dc#description hasValue "This ooMediator translates the owl
description of the domain ontology to wsml ontology."
    dc#type hasValue _http://www.wsmo.org/2004/d2/#ooMediator
```

- 41 -
This mediator transforms Bank3 ontology in OWL format into WSML format.

Listing 23. OWL2WSML mediator

```
namespace { _http://users.isoco.net/~slosada/ontologies/bank3/OWL2WSMLOntologyBank3.wsml#,
    dc _http://purl.org/dc/elements/11#,
    foaf _http://xmlns.com/foaf/01/,
    xsd _http://www.w3c.org/2001/XMLSchema#,
    wsml _http://www.wsmo.org/2004/wsml#}

ooMediator _http://users.isoco.net/~slosada/ontologies/bank3/OWL2WSMLOntologyBank3.wsml

nonFunctionalProperties
dc#description hasValue "This ooMediator translates the owl description of the domain ontology to wsml ontology."
dc#type hasValue _http://www.wsmo.org/2004/d2/#ooMediator
wsml#version hasValue "$Revision: 1.0 $"

endNonFunctionalProperties

source _http://users.isoco.net/~slosada/ontologies/bank3/OntologyBank3.owl

```

Listing 24. ooMediator Bank1 Ontology into Financial ontology.

```
namespace { _http://users.isoco.net/~slosada/ontologies/bank1/OntologyBank1toFinancialOntology.wsml#,
    dc _http://purl.org/dc/elements/11#,
    foaf _http://xmlns.com/foaf/01/,
    xsd _http://www.w3c.org/2001/XMLSchema#,
    wsml _http://www.wsmo.org/2004/wsml#}

ooMediator _http://users.isoco.net/~slosada/ontologies/bank1/OntologyBank1toFinancialOntology.wsml

nonFunctionalProperties
dc#description hasValue "This ooMediator translates the bank1 domain ontology in financial ontology"
dc#type hasValue _http://www.wsmo.org/2004/d2/#ooMediator
wsml#version hasValue "$Revision: 1.0 $"

endNonFunctionalProperties

source _http://users.isoco.net/~slosada/ontologies/bank1/OntologyBank1.owl

target _http://users.isoco.net/~slosada/ontologies/bank1/OntologyBank1.wsml

usesService _http://www.isoco.com/dataMediation/service
```

Listing 25. ooMediator Bank2 Ontology into Financial ontology.

```
namespace { _http://users.isoco.net/~slosada/ontologies/bank2/OntologyBank2toFinancialOntology.wsml#,
    dc _http://purl.org/dc/elements/11#,
    foaf _http://xmlns.com/foaf/01/,
    xsd _http://www.w3c.org/2001/XMLSchema#,
    wsml _http://www.wsmo.org/2004/wsml#}

ooMediator _http://users.isoco.net/~slosada/ontologies/bank2/OntologyBank2toFinancialOntology.wsml

nonFunctionalProperties
dc#description hasValue "This ooMediator translates the bank1 domain ontology in financial ontology"
```

- 42 -
in financial ontology"

```
namespace {"http://users.isoco.net/~slosada/ontologies/bank3/OntologyBank3toFinancialOntology.wsml#", 
    dc       "http://purl.org/dc/elements/11#",  
    foaf     "http://xmlns.com/foaf/01/",  
    xsd      "http://www.w3c.org/2001/XMLSchema#",  
    wsml     "http://www.wsmo.org/2004/wsml#"}

nonFunctionalProperties  
    dc#description hasValue "This ooMediator translates the bank3 domain ontology in financial ontology"  
    wsm#version hasValue "$Revision: 1.0 $"
endNonFunctionalProperties
source "http://users.isoco.net/~slosada/ontologies/bank3/OntologyBank3.wsml"  
target "http://users.isoco.net/~slosada/ontologies/bankinter/FinancialOntology.wsml"  
usesService "http://www.isoco.com/dataMediation/service
```


4. CONCLUSIONS

The purpose of this document is providing WSMO descriptions of mortgage comparison service, based on the four elements: WS, goals, ontologies and mediators. This deliverables illustrates in detail WSMO descriptions (of goals, mediators and WS) and the domain ontologies created to support the overall architecture.

These descriptions will be used to test the DIP tecnology and will be relevant input for WP 1,2,3,4,5,6. This deliverable show the application of SWS in banking sector an its befits.

As far as WSMO is concerned, we have been able to use it to experess all requirements of the financial sub-domain considered.

Since the chosen domain does not include transactions -only information provision- the WSMO constructors assumptions and effects have not necesarry.

REFERENCES


WSMO descriptions of application.


APENDIX A

Some of the ontologies used in this use case are modelled in OWL. In order to use them within WSML, the syntax has to be converted to WSML (using an OO Mediator). A mediation service must be developed. The following listings provide the ontologies in OWL abstract syntax. The conversion was done using the OWL API developed by the University of Manchester. The OWL API provides programmatic access to data structures representing OWL ontologies.

This bank uses extends proton and models concepts for this bank. PROTON Ontology (PROTo ONtology). has been developed in the scope of the SEKT Project.

Listing A 1 Bank2 Domain Ontology.

<table>
<thead>
<tr>
<th>Namespace(rdf)</th>
<th>= <a href="http://www.w3.org/1999/02/22-rdf-syntax-ns#">http://www.w3.org/1999/02/22-rdf-syntax-ns#</a></th>
</tr>
</thead>
<tbody>
<tr>
<td>Namespace(xsd)</td>
<td>= <a href="http://www.w3.org/2001/XMLSchema#">http://www.w3.org/2001/XMLSchema#</a></td>
</tr>
<tr>
<td>Namespace(rdfs)</td>
<td>= <a href="http://www.w3.org/2000/01/rdf-schema#">http://www.w3.org/2000/01/rdf-schema#</a></td>
</tr>
<tr>
<td>Namespace(owl)</td>
<td>= <a href="http://www.w3.org/2002/07/owl#">http://www.w3.org/2002/07/owl#</a></td>
</tr>
<tr>
<td>Namespace(a)</td>
<td>= <a href="http://users.isoco.net/~slosada/ontologies/bank2/OntologyBank2.owl#">http://users.isoco.net/~slosada/ontologies/bank2/OntologyBank2.owl#</a></td>
</tr>
<tr>
<td>Namespace(b)</td>
<td>= <a href="http://proton.semanticweb.org/2004/12/protonu#">http://proton.semanticweb.org/2004/12/protonu#</a></td>
</tr>
<tr>
<td>Namespace(c)</td>
<td>= <a href="http://proton.semanticweb.org/2004/12/protonf#">http://proton.semanticweb.org/2004/12/protonf#</a></td>
</tr>
</tbody>
</table>

Ontology( <http://users.isoco.net/~slosada/ontologies/bank2/OntologyBank2.owl> )

Annotation( owl:imports http://proton.semanticweb.org/2004/12/protonu)
DatatypeProperty(a:capital
domain(a:MortgageLoan)
range(xsd:double))
DatatypeProperty(a:delayInterestRate
domain(a:FutureMortgageLoan)
range(xsd:float))
DatatypeProperty(a:endingDate
domain(a:Quota)
range(xsd:date))
DatatypeProperty(a:expirationDate
domain(a:FinancialProduct)
range(xsd:date))
DatatypeProperty(a:handlingCapital
domain(a:FutureMortgageLoan)
range(xsd:float))
DatatypeProperty(a:homeInsurance
domain(a:FutureMortgageLoan)
range(xsd:boolean))
DatatypeProperty(a:initialQuota
domain(a:FutureMortgageLoan)
range(xsd:float))
DatatypeProperty(a:initialPeriod
domain(a:FutureMortgageLoan)
range(xsd:duration))
DatatypeProperty(a:interestDelay
domain(a:FutureMortgageLoan)
range(xsd:float))
DatatypeProperty(a:interestNextRevision
domain(a:FutureMortgageLoan)
range(xsd:float))
DatatypeProperty(a:interestRateValue
domain(unionOf(a:FinancialProduct a:ProductRateApplication))
range(xsd:float))
DatatypeProperty(a:lifeInsurance
domain(a:FutureMortgageLoan)
range(xsd:boolean))
DatatypeProperty(a:login
range(xsd:string))
DatatypeProperty(a:mortgageTaxation
domain(a:FutureMortgageLoan)
range(xsd:float))
DatatypeProperty(a:name
domain(a:FinancialProduct)
range(xsd:string))
DatatypeProperty(a:openingCommission Functional
domain(a:FutureMortgageLoan)
range(xsd:string))
DatatypeProperty(a:password
range(xsd:string))
DatatypeProperty(a:periodicityCuota
domain(a:FutureMortgageLoan)
range(xsd:duration))
DatatypeProperty(a:quotaAfterRevision
domain(a:FutureMortgageLoan)
range(xsd:float))
DatatypeProperty(a:revisionTerm
domain(a:FutureMortgageLoan)
range(xsd:duration))
DatatypeProperty(a:revisionTermNext
domain(a:MortgageLoan)
range(xsd:duration))
DatatypeProperty(a:saleCostProperty
domain(a:CurrentMortageLoan)
range(xsd:string))
DatatypeProperty(a:signalDateContract
domain(a:FinancialProduct)
range(xsd:date))
DatatypeProperty(a:startingDate
domain(a:Quota)
range(xsd:date))
DatatypeProperty(a:term
domain(a:FutureMortgageLoan)
range(xsd:duration))
DatatypeProperty(a:termRateFixed Functional
domain(a:ProductRateApplicationMixed)
range(xsd:duration))
DatatypeProperty(a:typeOfRate
range(oneOf("fixed"^^<http://www.w3.org/2001/XMLSchema#string>
"variable"^^<http://www.w3.org/2001/XMLSchema#string> "mixed"^^<http://www.w3.org/2001/XMLSchema#string>)))
DatatypeProperty(a:typeReferenceRate
range(oneOf("CECA"^^<http://www.w3.org/2001/XMLSchema#string>
"Euribor"^^<http://www.w3.org/2001/XMLSchema#string> "IRPH-ent"^^<http://www.w3.org/2001/XMLSchema#string>
"IRPH-cajas"^^<http://www.w3.org/2001/XMLSchema#string>)))
DatatypeProperty(a:value Functional
range(xsd:float))

Class(c:Person partial)
Class(c:Product partial)
Class(c:Service partial)
Class(b:Bank partial)
Class(b:Division partial)
Class(a:AddedValue partial
a:Service)
Class(a:AddedValue partial
annotation(rdfs:comment "The sales revenue from selling a product less the cost of the materials or purchases used
in those products. It is an indicator of relative efficiency within and between firms, although in the latter case it is open to
distortion where mark-up varies between standard and premium-priced segments of a market"@en)
annotation(rdfs:comment "Es el valor con que enriqueces algo al transformarlo. Normalmente se calcula como la
diferencia entre el valor de compra y el valor de venta, aunque es algo más intangible"@es))

Class(a:Administrative partial
a:Employee)
Class(a:Administrative partial
annotation(rdfs:comment "Employee with administrative functions"@en)
annotation(rdfs:comment "Empleado con funciones de administración."@es))

Class(a:Asset partial
a:FinancialProduct)
Class(a:Asset partial
annotation(rdfs:comment "The land or property of a company or individual, payments due from bills, investments, and
anything else owned that can be turned into cash"@en)
annotation(rdfs:comment "Activo bancario. Cuenta con saldo a favor del cliente."@es))

Class(a:Branch partial
a:Channel)
Class(a:Branch partial
annotation(rdfs:comment "Physical bank office"@en)
annotation(rdfs:comment "Oficina bancaria física"@es))

Class(a:Collection partial
a:Service)
Class(a:Collection partial
annotation(rdfs:comment "Ingreso en cuenta"@es)
annotation(rdfs:comment "Deposit in a saving account"@en))

Class(a:Commercial partial
a:Employee)
Class(a:Commercial partial
annotation(rdfs:comment "Empleado con funciones comerciales"@es)
annotation(rdfs:comment "Employee with commercial functions"@en))

Class(a:Corporative partial
a:Department)
Class(a:Corporative partial
annotation(rdfs:comment "Company with a specific set of characteristics that require a personalised commercial
strategy. Depending on each bank, the set of characteristics may change, although they normally refer to number of
employees and annual turnover. It includes company groups, multinational companies, etc."@en)
annotation(rdfs:comment "Empresa que por sus características merece un tratamiento comercial individualizado."@es))
Dependiendo de cada Banco estas características pueden variar, aunque normalmente se refieren a número de empleados y/o facturación anual. Incluye grupos de empresas, multinacionales, etc.

Class(a:CurrentMortgageLoan partial
  a:MortgageLoan)
Class(a:CurrentMortgageLoan partial
  annotation(rdfs:comment "Préstamo hipotecario actual que el usuario está pensando en cambiar"@es)
  annotation(rdfs:comment "Current mortgage loan that the user is willing to change."@en))
Class(a:Department partial
  b:Division)
Class(a:Department partial
  annotation(rdfs:comment "Internal area of the bank that performs a specific function"@en)
  annotation(rdfs:comment "Departamento. Area interna del Banco que tiene una misión dentro de Él"@es))
Class(a:Employee partial
  c:Person)
Class(a:FinancialProduct partial
  restriction(a:name cardinality(1))
  c:Product
  restriction(a:APR cardinality(1)))
Class(a:FinancialProduct partial
  annotation(rdfs:comment "Cualquier producto bancario que requiere la firma de un contrato (ejemplo: una cuenta corriente o un crédito)."@es)
  annotation(rdfs:comment "Bank product that requires the signature of a contract between the customer and the bank"@en))
Class(a:FutureMortgageLoan partial
  a:MortgageLoan)
Class(a:FutureMortgageLoan partial
  annotation(rdfs:comment "Préstamo hipotecario por el que el usuario está pensando cambiar su préstamo actual"@es)
  annotation(rdfs:comment "Future mortgage loan that the user will use instead of the current one"@en))
Class(a:Individuals partial
  a:Department)
Class(a:Individuals partial
  annotation(rdfs:comment "Department that deals with physical persons"@en))
Class(a:InvestmentAccount partial
  a:Asset)
Class(a:InvestmentAccount partial
  annotation(rdfs:comment "Account setup to perform an investment, such as a fixed term deposit"@en)
  annotation(rdfs:comment "Cuenta de inversión. Por ejemplo una imposición a plazo fijo (IPF) o un depósito remunerado"@es))
Class(a:InvestmentFund partial
  a:Asset)
Class(a:InvestmentFund partial
  annotation(rdfs:comment "Fondo de Inversión. Club de inversión donde muchas personas ponen un dinero que el Banco invierte en su nombre y cuyos beneficios/ pérdidas se reinvierten. Es disponible en cualquier momento vendiendo las participaciones que en su día se compraron."@es)
  annotation(rdfs:comment "Investment club where a set of customers put their money so that the bank performs an investment on behalf of them."@en))
Class(a:Invoice partial
  a:Payment)
Class(a:Invoice partial
  annotation(rdfs:comment "An itemized statement given to a buyer by a seller and usually specifying the price of goods or services and the terms of sale"@en)
  annotation(rdfs:comment "Factura"@es))
Class(a:Liability partial
  a:FinancialProduct)
Class(a:Liability partial
  annotation(rdfs:comment "The amount that is owed by an individual or company, whether money, products, or services, to others."@en)
  annotation(rdfs:comment "Cantidad que un individuo o compañía debe a otros. Esta cantidad puede ser dinero, productos o servicios"@es))
Class(a:Loan partial
  a:Asset)
Class(a:Loan partial
WSMO descriptions of application.

annotation(rdfs:comment "Préstamo, crédito, cualquier cuenta con saldo a favor del banco"@es)
annotation(rdfs:comment "Money let out at interest"@en)

Class(a:MortgageLoan partial a:Loan)
Class(a:MortgageLoan partial annotation(rdfs:comment "Préstamo hipotecario"@es)
annotation(rdfs:comment "A long-term loan backed by real estate or valuable property, usually the item purchased
with the loan. The creditor can claim that property if all payments are not made by the borrower when they are
due"@en)

Class(a:Payment partial a:Service)
Class(a:Payment partial annotation(rdfs:comment "Money given to pay for something"@en)
annotation(rdfs:comment "Pago"@es)

Class(a:ProductRateApplication partial b:Bank)
Class(a:ProductRateApplication partial annotation(rdfs:comment "Tipo de interés aplicado"@es)
annotation(rdfs:comment "Applied interest rate"@en)

Class(a:ProductRateApplicationFixed partial a:ProductRateApplication)
Class(a:ProductRateApplicationFixed partial annotation(rdfs:comment "tipo de interés fijo. Es un tipo de interés que no cambia en toda la vida del
préstamo."@es)
annotation(rdfs:comment "Fixed interest rate. It never varies during the mortgage life"@en)

Class(a:ProductRateApplicationMixed partial a:ProductRateApplication)
Class(a:ProductRateApplicationMixed partial annotation(rdfs:comment "tipo de interés mixto. Es una mezcla de tipo fijo y variable. Normalmente fijo en las
primeras cuotas y variable en las demás"@es)
annotation(rdfs:comment "Mixed interest rate, usually composed of fixed and variable interest rates"@en)

Class(a:ProductRateApplicationVariable partial a:ProductRateApplication)
Class(a:ProductRateApplicationVariable partial annotation(rdfs:comment "tipo de interés variable. Es un tipo de interés que se modifica periódicamente en función
de los tipos de interés de mercado que se hayan predefinido en contrato"@es)
annotation(rdfs:comment "Variable interest rate. It may vary during the mortgage life"@en)

Class(a:Quota partial b:Bank)
Class(a:Quota partial annotation(rdfs:comment "Amount to be paid in a loan"@en)

Class(a:SME partial a:Department)
Class(a:SME partial annotation(rdfs:comment "Pequeña y Mediana Empresa (PYME)"@es)
annotation(rdfs:comment "Small or Medium Enterprise"@en)

Class(a:SavingAccount partial a:Asset)
Class(a:SavingAccount partial annotation(rdfs:comment "Cuenta de Ahorro. Es lo mismo que una cuenta corriente, pero con libreta y sin talonario
de cheques. Normalmente está remunerada a un tipo de interés que suele ser bajo"@es)
annotation(rdfs:comment "Account without a chequebook and normally with a low interest rate"@en)

Class(a:Service partial c:Service)
Class(a:Service partial annotation(rdfs:comment "Service es el tipo de cosas que se pueden hacer pero que no requieren contrato (ejemplo:
una orden de transferencia). Service tambien es cualquier producto no financiero que distribuya un banco."@es)
annotation(rdfs:comment "Financial products offered by a bank or services that do not require a contract, like a bank
transfer order"@en)

Class(a:ServiceContractedByCustomerInChannel partial b:Bank)
Class(a:ServiceContractedByCustomerInChannel partial

- 49 -
This ontology extends a financial ontology based in SUMO ontology.

Listing A 2. Bank3 domain ontology.

```plaintext
Namespace(rdf) = <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
Namespace(xsd) = <http://www.w3.org/2001/XMLSchema#>
Namespace(rdfs) = <http://www.w3.org/2000/01/rdf-schema#>
Namespace(owl) = <http://www.w3.org/2002/07/owl#>
Namespace(a) = <http://users.isoco.net/~slosada/ontologies/bank3/OntologyBank3.owl#>
Namespace(b) = <http://users.isoco.net/~slosada/ontologies/FinancialSUMO.owl#>
Namespace(c) = <http://www.daml.ecs.soton.ac.uk/ont/currency.owl#>

Ontology(<http://users.isoco.net/~slosada/ontologies/bank3/OntologyBank3.owl>)
Annotation(owl:imports http://www.daml.ecs.soton.ac.uk/ont/currency.owl)
Annotation(owl:imports http://users.isoco.net/~slosada/ontologies/FinancialSUMO.owl)
```

```plaintext
ObjectProperty(a:CIF
domain(a:Company))
ObjectProperty(a:NIF
domain(unionOf(a:Person a:SOHO)))
ObjectProperty(a:cancelationCommission
Functional InverseFunctional
domain(a:CurrentMortageLoan)
range(a:Quota))
ObjectProperty(a:channel
domain(a:ServiceContractedByCustomerInChannel)
range(a:Channel))
ObjectProperty(a:currencyProduct
range(c:Currency))
ObjectProperty(a:customer
domain(a:ServiceContractedByCustomerInChannel)
range(a:Customer))
ObjectProperty(a:interestRateType
range(a:ProductRateApplication))
ObjectProperty(a:openingCommission
Functional
domain(a:FutureMortgageLoan)
range(a:Quota))
ObjectProperty(a:password
domain(a:User))
ObjectProperty(a:payments
domain(a:SavingAccount))
```
WSMO descriptions of application.

ObjectProperty(a:rateVariable
domain(a:ProductRateApplicationMixed)
range(a:ProductRateApplicationVariable))
ObjectProperty(a:service
domain(a:ServiceContractedByCustomerInChannel))
ObjectProperty(a:subrogationCommission Functional
domain(a:FutureMortgageLoan)
range(a:Quota))
ObjectProperty(a:titularity Symmetric
domain(a:Customer))

DatatypeProperty(a:APR
domain(a:Channel)
range(xsd:float))
DatatypeProperty(a:buyCostProperty
domain(a:FutureMortgageLoan)
range(xsd:string))
DatatypeProperty(a:capital
domain(b:Mortgage)
range(xsd:double))
DatatypeProperty(a:delayInterestRate
domain(a:FutureMortgageLoan)
range(xsd:float))
DatatypeProperty(a:endingDate
domain(a:Quota)
range(xsd:date))
DatatypeProperty(a:expirationDate
domain(a:Quota)
range(xsd:date))
DatatypeProperty(a:handlingCapital
domain(a:CurrentMortageLoan)
range(xsd:float))
DatatypeProperty(a:homeInsurance
domain(a:FutureMortgageLoan)
range(xsd:boolean))
DatatypeProperty(a:initialQuota
domain(a:FutureMortgageLoan)
range(xsd:float))
DatatypeProperty(a:initialPeriod
domain(a:FutureMortgageLoan)
range(xsd:float))
DatatypeProperty(a:interestDelay
domain(a:FutureMortgageLoan)
range(xsd:float))
DatatypeProperty(a:interestNextRevision
domain(a:FutureMortgageLoan)
range(xsd:float))
DatatypeProperty(a:interestRateValue
domain(a:ProductRateApplication)
range(xsd:float))
DatatypeProperty(a:lifeInsurance
domain(a:FutureMortgageLoan)
range(xsd:boolean))
DatatypeProperty(a:login
domain(a:User)
range(xsd:string))
DatatypeProperty(a:mortgageTaxation
domain(a:FutureMortgageLoan)
range(xsd:float))
DatatypeProperty(a:name
domain(a:Customer)
range(xsd:string))
DatatypeProperty(a:periodicyCuota
domain(a:FutureMortgageLoan)
range(xsd:duration))
DatatypeProperty(a:quotaAfterRevision
domain(a:FutureMortgageLoan)
range(xsd:float))
DatatypeProperty(a:revisionTerm
domain(a:FutureMortgageLoan)
range(xsd:duration))
DatatypeProperty(a:revisionTermNext
domain(b:Mortgage)
range(xsd:duration))
### WSMO descriptions of application.

<table>
<thead>
<tr>
<th>DatatypeProperty(a: saleCostProperty)</th>
<th>domain(a: CurrentMortageLoan)</th>
<th>range(xsd:string)</th>
</tr>
</thead>
<tbody>
<tr>
<td>DatatypeProperty(a: signalDateContract)</td>
<td>range(xsd:date)</td>
<td></td>
</tr>
<tr>
<td>DatatypeProperty(a: startingDate)</td>
<td>domain(a: Quota)</td>
<td>range(xsd:date)</td>
</tr>
<tr>
<td>DatatypeProperty(a: term)</td>
<td>domain(a: FutureMortgageLoan)</td>
<td>range(xsd:duration)</td>
</tr>
<tr>
<td>DatatypeProperty(a: termRateFixed)</td>
<td>domain(a: ProductRateApplicationMixed)</td>
<td>range(xsd:duration)</td>
</tr>
<tr>
<td>DatatypeProperty(a: typeOfRate)</td>
<td>range(oneOf(&quot;fixed&quot;^^<a href="http://www.w3.org/2001/XMLSchema#string">http://www.w3.org/2001/XMLSchema#string</a> &quot;mixed&quot;^^<a href="http://www.w3.org/2001/XMLSchema#string">http://www.w3.org/2001/XMLSchema#string</a> &quot;variable&quot;^^<a href="http://www.w3.org/2001/XMLSchema#string">http://www.w3.org/2001/XMLSchema#string</a>))</td>
<td></td>
</tr>
<tr>
<td>DatatypeProperty(a: typeReferenceRate)</td>
<td>range(oneOf(&quot;IRPH-ent&quot;^^<a href="http://www.w3.org/2001/XMLSchema#string">http://www.w3.org/2001/XMLSchema#string</a> &quot;CECA&quot;^^<a href="http://www.w3.org/2001/XMLSchema#string">http://www.w3.org/2001/XMLSchema#string</a> &quot;Euribor&quot;^^<a href="http://www.w3.org/2001/XMLSchema#string">http://www.w3.org/2001/XMLSchema#string</a> &quot;IRPH-cajas&quot;^^<a href="http://www.w3.org/2001/XMLSchema#string">http://www.w3.org/2001/XMLSchema#string</a>))</td>
<td></td>
</tr>
<tr>
<td>DatatypeProperty(a: value Functional)</td>
<td>range(xsd:float)</td>
<td></td>
</tr>
</tbody>
</table>

**Class (b: BankOrganization partial)**
- **Class (b: ConventionalMortgage partial)**
- **Class (b: Mortgage partial)**
- **Class (b: Payment partial)**
- **Class (b: TraditionalSavingsAccount partial)**
- **Class (b: interestEarned partial)**
- **Class (b: relations partial)**

**Class (a: Administrative partial)**
- **Employee**
  - annotation(rdfs:comment "Empleado con funciones de administración"@es)
  - annotation(rdfs:comment "Employee with administrative functions"@en)

**Class (a: Branch partial)**
- **Channel**
  - annotation(rdfs:comment "Oficina bancaria física"@es)
  - annotation(rdfs:comment "Physical bank office"@en)

**Class (a: Commercial partial)**
- **Employee**
  - annotation(rdfs:comment "Employee with commercial functions"@en)

**Class (a: Company partial)**
- **Customer**
  - annotation(rdfs:comment "A number of people grouped together as a business enterprise. Types of companies include public limited companies, partnerships, joint ventures and proprietorships, and branches of foreign companies"@en)

**Class (a: Corporative partial)**
- **Department**
  - annotation(rdfs:comment "Company with a specific set of characteristics that require a personalised commercial treatment. Depending on each bank, the set of characteristics may change, although they normally refer to number of employees and annual turnover. It includes company groups, multinational companies, etc."@en)
  - annotation(rdfs:comment "Empresa que por sus características merece un tratamiento comercial individualizado. Dependiendo de cada Banco estas características pueden variar, aunque normalmente se refieren a número de empleados y/o facturación anual. Incluye grupos de empresas, multinacionales, etc."@es)
WSMO descriptions of application.

Class(a:Currency partial
c:Currency)
 annotation(rdfs:comment "Money in circulation"@en)
 annotation(rdfs:comment "Divisa, incluyendo la moneda nacional"@es)
)

Class(a:CurrentMortgageLoan partial
b:ConventionalMortgage)
 annotation(rdfs:comment "Current mortgage loan that the user is willing to change."@en)
 annotation(rdfs:comment "Préstamo hipotecario actual que el usuario está pensando en cambiar"@es)
)

Class(a:Customer partial
a:User)
 annotation(rdfs:comment "Cliente que mantiene una relación contractual con el banco (normalmente una cuenta bancaria)"@es)
 annotation(rdfs:comment "Bank client, who usually has a contractual relationship with the bank"@en)
)

Class(a:Department partial
a:User)
 annotation(rdfs:comment "Departamento. Área interna del Banco que tiene una misión dentro de él"@es)
 annotation(rdfs:comment "Internal area of the bank that performs a specific function"@en)
)

Class(a:Employee partial
a:User)

Class(a:FutureMortgageLoan partial
b:ConventionalMortgage)
 annotation(rdfs:comment "Future mortgage loan that the user will use instead of the current one"@en)
)

Class(a:Individuals partial
a:Department)
 annotation(rdfs:comment "Department that deals with physical persons"@en)
)

Class(a:Invoice partial
a:Payment)
 annotation(rdfs:comment "An itemized statement given to a buyer by a seller and usually specifying the price of goods or services and the terms of sale"@en)
 annotation(rdfs:comment "Factura"@es)
)

Class(a:Person partial
a:Customer)
 annotation(rdfs:comment "Persona física o jurídica"@es)
 annotation(rdfs:comment "Bank client that represents a single person (physical or juridical)"@en)
)

Class(a:ProductRateApplication partial
a:ProductRateApplicationFixed)
 annotation(rdfs:comment "Applied interest rate"@en)
 annotation(rdfs:comment "Tipo de interés aplicado"@es)
)

Class(a:ProductRateApplicationFixed partial
a:ProductRateApplicationMixed)
 annotation(rdfs:comment "Fixed interest rate. It never varies during the mortgage life"@en)
 annotation(rdfs:comment "tipo de interés fijo. Es un tipo de interés que no cambia en toda la vida del préstamo"@es)
)

Class(a:ProductRateApplicationMixed partial
a:ProductRateApplicationVariable)
 annotation(rdfs:comment "Mixed interest rate, usually composed of fixed and variable interest rates"@en)
 annotation(rdfs:comment "tipo de interés mixto. Es una mezcla de tipo fijo y variable. Normalmente fijo en las primeras cuotas y variable en las demas"@es)
)

Class(a:ProductRateApplicationVariable partial
a:ProductRateApplication)
WSMO descriptions of application.

- Class(a:ProductRateApplicationVariable partial annotation(rdfs:comment "Variable interest rate. It may vary during the mortgage life"@en)
  annotation(rdfs:comment "tipo de interés variable. Es un tipo de interés que se modifica periódicamente en función de los tipos de interés de mercado que se hayan predefinido en contrato"@es)
}

- Class(a:Quota partial annotation(rdfs:comment "Amount to be paid in a loan"@en)
}

- Class(a:SME partial a:Department)

- Class(a:SME partial annotation(rdfs:comment "Small or Medium Enterprise"@en)
  annotation(rdfs:comment "Pequeña y Mediana Empresa (PYME)"@es)
}

- Class(a:SOHO partial a:Customer)

- Class(a:SOHO partial annotation(rdfs:comment "Small Office, Home Office. Normally se refiere a profesionales que trabajan en su propio despacho o negocio"@es)
  annotation(rdfs:comment "Small Office, Home Office. It usually refers to professionals who work in their own offices"@en)
}

- Class(a:SavingAccount partial b:TraditionalSavingsAccount)

- Class(a:SavingAccount partial annotation(rdfs:comment "Account without a chequebook and normally with a low interest rate"@en)
  annotation(rdfs:comment "Cuenta de Ahorro. Es lo mismo que una cuenta corriente, pero con libreta y sin talonario de cheques. Normalmente está remunerada a un tipo de interés que suele ser bajo"@es)
}

- Class(a:ServiceContractedByCustomerInChannel partial b:relations)

- Class(a:ServiceContractedByCustomerInChannel partial annotation(rdfs:comment "Productos contratados por un canal"@es)
  annotation(rdfs:comment "Product or service contracted by a channel"@en)
}

- Class(a:Staff partial a:Employee)

- Class(a:Staff partial annotation(rdfs:comment "Central Services of a company, such as Human Resources, Management, Innovation, etc."@en)
  annotation(rdfs:comment "Servicios Centrales de una empresa. Por ejemplo: Recursos humanos, Alta dirección, Inmuebles y Servicios Generales, Innovación, etc."@es)
}

- Class(a:User partial)

- Class(a:vBanking partial a:Channel)

- Class(a:vBanking partial annotation(rdfs:comment "Virtual Banking. Banca por medios físicos o sin intermediación humana."@es)
  annotation(rdfs:comment "Virtual Banking. Banking without human intervention"@en)
}

- Class(c:Currency partial)

- AnnotationProperty(rdfs:comment)

- DisjointClasses(a:SOHO a:Company)

- DisjointClasses(a:ProductRateApplicationVariable a:ProductRateApplicationMixed)

- DisjointClasses(a:Person a:SOHO)

- DisjointClasses(a:ProductRateApplicationVariable a:ProductRateApplicationMixed)

- DisjointClasses(a:ProductRateApplicationFixed a:ProductRateApplicationMixed)

- DisjointClasses(a:ProductRateApplicationFixed a:ProductRateApplicationMixed)

- DisjointClasses(a:Person a:Company)

- DisjointClasses(a:ProductRateApplicationFixed a:ProductRateApplicationVariable)

- DisjointClasses(a:ProductRateApplicationFixed a:ProductRateApplicationVariable)

- DisjointClasses(a:Person a:Company)
WSMO descriptions of application.