



Data, Information and Process Integration
with Semantic Web Services

DIP

Data, Information and Process Integration with Semantic Web Services

FP6 - 507483

Deliverable

WP10: Case study eBanking

D10.3

Financial Ontology

Mónica Martínez Montes

José Luís Bas

Sergio Bellido

Oscar Corcho

Silvestre Losada

22 December 2004

EXECUTIVE SUMMARY

This document describes the financial ontology that has been created for the first eBanking case study in DIP, which is described in the deliverable D10.2. This application presents a mortgage simulation and comparison service.

This financial ontology consists of several ontologies at different levels of abstraction: services and products; and channels, users and currencies. All of them are described in section 4. Some of these ontologies are based on IFX¹ (International Financial eXchange Forum) standard for the financial area.

Building an ontology in the financial area is difficult, due to the fact that it is a dynamic domain: new products appear on almost a daily basis and some of them cannot be categorised a priori. Besides, there is a high complexity in the current financial standards, such as IFX, and the agreement between different financial entities is difficult as well.

The financial ontology has been developed following the Methontology methodology [4].

This deliverable is especially relevant to the workpackage 3.

¹ <http://www.ifxforum.org/ifxforum.org/standards/index.cfm>

Document Information

IST Project Number	FP6 – 507483	Acronym	DIP
Full title	Data, Information, and Process Integration with Semantic Web Services		
Project URL	http://dip.semanticweb.org		
Document URL			
EU Project officer	Daniele Rizzi		

Deliverable	Number	10.3	Title	Financial Ontology
Work package	Number	10	Title	Case Study eBanking








Date of delivery	Contractual	M 12	Actual	31-12-04
Status	Version. 1.0		final <input type="checkbox"/>	
Nature	Prototype <input type="checkbox"/> Report <input checked="" type="checkbox"/> Dissemination <input type="checkbox"/>			
Dissemination Level	Public <input type="checkbox"/> Consortium <input checked="" type="checkbox"/>			

Authors (Partner)	Mónica Martínez Montes (Bankinter), José Luís Bas (Bankinter), Sergio Bellido (Bankinter), Oscar Corcho (iSOCO), Silvestre Losada (iSOCO)			
Responsible Author	Mónica Martínez Montes		Email	mmtnez@bankinter.es
	Partner	Fundación de la Innovación. Bankinter	Phone	

Abstract (for dissemination)	This document describes a financial ontology that has been developed for the first eBanking case study (mortgage simulator/comparator).
Keywords	Financial ontology, mortgage

Version Log			
Issue Date	Rev No.	Author	Change
19-10-04	001	Oscar Corcho, Silvestre Losada	Table of contents
2-11-04	002	José Luis Bas, Sergio Bellido, Mónica Martínez, Oscar Corcho, Silvestre Losada	Added introduction Added first draft of ontology description
6-12-04	003	Oscar Corcho, José Luis Bas	Version for QA
20-12-04	004	Oscar Corcho	Final version after QA

Project Consortium Information

Partner	Acronym	Contact
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 Innovacion. BankInter Paseo Castellana, 29 28046 Madrid, Spain Email: mmtnez@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 Rowlett, 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 Forschungszentrum Informatik Haid-und-Neu Strasse 10-14 76131 Karlsruhe Germany Email: abecker@fzi.de Tel: +49 721 9654 0

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  Changing the rules of business	Christian de Sainte Marie 9 Rue de Verdun, 94253 Gentilly, France Email: csm@ilog.fr Tel: +33 1 49082981
inubit AG	Inubit  the integration experts	Torsten Schmale inubit AG Lützowstraße 105-106 D-10785 Berlin Germany Email: ts@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: rbenjamins@isoco.com Tel. +34 913 349 797
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 Dörner SAP Research, CEC Karlsruhe SAP AG Vincenz-Priessnitz-Str. 1 76131 Karlsruhe, Germany Email: elmar.dorner@sap.com Tel: +49 721 6902 31
Sirma AI Ltd.	Sirma  Ontotext Knowledge and Language Engineering Lab of Sirma	Atanas Kiryakov, Ontotext Lab, - Sirma AI EAD Office Express IT Centre, 3rd Floor 135 Tzarigradsko Chausse Sofia 1784, Bulgaria Email: atanas.kiryakov@sirma.bg Tel.: +359 2 9768 303
Tiscali Österreich GmbH	Tiscali 	Dieter Haacker Tiscali Österreich GmbH. Diefenbachgasse 35 A-1150 Vienna Austria Email: Dieter.Haacker@at.tiscali.com

		Tel: +43 1 899 33 160
Unicorn Solution Ltd.	Unicorn 	Jeff Eisenberg Unicorn Solutions Ltd, Malcha Technology Park 1 Jerusalem 96951 Israel Email: Jeff.Eisenberg@unicorn.com Tel.: +972 2 6491111
Vrije Universiteit Brussel	VUB 	Carlo Wouters Starlab- VUB Vrije Universiteit Brussel Pleinlaan 2, G-10 1050 Brussel ,Belgium Email: carlo.wouters@vub.ac.be Tel.: +32 (0) 2 629 3719

TABLE OF CONTENTS

EXECUTIVE SUMMARY	I
TABLE OF CONTENTS	VI
LIST OF FIGURES	VII
1 INTRODUCTION	1
2 ONTOLOGY BUILDING METHODOLOGIES	1
2.1 Ontology Development Process	2
2.2 Ontology Life Cycle	3
3 DEVELOPMENT OF THE FINANCIAL ONTOLOGY	3
3.1 Specification. Goal and scope of the ontology	3
3.2 Knowledge sources.....	5
4 CONCEPTUALISATION	5
4.1 Knowledge Architecture.....	5
4.2 Ontology conceptualisation: Service ontology.....	6
4.3 Ontology conceptualisation: Product ontology	7
4.4 Ontology conceptualisation: Channel ontology	11
4.5 Ontology conceptualisation: User ontology	11
4.6 Ontology conceptualisation: Currency ontology.....	13
5 IMPLEMENTATION	14
6 REFERENCES	14

LIST OF FIGURES

Figure 1. Methontology development process (adapted from [3]).	3
Figure 2. MindMap describing the main components of the financial ontology	4
Figure 3. IFX as a framework for interoperability of financial systems.	5

1 INTRODUCTION

The overall objective of WP10 in the DIP integrated project is the development of a case study in the ebanking domain. The case study selected as a result of deliverable D10.1 [2] is a Simulation tool for Mortgage Comparison.

The financial ontology described in this deliverable has been developed using Methontology [4], a methodology for ontology construction that has been recommended in workpackage 3 for the development of the ontologies needed in all the workpackages.

This ontology is based on existing widely known and used standards in the financial domain, such as IFX² (Interactive Financial eXchange). It is beyond the scope of this deliverable and of the specific case study the deepening in all the branches of this standard in general. For this reason, we have focused on refining those parts of IFX that are applicable to our specific case study, as follows:

- 1) We have considered all the cases that can be applied to mortgage processes, which are about the comparison and simulation of mortgages.
- 2) We have established relationships between all the concepts available from the same point of view, with special attention to the possible combinations of information that a simulator/comparison tool can perform.

The current version of the financial ontology is the result of the analysis of the mortgage offerings provided by a representative set of Spanish banks that have an Internet presence providing quotes for mortgage offering services. From this analysis we extracted the most representative concepts, unifying the different ways used to express them and removing duplicates. For instance, there are several bases that can be applied to calculate periodically the rates of mortgages at a variable rate. All of them have been grouped under the concept *ProductRateApplication*.

To establish relations between concepts, we have filtered those concepts that can be obtained by using other pieces of data. For instance, the number of quotas can be expressed in years, months or number of quotas. All of them are grouped under a common concept (*quota*), and they can be translated to the format used by each simulation Web service provided by each entity. For instance, if an entity applies 14 annual quotas instead of 12, this entity will be in charge of transforming the data available in the registry.

On the other hand we have tried to cover the most common cases of the European mortgage market, instead of covering all the cases and peculiarities that can occur in all the national markets. For instance, there are mortgages whose rates are related to specific balances in a saving account or to other financial products. These situations would have a negative influence in the complexity of the framework of this case study, hence losing our general focus.

2 ONTOLOGY BUILDING METHODOLOGIES

As described in the introduction of this deliverable, and as proposed by WP3, we have used Methontology [4] to develop the ontology described in this deliverable.

² <http://www.ifxforum.org/ifxforum.org/standards/index.cfm>

Methontology enables the construction of ontologies at the knowledge level and includes [1]: the identification of the ontology development process, a life cycle based on evolving prototypes, and particular techniques for carrying out each activity. Methontology is supported by ODE [1], [4] and WebODE, although other ontology tools can be used to create ontologies with it. In this case, Protégé-2000 [7] with its OWL plug-in [6] has been used.

2.1 Ontology Development Process

The ontology development process [3] refers to *which* activities are carried out when building ontologies. It is crucial to identify these activities in order to reach agreement on ontologies that are to be built co-operatively by geographically distant teams, with some assurance of correctness and completeness. If this is the case, it is advisable to perform the three categories of activities presented below and steer clear of anarchic constructions (Figure 1).

- **Project Management Activities** include planning, control and quality assurance. *Planning* identifies which tasks are to be performed, how they will be arranged, how much time and what resources are needed for their completion. This activity is essential for reusing ontologies which have already been built or for building ontologies that require different levels of abstraction and generalisation. *Control* guarantees that planned tasks are completed according to the way they were intended to be performed. Finally, *Quality Assurance*, assures that the quality of each and every product outputted (ontology, software and documentation) is satisfactory. [8] describes how these activities are performed.
- **Development-Oriented Activities** include specification, conceptualisation, formalisation and implementation. *Specification* states why the ontology is being built, what are its intended uses and who are the end-users. *Conceptualisation* structures the domain knowledge as meaningful models at the knowledge level. *Formalisation* transforms the conceptual model into a formal or semi-computable model. *Implementation* builds computable models in a computational language. Finally, *Maintenance* updates and corrects the ontology. [4] gives details of how all the development activities, except Formalisation and Maintenance, are performed.
- **Support Activities** include a series of activities, performed at the same time as development-oriented activities, without which no ontology could be built. They include knowledge acquisition, evaluation, integration, documentation and configuration management. *Knowledge Acquisition* acquires knowledge of a given domain. *Evaluation* makes a technical judgement of the ontologies, their associated software environments and documentation with respect to a frame of reference during each phase and between phases of their life cycle [5]. *Integration* of ontologies is required when building a new ontology reusing other ontologies that are already available. *Documentation* details, clearly and exhaustively, each and every one of the phases completed and products generated. *Configuration Management* records all the versions of the documentation, software and ontology code to control the changes. In [4], [5], a description is given of how Knowledge Acquisition was performed in the CHEMICALS ontology (an ontology about chemical elements and their properties), and Evaluation, Integration and Configuration Management is discussed in [6], where the documentation produced is discussed as part of the description of each activity.

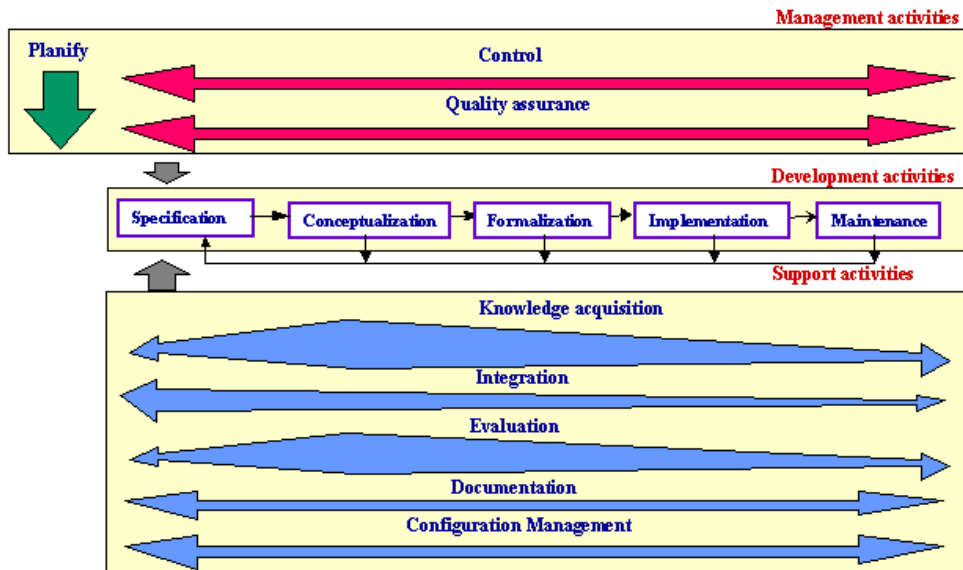


Figure 1. Methontology development process (adapted from [3]).

2.2 Ontology Life Cycle

It identifies the *set of stages* through which the ontology moves during its lifetime, describes what activities are to be performed in each stage and how the stages are related (relation of precedence, return, etc.). In [3], a justification is given of why the ontology life cycle should be based on evolving prototypes. For each prototype, Methontology proposes to begin with the *specification* of the ontology. Simultaneously with this phase, the knowledge acquisition activity starts. Once the first prototype has been specified, the construction of the conceptual model is built at the *conceptualisation* phase. It is like assembling a jigsaw puzzle from the pieces supplied by the knowledge acquisition activity. The puzzle is completed during the conceptualisation stage [4]. After the conceptualisation, *formalisation* and *implementation* of knowledge are carried out.

Formalisation is not a mandatory activity, because using ontology tools the conceptualisation model is usually automatically implemented with translators to ontology languages.

Control, quality assurance, integration, evaluation, documentation, and configuration management are carried out simultaneously to the development activities. However, the stage where the effort for doing integration and evaluation is bigger is the conceptualisation one.

3 DEVELOPMENT OF THE FINANCIAL ONTOLOGY

3.1 Specification. Goal and scope of the ontology

In the specification phase we have also used an approach that have been proposed in another ontology building methodology, On-To-Knowledge [9] for the kick-off phase of the ontology building process. We present in figure 2 a MindMap® with the main components of the ontology, obtained as the result of several brainstorming sessions with financial experts. The elements in this map are described in the following sections.

3.2 Knowledge sources

As specified in the introduction, the financial ontology is based on the Interactive Financial eXchange (IFX), which is an XML-based, financial messaging protocol, built by financial industry and technology leaders. Its goal is twofold:

- To use real business use cases and develop content that is meaningful and useful to the financial services industry.
- To create a strong, flexible, open architecture that will support extending the protocol in an efficient, interoperable manner.

Figure 3 shows that IFX is designed for interoperability of systems seeking to exchange financial information internally and externally. This means that not only does IFX address the data exchange requirements for the environments defined above, but it also allows interoperability among all of the business areas. For the sake of developing the financial ontology described in this deliverable we only focus on the data exchange aspects.

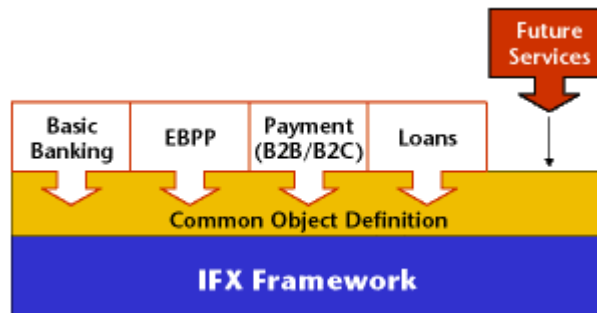


Figure 3. IFX as a framework for interoperability of financial systems.

Currently IFX provides content rich conversations in the areas of:

- Electronic Bill Presentment and Payment.
- Business to Business Payments.
- Business to Business Banking (such as balance and transaction reporting, remittance information).
- Automated Teller Machine communications.
- Consumer to Business Payments.
- Consumer to Business Banking.

4 CONCEPTUALISATION

In this section we present the conceptualisation of the ontology according to the intermediate representations proposed by Methontology for this conceptualisation phase.

4.1 Knowledge Architecture

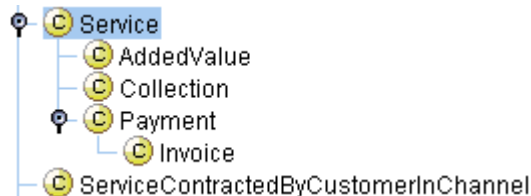
The financial ontology is composed of several ontologies at different levels of abstraction: services and products, channels, users, and currencies. In the following sections we provide the details of these ontologies.

4.2 Ontology conceptualisation: Service ontology

Terms glossary

Name	Description	Type
Service	Financial products offered by a bank or services that do not require a contract, like a bank transfer order	Concept
AddedValue	Bank service. 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	Concept
Collection	Bank service. Deposit in a saving account	Concept
Payment	Bank service. Money given to pay for something	Concept
Invoice	An itemized statement given to a buyer by a seller and usually specifying the price of goods or services and the terms of sale	Concept
ServiceContractedByCustomerInChannel	Product or service contracted by a channel	Concept
service(<i>ServicesContractedByCustomerInChannel, Service</i>)	Service contracted	Relation
channel(<i>ServicesContractedByCustomerInChannel, Channel</i>)	Channel where the contract has taken place	Relation
customer(<i>ServicesContractedByCustomerInChannel, Customer</i>)	Customer that has contracted a service	Relation

Concept classification tree



Concept dictionary

Concept name	Class attributes	Instance attributes	Relations
Service	--	--	--
AddedValue	--	--	--
Collection	--	--	--
Payment	--	--	--
Invoice	--	--	--
ServiceContractedByCustomerInChannel	--	--	service channel customer

Binary relation table

Relation name	Source concept	Source cardinality (Max)	Target concept
service	ServicesContractedByCustomerInChannel	n	Service
channel	ServicesContractedByCustomerInChannel	n	Channel
customer	ServicesContractedByCustomerInChannel	n	Customer

4.3 Ontology conceptualisation: Product ontology

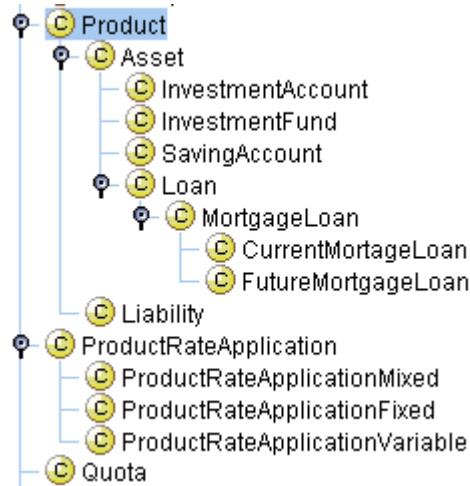
Terms glossary

Name	Description	Type
Product	Bank product that requires the signature of a contract between the customer and the bank	Concept
Asset	The land or property of a company or individual, payments due from bills, investments, and anything else owned that can be turned into cash	Concept
InvestmentAccount	Account setup to perform an investment, such as a fixed term deposit	Concept
InvestmentFund	Investment club where a set of customers put their money so that the bank performs an investment on behalf of them.	Concept
SavingAccount	Account without a chequebook and normally with a low interest rate	Concept
Loan	Money let out at interest	Concept
MortgageLoan	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	Concept
CurrentMortgageLoan	Current mortgage loan that the user is willing to change.	Concept
FutureMortgageLoan	Future mortgage loan that the user will use instead of the current one	Concept
Liability	The amount that is owed by an individual or company, whether money, products, or services, to others.	Concept
ProductRateApplication	Applied interest rate	Concept
ProductRateApplicationMixed	Mixed interest rate, usually composed of fixed and variable interest rates	Concept
ProductRateApplicationFixed	Fixed interest rate. It never varies during the mortgage life	Concept
ProductRateApplicationVariable	Variable interest rate. It may vary during the mortgage life	Concept
Quota	Amount to be paid in a loan	Concept
cancelationCommission	Commission to be paid when a mortgage is cancelled	Relation
openingCommission	Commission to be paid when a mortgage is opened	Relation
subrogationCommission	Commission to be paid when a mortgage is subrogated	Relation
currencyProduct	Currency of a bank product	Relation

interestRateType	Type of interest rate of a bank product	Relation
payments	Payments due in a saving account	Relation
handlingCapital	--	Instance Attribute
interestNextRevision	--	Instance Attribute
saleCostProperty	--	Instance Attribute
buyCostProperty	--	Instance Attribute
quotaAfterRevision	--	Instance Attribute
delayInterestRate	--	Instance Attribute
homeInsurance	--	Instance Attribute
initalQuota	--	Instance Attribute
initialPeriod	--	Instance Attribute
interesDelay	--	Instance Attribute
lifeInsurance	--	Instance Attribute
mortagageTaxation	--	Instance Attribute
periodicityQuota	--	Instance Attribute
revisionTerm	--	Instance Attribute
term	--	Instance Attribute
interestRateValue	--	Instance Attribute
revisionTermNext	--	Instance Attribute
capital	--	Instance Attribute
APR	--	Instance Attribute
expirationDate	--	Instance Attribute
signalDateContract	--	Instance Attribute

termRateFixed	--	Instance Attribute
endingDate	--	Instance Attribute
startingDate	--	Instance Attribute

Concept classification tree



Concept dictionary

Concept name	Class attributes	Instance attributes	Relations
Product	--	APR expirationDate signalDateContract interestRateValue	currencyProduct interestRateType
Asset	--	--	--
InvestmentAccount	--	--	--
InvestmentFund	--	--	--
SavingAccount	--	--	payments
Loan	--	--	--
MortgageLoan	--	revisionTermNext capital	--
CurrentMortgageLoan	--	handlingCapital interestNextRevision saleCostProperty	cancelationCommission
FutureMortgageLoan	--	buyCostProperty quotaAfterRevision delayInterestRate homeInsurance initalQuota initialPeriod interesDelay lifeInsurance	openingCommission subrotationCommission

		mortgageTaxation periodicityQuota revisionTerm term	
Liability	--	--	--
ProductRateApplication	--	interestRateValue	--
ProductRateApplicationMixed	--	termRateFixed	--
ProductRateApplicationFixed	--	--	--
ProductRateApplicationVariable	--	--	--
Quota	--	endingDate startingDate	--

Binary relation table

Relation name	Source concept	Source cardinality (Max)	Target concept
cancelationCommission	CurrentMortgageLoan	1	Quota
openingCommission	FutureMortgageLoan	1	Quota
subrogationCommission	FutureMortgageLoan	1	Quota
currencyProduct	Product	1	Currency
interestRateType	Product	n	ProductRateApplication
payments	SavingAccount	n	Payment

Instance attribute table

Instance attribute name	Concept name	Value type	Cardinality
handlingCapital	CurrentMortgageLoan	String	(0, n)
interestNextRevision	CurrentMortgageLoan	String	(0, n)
saleCostProperty	CurrentMortgageLoan	String	(0, n)
buyCostProperty	FutureMortgageLoan	String	(0, n)
quotaAfterRevision	FutureMortgageLoan	String	(0, n)
delayInterestRate	FutureMortgageLoan	String	(0, n)
homeInsurance	FutureMortgageLoan	String	(0, n)
initalQuota	FutureMortgageLoan	String	(0, n)
initialPeriod	FutureMortgageLoan	String	(0, n)
interesDelay	FutureMortgageLoan	String	(0, n)
lifeInsurance	FutureMortgageLoan	String	(0, n)
mortgageTaxation	FutureMortgageLoan	String	(0, n)
periodicityQuota	FutureMortgageLoan	String	(0, n)

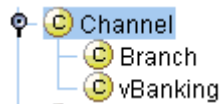
revisionTerm	FutureMortgageLoan	String	(0, n)
term	FutureMortgageLoan	String	(0, n)
interestRateValue	Product, ProductRateApplication	String	(0, n)
revisionTermNext	MortgageLoan	String	(0, n)
capital	MortgageLoan	String	(0, n)
APR	Product	String	(0, 1)
expirationDate	Product	String	(0, n)
signalDateContract	Product	String	(0, n)
termRateFixed	ProductRateAplicationMixed	String	(0, 1)
endingDate	Quota	String	(0, n)
startingDate	Quota	String	(0, n)

4.4 Ontology conceptualisation: Channel ontology

Terms glossary

Name	Description	Type
Channel	Communication means used in the relationship between the bank and its customers, including branches, phone, Internet, virtual banking, etc.	Concept
Branch	Physical bank office	Concept
vBanking	Virtual Banking. Banking without human intervention	Concept

Concept classification tree



Concept dictionary

Concept name	Class attributes	Instance attributes	Relations
Channel	--	--	--
Branch	--	--	--
vBanking	--	--	--

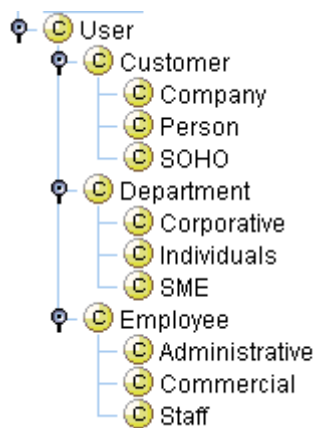
4.5 Ontology conceptualisation: User ontology

Terms glossary

Name	Description	Type
User	Any user of the system: customers, departments, and employees	Concept
Customer	Bank client, who usually has a contractual relationship with the bank	Concept
Company	A number of people grouped together as a business enterprise. Types of companies include public limited companies, partnerships, joint ventures	Concept

	and proprietorships, and branches of foreign companies	
Person	Bank client that represents a single person (physical or juridical)	Concept
SOHO	Small Office, Home Office. It usually refers to professionals who work in their own offices	Concept
Department	Internal area of the bank that performs a specific function	Concept
Corporative	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.	Concept
Individual	Department that deals with physical persons	Concept
SME	Small or Medium Enterprise	Concept
Employee	Employee	Concept
Administrative	Employee with administrative functions	Concept
Commercial	Employee with commercial functions	Concept
Staff	Central Services of a company, such as Human Resources, Management, Innovation, etc.	Concept
titularity	Product that a customer holds	Relation
login	User login	Instance Attribute
password	User password	Instance Attribute
name	Product or customer name	Instance Attribute
CIF	Código de Identificación Fiscal	Instance Attribute
NIF	Número de Identificación Fiscal	Instance Attribute

Concept classification tree



Concept dictionary

Concept name	Class attributes	Instance attributes	Relations
User	--	login password	--
Customer	--	name	titularity

Company	--	CIF	--
Person	--	NIF	--
SOHO	--	NIF	--
Department	--	--	--
Corporative	--	--	--
Individuals	--	--	--
SME	--	--	--
Employee	--	--	--
Administrative	--	--	--
Commercial	--	--	--
Staff	--	--	--

Binary relation table

Relation name	Source concept	Source cardinality (max)	Target concept
titularity	Customer	N	Product

Instance attribute table

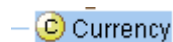
Instance attribute name	Concept name	Value type	Cardinality
login	User	String	1..1
password	User	String	1..1
name	Customer	String	1..1
NIF	Company	String	1..1
CIF	Person, SOHO	String	1..1

4.6 Ontology conceptualisation: Currency ontology

Terms glossary

Name	Description	Type
Currency	Money in circulation	Concept

Concept classification tree



Concept dictionary

Concept name	Class attributes	Instance attributes	Relations
Currency	--	--	--

5 IMPLEMENTATION

The financial ontology has been implemented in OWL, since it has been modelled with the Protégé tool³ and the OWL plug-in [6]. It will be transformed into WSML for the implementation to be used in the case study deployment phase. The source code is available in the DIP BSCW server⁴.

6 REFERENCES

- [1] Blázquez M, Fernández-López M, García-Pinar JM, Gómez-Pérez A (1998) *Building Ontologies at the Knowledge Level using the Ontology Design Environment*. In: Gaines BR, Musen MA (eds) 11th International Workshop on Knowledge Acquisition, Modeling and Management (KAW'98). Banff, Canada, SHARE4:1–15
- [2] D10.1 Analysis Report on eBanking Business Needs. DIP deliverable. Mónica Martínez Montes, José Luís Bas, Sergio Bellido, José Manuel López, Silvestre Losada, Richard Benjamins (2004).
- [3] Fernández-López M, Gómez-Pérez A, Juristo N (1997) *Methontology: From Ontological Art Towards Ontological Engineering*. Spring Symposium on Ontological Engineering of AAAI. Stanford University, California, pp 33–40
- [4] Fernández-López M, Gómez-Pérez A, Pazos A, Pazos J (1999) *Building a Chemical Ontology Using Methontology and the Ontology Design Environment*. IEEE Intelligent Systems & their applications 4(1):37–46
- [5] Gómez-Pérez A, Juristo N, Pazos J (1995) *Evaluation and assessment of knowledge sharing technology*. In: Mars N (ed) *Towards Very Large Knowledge Bases: Knowledge Building and Knowledge Sharing (KBKS'95)*. University of Twente, Enschede, The Netherlands. IOS Press, Amsterdam, The Netherlands, pp 289–296
- [6] Knublauch H (2003) *Editing Semantic Web Content with Protégé: the OWL Plugin*. 6th Protégé workshop. Manchester, United Kingdom
- [7] Noy NF, Ferguson RW, Musen MA (2000). *The knowledge model of Protege-2000: Combining interoperability and flexibility*. In: Dieng R, Corby O (eds) 12th International Conference in Knowledge Engineering and Knowledge Management (EKAW'00). Juan-Les-Pins, France. Springer-Verlag, LNAI 1937, Berlin, Germany.
- [8] Rojas MD (1998) *Ontologías de iones monoatómicos en variables físicos del medio ambiente*. Proyecto Fin de Carrera. Facultad de Informática, Universidad Politécnica de Madrid, Madrid, Spain
- [9] Staab S, Schnurr HP, Studer R, Sure Y (2001) Knowledge Processes and Ontologies. IEEE Intelligent Systems 16(1):26-34

³ <http://protege.stanford.edu/>

⁴ <https://bscw.dip.deri.ie/>