



REGNET

Cultural Heritage in REGIONal NETworks

Deliverable D8

The REGNET – System Version 2

Project acronym	REGNET	Contract nr.	IST-2000-26336
Type and Number	D8 The REGNET-System, Version 2		
Work package	WP3: : Validation of the REGNET demonstrator		
Task	T3.3: Implementation of the version 2 of the REGNET system		
Date of delivery	2002-11-28		
Code name	RN_D8v01	Version 01	draft <input type="checkbox"/> final <input checked="" type="checkbox"/>
Objective	Report		
Distribution Type	Restricted		
Authors (Partner)	VALT		
Abstract	This deliverable contains the results of Work Package 3 related to task 3.3		
Keywords List	REGNET System, Implementation		
Version Log	2002-11-28: First Version		



Table of Contents

Executive summary	5
1 Introduction	7
1.1 Iterations.....	7
2 Internationalisation	8
2.1 Portal	8
2.1.1 Internationalisation form.....	9
2.1.2 Form management.....	10
2.2 Data Entry.....	10
2.2.1 Internationalisation form.....	10
2.2.2 Form management.....	10
2.3 Search and retrieval	10
2.3.1 Internationalisation form.....	10
2.3.2 Form management.....	10
2.4 Product Catalogue Management.....	10
2.4.1 Internationalisation form.....	10
2.4.2 Form management.....	12
2.5 EShop.....	13
2.5.1 Internationalisation form.....	13
2.5.2 Form management.....	14
2.6 Auction.....	15
2.6.1 Internationalisation form.....	15
2.6.2 Form management.....	16
2.7 eProcurement.....	17
2.7.1 Internationalisation form.....	17
2.7.2 Form management.....	17
2.8 Knowledge base access.....	21
2.9 Topic Map Viewer.....	21
2.9.1 Internationalisation form.....	23
2.9.2 Form management.....	23
2.10 Publisher.....	23
2.10.1 Internationalisation form.....	23
2.10.2 Form management.....	27
3 Front End Integration	30
4 Topic Map.....	34
4.1.1 Topic Map format	34
4.1.2 Architecture of the authoring tool	34
4.1.3 User interface of the authoring tool.....	34
5 Distributed REGNET.....	37
5.1 Multi-Portal	37
5.1.1 Creation of a new Account.....	38
5.1.2 Editing of data related to an existent Account.....	39
5.1.3 Login.....	39
5.1.4 Deletion of an existent Account	41
5.2 REGNET connector.....	41
5.3 EbXML at work	41
5.3.1 Quick ebXML description	41
5.3.2 The need for ebXML	41



5.3.3	ebXML Architecture.....	42
5.3.4	Example	44
5.3.5	ebXML messaging	44
5.3.6	The Business Process Specifications Schema.....	46
5.3.7	CPP Definition	51
5.3.8	The Registry/Repository.....	57
5.4	<i>Regnet connector</i>	58
5.4.1	Implementing an ebXML messaging service	58
5.4.2	EbXML message example	59
5.4.3	Modelling business process	60
5.4.4	Installing the ebXML registry/repository.....	66
5.5	<i>EbXML Glossary</i>	67
6	Epublishing	78
6.1	<i>Additional functionalities in the 2nd version of the Electronic Publisher</i>	78
6.1.1	Internationalisation of User Interface	78
6.1.2	Internationalisation of the Publication	78
6.1.3	Timeline Storyboard	78
6.1.4	Integration of the Ontology.....	78
6.2	<i>Publishing workflow</i>	78
6.3	<i>Refinement of existing components</i>	80
6.3.1	Location storyboard.....	80
6.3.2	Integration of output formats	80
7	Auction	81
7.1	<i>Overview</i>	81
7.2	<i>Structure and content of the total system</i>	81
7.2.1	Functionalities	81
7.2.2	Technical features	81
8	Delivery	85
8.1	<i>Architecture</i>	85
8.2	<i>GLUE functionalities</i>	85
8.2.1	Publishing and Unpublishing.....	85
8.2.2	WSDL Generation	87
8.2.3	Invoking	87
8.2.4	Invoking using HTTP GET/POST	88
8.2.5	SOAP interface of Delivery system.....	88
8.2.6	Configuration process	89
9	Index+ gateway	90
9.1	<i>Introduction</i>	90
9.2	<i>Index+ Gateway component</i>	91
9.3	<i>Index+ features</i>	92
10	Cooperative thesauri	95
10.1	<i>Functionalities</i>	95
10.2	<i>Structure</i>	95
10.3	<i>Thesaurus Doctype</i>	95
11	Annexe A: Installation manuel	96
11.1	<i>Auction</i>	96
11.2	<i>Delivery</i>	97



11.2.1	Configuration	97
11.2.2	Requirements	97
11.2.3	Installation procedure	97
11.3	<i>Index+ gateway [SPACE]</i>	97
11.3.1	Hardware & software requirements	97
11.3.2	Instructions for installations of entire environment	98
11.3.3	Instructions for configuring and running webservice	99
11.4	<i>Cooperative thesauri</i>	100
12	Annexe B: User manuel	104
12.1	<i>Auction</i>	104
12.1.1	GENERAL	104
12.1.2	USERS REGISTRATION	104
12.1.3	SEEING THE LOTS	105
12.1.4	PRODUCTS SEARCH	107
12.1.5	1.3.5 PROFILE CHANGE	108
12.1.6	ADD ITMES IN THE AUCTION SYSTEM	111
12.1.7	AUCTION PARTICIPATION	112
12.1.8	AUCTION SYSTEM ADMINISTRATION	114
12.1.9	MANAGE ITEMS	116
12.1.10	DELETE ITEMS	117
12.1.11	MANAGE USERS	118
12.1.12	DELETE USERS	119
12.1.13	ADD NEW CATEGORY	120
12.1.14	EXIT	121
12.2	<i>Delivery</i>	122
12.3	<i>Index+ gateway</i>	123
12.4	<i>Cooperative thesauri</i>	135
12.4.1	Navigation	136
12.4.2	Adding/updating/deleting entries	137
12.5	<i>“Topic Maps Generator” tool</i>	137
13	Annexe C: Internationalisation forms	143
13.1	<i>eProcurement</i>	143
13.1.1	Bulgarian	143
13.1.2	Catalan	145
13.1.3	German	147
13.1.4	Dutch	149
13.1.5	English	152
13.1.6	Spanish	152
13.1.7	French	154
13.1.8	Greek	156
13.1.9	Italian	158
13.1.10	Russian	160
13.1.11	Swedish	162
	List of Figures	166



Executive summary

This document includes the activities from Work Package 3 that have been carried out within Work Area B (Platform Engineering) of the REGNET project. The main emphasis of the specific document is focused on providing a complete overview of the second version of the REGNET System and the functionalities that are supported by it. This document complete deliverable D5 that describes the REGNET System version 1 and detail new functionalities developed during this period.

The first part is an introduction which detail iterations splitting according to the construction phase of the Unified Process adopted for this project.

The second part details internationalisation process adopted by each software module. The following languages has been taken into account: Bulgarian, Catalan, Deutsch, Dutch, English, Spanish, French, Greek, Italian, Russian and Swedish.

The third part gives an overview of the new REGNET mockup developed in order to fix ergonomoy of the system. The look and feel will be adopted by all modules.

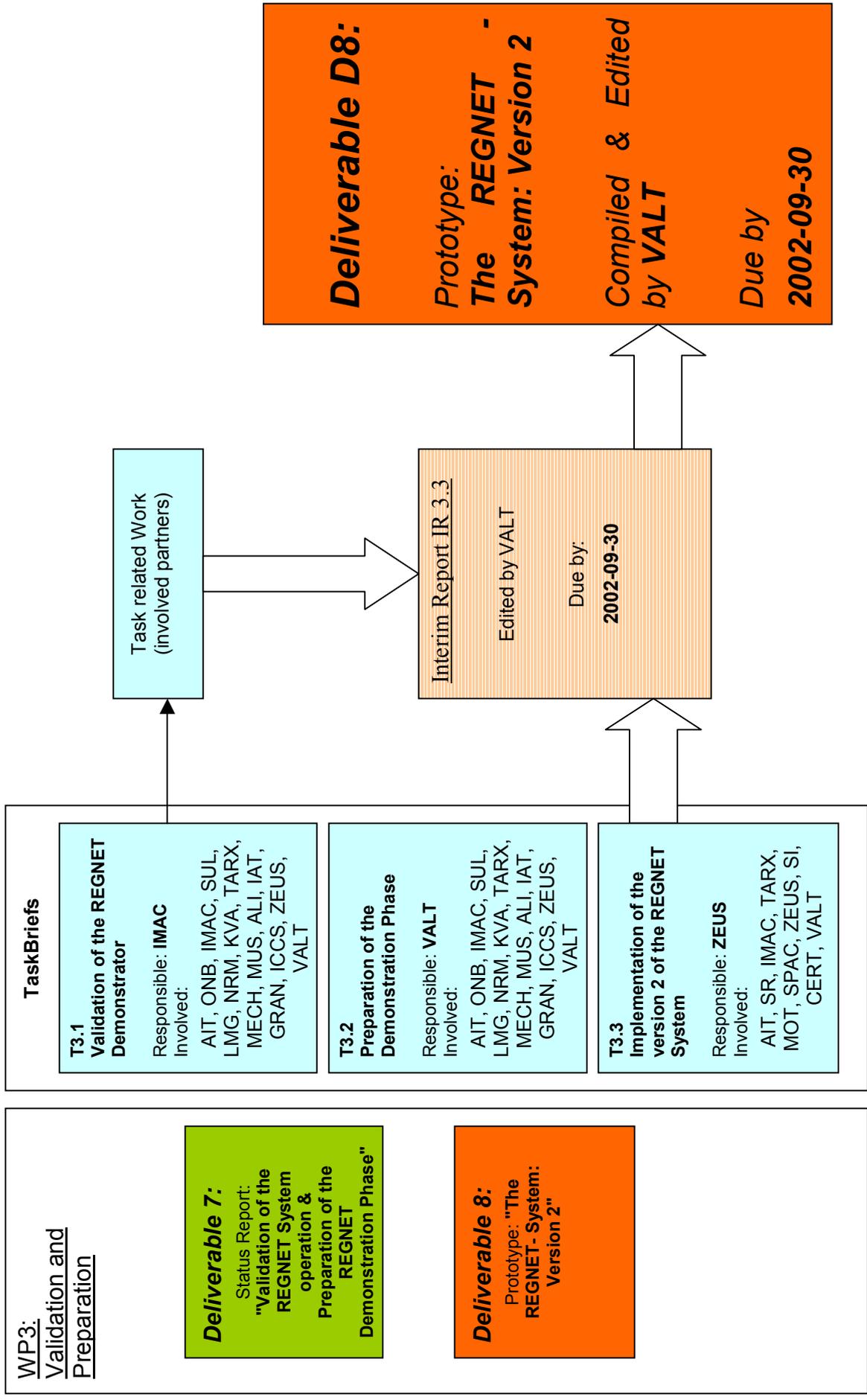
The fourth part is dedicated to discribe enhancement done to Topic Map authoring tool.

The fifth part describes work done in order to implement the Regnet connector based on the ebXML framework. It gives a technical overview of the framework and describe the innovative work done in order to implement it.

Next parts describe ePublishing, auction, delivery, index+ gateway and cooperative thesauri modules.

Annexes are:

- A: Installation manuals. This part contain installation manual for new Regnet components (auction, delivery, Index+ gateway and cooperative thesauri). It describes necessary software and installation procedures.
- B: User manuals. These user manuals are dedicated to the new Regnet component. Users who want to learn how to use the software can use them.
- C: Some internationalisation forms.





1 Introduction

This document contains artefacts from the task 3.3 "Implementation of the version 2 of the REGNET system". It takes the same plan than IR2.2 (Implementation of the version 1 of the REGNET system). As far as process used for Regnet project is based on Unified Process, this document contains iterations and artifacts description.

1.1 Iterations

This task has been splitted into 2 iterations of 2,5 months. According to UP they are relevant to **construction** phase. The inception and elaboration phases has been finished in the previous work-packages (WP1, WP2). This task is a direct continuation of task 2.2 of WP2: system implementation version 1.

First iteration: Version 1 refinement

The aim of this iteration is to refine the first version of the REGNET system by implementing new Use-Cases defined during the analysis task.

This refinement deals mainly with the following aspects:

- Internationalisation: each module must be internationalised according to the specified set of languages: English, Dutch, German, Greek, Swedish, Bulgarian, Russian, Italian, Spanish and French.
- Front end integration: this work is relative the following tasks:
 - Look and Feel integration of the different modules.
 - Providing of content.
- Deep integration of Topic Map.

Second iteration: Extension

This iteration is dedicated to the development of new functionalities:

- Distributed version of the REGNET system:
 - Implementing mechanism in order to allow multi-portal synchronisation through Ontology component.
 - Implementing Regnet Connector based on ebXML framework allowing asynchronous exchange of information between different REGNET systems.
- Choice and integration of 3rd party Epublishing component.
- Ebusiness extension:
 - Development of the Auction component.
 - Development of the delivery component.
- Wap Access development.
- Development of a gateway in order to access OPEN HERITAGE / Index+ database.
- Development of a cooperative Thesauri.

2 Internationalisation

Internationalisation is based on forms defined by each partner in charge of a REGNET's module. These forms contain word to be translated in the different languages. Each module is free to manage these forms according to the more suitable algorithm. This part gives these forms and describes implementation management.

2.1 Portal

The Portal architecture separates the framework of the Portal itself and a set of "portlets" which are customisable by the user. The framework contains the user interfaces for all the common functionalities of the Portal, like navigation, user account management, language selection, and customisation. The core layer of the software tool that implements the Portal, that is Jakarta Jetspeed, manages these functionalities. Jetspeed on the other hand relies on Jakarta Turbine, which is a Servlet-based framework that allows experienced Java developers to quickly build secure web applications.

The portlets can be as slight as an HTTP forward or a more complex structure of pages, depending on how much of the application logic resides on the Portal and how much is located elsewhere. The portlets are the user doors to access the REGNET contents and are completely customisable for what regards their presence, position, appearance, etc. For what concerns the internationalisation, the developer can use the same APIs as for the framework.

Turbine supplies a default mechanism to intercept the language preferences from the user browser and to match these preferred languages with those managed by the application. The default language and other available languages are specified in property files and, for each language, there must be a resource bundle file containing a list of key-value couples. Each key is unambiguous for all the languages, and it is invoked in the GUI code to be substituted at run-time by the localized contents.

Figure 1 below shows that the algorithm of Turbine is based on HTTP Request analysis, and that each request is treated without memory of the previous one.

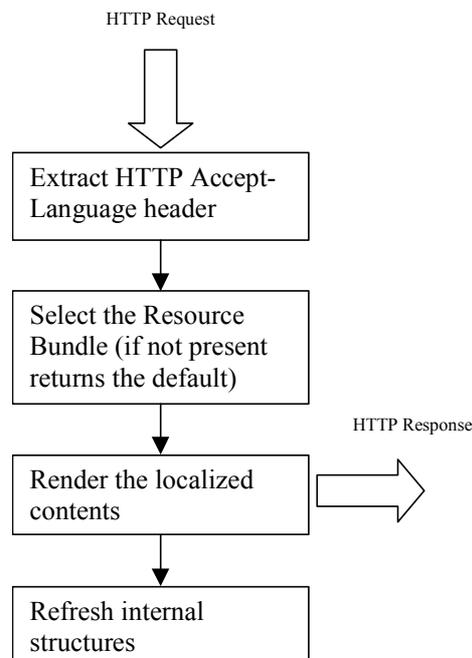


Figure 1: Turbine's extraction of language information

The requirement to supply a GUI mechanism to let the user to select the language for content rendering makes necessary to store this choice in the session environment. The result is represented in Figure 2. More in detail, it was created an "action" (`setLocale()`) to trigger the change of the language, implemented by the class `org.apache.turbine.modules.actions.SetLocale.class`. It is invoked from `*.vm` (templates for GUI pages) files with `$link.setAction("SetLocale")`.

The selected language is stored in the Session Environment and passed via the URL with the parameter defined in `LocalizationHelper.USER_LOCALE`. Storing into session is performed by `org.apache.jetspeed.modules.pages.JetspeedVelocityPage.doBuildBeforeAction()`.

The two classes `Org.apache.turbine.services.localization.LocalizationTool` and `Org.apache.turbine.services.localization.LocalizationHelper` manage the locale information in the session environment. `LocalizationTool` was modified to get this information from parameters in the URL, or from the session, and to store this information in the session.

The property file `TurbineResources.properties` (`<jetspeed_home>\webapp\WEB-INF\conf`) must contain the property `locale.language.supported` with all the languages (and countries) supported for the internationalization. For example: `locale.language.supported=en_UK_US-de-it-es-nl-ru-bg-ca-sv`. This property is loaded by the class `LocalizationHelper`. If the language required is not included in the property above, the default is used (it is English, set again in `TurbineResources.properties`).

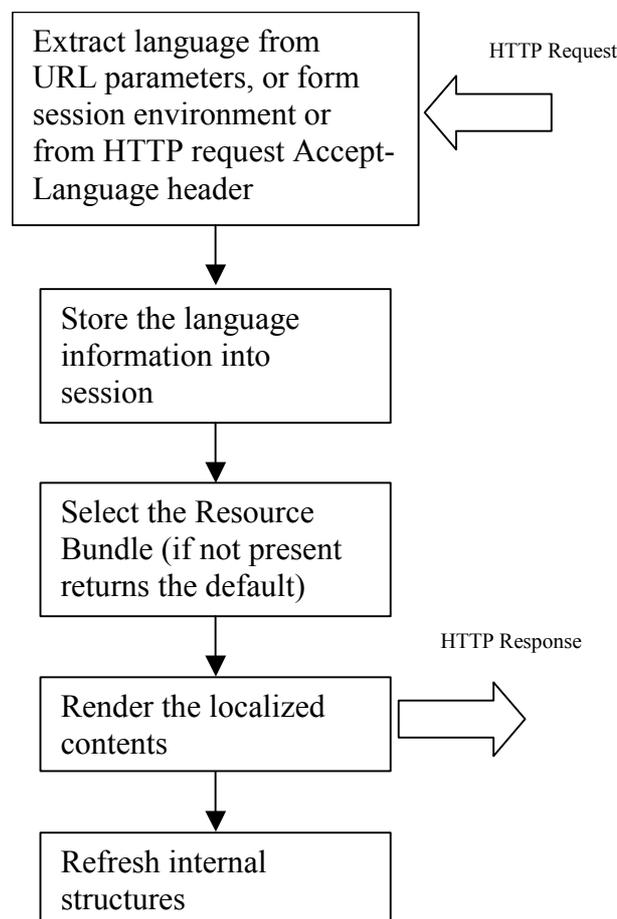


Figure 2: REGNET customisation to intercept language preferences

2.1.1 Internationalisation form

Language bundles are stored in property files in `Localization.jar` library, representing an extension of the package `org.regnet.portal.localization`.



The file name is in the form *RegnetLocalization_xx.properties* where “xx” represent the language and country codes according to ISO 639 and ISO 3166 standards. All the keys not found in a language specific file, will be sought in *RegnetLocalization.properties*.

2.1.2 Form management

To add a new key, simply add a couple [key=value] in each property file (included the generic one). The key is available in the template by the code *\$!10n.<key>* (note that l10n stands for “localization”). To add a new language, it is necessary to add the new property file into *Localization.jar* library (with the same path as the others) and to update the *locale.language.supported* property in *TurbineResources.properties* according to the above mentioned standards.

2.2 Data Entry

The interface language is selected on session startup with a http-get parameter named “lang” which is set to the requested ISO-3166 country code. This value is stored in the session. The forms itself contain only English terms. If the user specifies his language the corresponding translations for each English term are lookup from the internationalisation section of metadata file and stored in a dictionary of the session.

2.2.1 Internationalisation form

Internationalisation terms are stored as a xml in the metadata file. This file contains all translations of the terms used in the application as '`<term><translation lang="en">...</translation><translation lang=".."></translation>...</term>`'.

2.2.2 Form management

New translations are added via import translation function of the application administration by pasting a simple table (columns separated by tabulator, rows separated by newline). The first row must contain the ISO country code. The first column must always be English language.

2.3 Search and retrieval

Search and retrieval shares the same resources with data entry.

2.3.1 Internationalisation form

See 3.2.1

2.3.2 Form management

See 3.2.1

2.4 Product Catalogue Management

2.4.1 Internationalisation form

2.4.1.1 Index form

Insert item	Insert warehouse	Update, delete or Look Items	Insert service	Update, delete or Look services
For update, delete or look	Select catalogue	Select category	Let's go	For update, delete or look



items, please select catalogue and category				services, please select catalogue and category
---	--	--	--	--

2.4.1.2 Insert item form

It contains all the specific information for a user in order to insert a new item

Please, select catalogue	Submit	Category_id	Name	Description
UPCCode	Price	Price_type	Wholesale	Retail
Currency_id	Quantity	Manufacturer	Weight	W-measure
Volume	V-measure	Height	Length	Width
D-measure	Export	YES	NO	Insert
The good successfully inserted!	You can add its picture	Here!	Picture	Browse
Add picture	Reset	No file to upload	File copied	

2.4.1.3 Insert warehouse form

Insert new warehouse	Warehouse properties	User ID	Catalogue to insert to	Store area
Address	Street	City	State	Province
Post code	Country	Submit	The warehouse successfully inserted	

2.4.1.4 Insert service form

Please select a catalogue	Submit	Name	Description	Price
Currency_id	Insert	The good successfully inserted		

2.4.1.5 Update, delete or look form

This form contains all the specific information for a user in order to update, delete or look to items and services

For update, delete or look items, please select catalogue and category	Select catalogue	Select category	Let's go	Please, select an action
Action	Update	Delete	Look	Please, select a good
Name	Submit	Item_id	Category_id	Name



UPCCode	Price	Price_type	Retail	Wholesale
Currency_id	Manufacturer	Weight	W-measure	Volume
V-measure	Height	Length	Width	D-measure
Export	YES	NO	Description	Warehouse_id
Store_area_id	Address_id	The good successfully updated	You can change its picture	Here!
Picture	Browse	Add picture	Reset	For update, delete or look services, please select catalogue and category
The good successfully deleted	File already exists	Sorry there are no goods in this category		

2.4.1.6 B2B form

It contains information for the e-business component

Search services for	Search for items	ProdName	Category	Upccode
Quantity	Manufacturer	Store_Area	Price_Type	Company ID
Company Name	Item_id	Category_id	Name	Price
Currency_id	Order	Submit order	Contract	Item
Supplier	ID_type	Street	City	State
Province	Postcode	Country	Phone	Fax
e-mail	URL	Contract sum	Warehouse Address	Total
Sum	All	Retail	Wholesale	

2.4.2 Form management

2.4.2.1 PCM Form management

The product catalogue management system involves the following logical framework for working with it. The first step for a new user of PCM is to register himself in the system. Then he/she has to apply to the leaders of the project with a request for the creation of a new catalogue. The creation of the new catalogue needs an installation of the software on the remote (users) server and that is why it should be done. The new user should provide the information about the location (IP address) of the new catalogue and a URL (place of location in the document's tree on the user's web server).

After the installation of the new remote catalogue, the administrator of PCM will add a new catalogue to the database (table catalogue). From this moment this catalogue is accessible for the new user by his ID and password. He has the ability to insert new items or services to the catalogue, to edit the information of already existing items and services and to delete them from the catalogues. Moreover, he will have the ability to add information about the new warehouses to the PCM.

2.4.2.2 B2B Form management



The B2B system gives the ability to every person or organization to search within the system. The system's users have only a limited set of information about the products and services included in the system, an issues which restricts them from fulfilling any kind of business agreements.

After registering into the system, the user has the ability to receive a complete set of information about the products and services, the suppliers involved, any shipment conditions included, the location of the warehouses additionally giving them the opportunity to calculate the prices and the sum of each agreement.

The process of searching for a particular product can be considered very simple. The only required field to be completed is Product name, and then the user has to choose the category of the desired product, the Quantity (default value is one), store and Type of price (retail or wholesale). Additionally, (if it is known by the user) he/she can insert UPC (Universal Product Code), the name of the chosen manufacturer of the product and also the name & the Company ID of the supplier.

2.5 EShop

2.5.1 Internationalisation form

2.5.1.1 Index page form

In the index page form we have the following words:

Basket	Wish List	New Search	Order History	Item
Category	Price	Lower	Higher	Search
No	Name	Producer	Price	Photo
Server	Action	Details of the product	Add to basket	All

2.5.1.2 Basket form

The basket form contains some words from the index form and some new words that we present them in the following table:"

Basket	Wish List	New Search	Order History	Item
Category	Price	Lower	Higher	Search
All	No	Check	Name	Producer
Price	Quantity	Total	Buy	Save in wish list
Clear	Delete	Your basket is empty	Click here for new search	Items from your basket were successfully saved in a wish list!
Click here for new search				

2.5.1.3 Buying form

In the buying form we will have the following words:

Hello,	Is your home	YES	NO	If no, please fill
--------	--------------	-----	----	--------------------



	address the same as your delivery address?			in this form:
Address	Country	Your Country	Phone	City
Zip	Post code	E-mail	Submit	Reset
Please select the way of payment	Credit card	Check	Cash	Submit
Thank for purchase				

2.5.1.4 Wish list Form

It contains words both from basket form and index form

Basket	Wish List	New Search	Order History	Item
Category	Price	Lower	Higher	Search
All	No	Check	Name	Producer
Price	Quantity	Total	Buy	Clear
Delete	Your wish list is empty			

2.5.1.5 New Search form

Basket	Wish List	New Search	Order History	Item
Category	Price	Lower	Higher	Search
All				

2.5.1.6 Order History Form

Basket	Wish List	New Search	Order History	Item
Category	Price	Lower	Higher	Search
All	You have the following orders in history:	No	Date	Your order at the xx:xx o'clock
Your order had the following items:	No	Name	Producer	Quantity
Price	Sum	total		

2.5.2 Form management

So in the e-shop component we have six forms, which are the index, basket, buying, wish list, new search and order history form. The more suitable algorithm that describes the internationalization forms is giving below:



The first form is the index and presents the first page (default) of the e-shop component. From this page the user can use all the main functionalities of the e-shop such as the basket, the wish list, the order history and the searching functionality. Firstly a user has to search for new items and then he/she can add new item(s) in the basket of the e-shop component.. Then the user can buy the items he/she wants or he/she can move them into the wish list in order to buying them later, when he/she prefers. When a user has made any purchase he/she has the ability to see his/her order history. The searching functionality can be used from users at any time because it is provided in every page of the e-shop component. So the index, the wish list, the basket and the new search forms contain some same words that have to do with the searching functionality. Also the buying form in the basket form is the same with the buying form in the wish list, if users want to make any purchase from the wish list.

2.6 Auction

2.6.1 Internationalisation form

2.6.1.1 Index form

The first page of the auction system it contains the following items (following functionalities)

Current lots	Future auction	Search	Register new item	Personal activity
--------------	----------------	--------	-------------------	-------------------

2.6.1.2 Current lots form

This form has all the information of the current lots

Name	Description	Producer	Picture	Start price
Started from	Bargain?	Page	Go	Lot
Time of finishing	List of bids	Your price	From...on	Sorry your price was approached before
Ok!				

2.6.1.3 Future auctions form

This form has all the specific information of the future auctions that have not started yet.

Name	Description	Producer	Picture	Start price
Started from				

2.6.1.4 Searching form

Fill in the searching form	Name	AND	OR	Description
Category	Open time	Year	Month	Day
Hour	Before	After	Just in	Search
Reset	Close time	Searching results	This lot is already closed	There is no items on your requirements

2.6.1.5 Register new item form

It contains all the necessary data that a users has to add in order to register a new item into the auction system

Add item for	Name	Category	All	Description
--------------	------	----------	-----	-------------



auction				
Producer	Start price	Reserve price	Time of start	Time of end
Year	Month	Day	Hour	Picture
Browse	Add item	Reset	File copied	File already exists

2.6.1.6 Personal activity form

This form contains all the information about the user's personal activity. That means that has information about his/her items and also the history of his/her bids

Your items	Your bids	Name	Status	Producer
Picture	Start price	Date of insert	Bids	Yes
No	Look	Bids on item..	Start price	Reserve price
...was inserted	Start time	End time	Sold?	Price
Time	Your current bids	History of your bids	You bid with item	Started from
Your bids on item	Time	Price		

2.6.1.7 Administration form

The administration form contains the following words that must be translated. The functionalities of the specific component are the managing of items and users and also the adding of new categories.

Hello administrator	Login	Password	Submit	Choose action
Manage items	Manage users	Delete items	Delete users	New category
Exit	List of items	Back	ID name	Description
Producer	Start price	Reserve price	Inserted	Started from
End time	Result	Allow?	Yes	No
Submit	List of users	Back	ID	Login
Password	Name	Last Name	Address	Phone
Email	URL	Sale?	Delete?	List of existed categories
Add new category for items	Add	Clean		

2.6.2 Form management

Auction system was build with a purpose to provide client to bid for the different products.

So a user can see the current lots page or can use the searching functionality in order to choose the specific items he/she prefers. The searching system provides ability for multi criteria search. Criteria might be such as name of the product, its description, category, time of start and time of finish auction with an ability to search before, after or just in the appointed date.



Also a user has the ability of adding new items in the auction system. Additionally users have the ability to see and make changes in their own profile data, and can see their history bids in the different auctions.

Also auction system has its own administrative system. This system allows administrator manage the items, users and add the different categories to the database. It doesn't allow him to remove the existed categories, because it might be a lot of items in the auction now and it might destroy the history records.

So firstly when a user login in the auction system, then he/she has the ability to choose between the following functionalities: can see the current lots, can search for items, can register a new item or can see his/her personal activities. After the searching functionality, users can bid to the items they prefer and then the user's personal activity will be updated. Also users can register new items in the auction system. After the registration user's items in the personal activity will be updated.

2.7 eProcurement

The internationalization process is based on the internationalization (or "i18n" which is a common abbreviation for internationalization, because there are 18 letters between the first "i" and the last "n") tag extension library from Jakarta Taglibs. The goal of these tag libraries is to provide an open-source repository for JSP custom taglibs and Web Publishing tool extensions which support JSP custom taglibs.

Tag libraries are composed of a set of custom tags. These custom tags help separate presentation from implementation. What this means is that web designers can change the layout without worrying about modifying the underlying logic. Custom tags also help developers avoid embedding scripting code within the JSP page as well as encourage reuse and ease maintainability.

Custom tags are not just bean tags. Custom tags can modify the content within the tag body and have access to the application context. Some of the ways they can be used include dynamically generating page content and implementing flow of control. They can interact with each other including being nested.

JSP custom tags were introduced in JSP version 1.1. New features were added in JSP version 1.2. Custom tag libraries written for JSP 1.1 are backward compatible and can be used in a servlet container which supports JSP 1.2. Tag libraries are organized based on what version of the JSP specification they were written for.

2.7.1 Internationalisation form

Forms are given into annex.

2.7.2 Form management

2.7.2.1 The internationalization tag library

The i18n custom tag library contains tags that help manage the complexity of creating internationalized web applications. These tags provide similar (though not identical) functionality to the internationalization available in the struts framework, but do not require adopting the entire struts framework.

2.7.2.2 Requirements

This custom tag library requires no software other than a servlet container that supports the JavaServer Pages Specification, version 1.1 or higher.

2.7.2.3 Configuration

Follow these steps to configure your web application with this tag library:

Copy the tag library descriptor file to the /WEB-INF subdirectory of your web application.



Copy the tag library JAR file to the /WEB-INF/lib subdirectory of your web application.

Add a <taglib> element to your web application deployment descriptor in /WEB-INF/web.xml like this:

```
<taglib>
```

```
  <taglib-uri>http://jakarta.apache.org/taglibs/i18n-1.0</taglib-uri>
```

```
  <taglib-location>/WEB-INF/i18n.tld</taglib-location>
```

```
</taglib>
```

To use the tags from this library in your JSP pages, add the following directive at the top of each page:

```
<%@ taglib uri="http://jakarta.apache.org/taglibs/i18n-1.0" prefix="i18n" %>
```

where "i18n" is the tag name prefix you wish to use for tags from this library. You can change this value to any prefix you like.

2.7.2.4 Translation process

Translations are stored in properties files. A properties file is a simple text file. You can create and maintain a properties file with just about any text editor.

The name of this file begins with the base name of your ResourceBundle, then the suffix of the language is added e.g. _fr for French or _bg for Bugarian, and ends with the .properties suffix. Note if no language suffix is added this file is used as the default one.

Note that in the preceding file the comment lines begin with a pound sign (#). The other lines contain key-value pairs. The key is on the left side of the equal sign and the value is on the right. For instance, s2 is the key that corresponds to the value disk. The key is arbitrary. We could have called s2 something else, like msg5 or diskID.

Once defined, however, the key should not change because it is referenced in the source code. The values may be changed. In fact, when your localizers create new properties files to accommodate additional languages, they will translate the values into various languages

To support an additional Locale, your localizers will create a new properties file that contains the translated values. No changes to your source code are required, because your program references the keys, not the values.

For example, to add support for the German language, your localizers would translate the values in LabelsBundle.properties and place them in a file named LabelsBundle_de.properties. Notice that the name of this file, like that of the default file, begins with the base name LabelsBundle and ends with the .properties suffix. However, since this file is intended for a specific Locale, the base name is followed by the language code (de).

For the procurement one properties file are splitted by jsp page, that is to say that one properties file is associated to one jsp file.

For example this is the contents of the properties file for the French language related to the jsp page create_catalog.jsp

```
# This is the default create_catalog_fr.properties file
```

```
supplier = fournisseur
```

```
eshop = boutique en ligne ...
```

```
youmustcreateashowcase=Vous devez créer une vitrine
```

2.7.2.5 Tag Summary

The epurchase uses mainly the following tags.

Bundle	Define a resource bundle for use by other i18n tags.
Message	Displays internationalized text from a resource



	bundle.
--	---------

2.7.2.6 Tag Reference

Bundle

- Description :

Establishes the ResourceBundle to use for other i18n tags on the page. Also determines the most appropriate Locale to use based on browser settings if a locale is not provided.

- Restrictions :

This tag must be placed early in the page (before any HTML) in order for the output stream's content type to be set from the selected locale. If a locale (or localeRef) is not provided, the "best" locale will be determined from the browser settings and available locales for the requested bundle.

- Attributes :

Name	Description	Required
id	Script variable id for use with standard jsp:getProperty tag and as an attribute to other tags in this tag library.	No
baseName	Used along with the provided (or auto-sensed) locale to locate the desired ResourceBundle.	Yes
changeResponseLocale	Specifies whether or not the response locale should be changed to match the locale used by this tag.	No
scope	Specifies the scope (application, session, request, page) that this bundle will be made available to message and other tags.	No
locale	Allows the page developer to provide a direct object reference to the user's preferred locale.	No
localeRef	Allows the page developer to provide the name of an attribute whose value is the user's preferred locale. This attribute may exist in the page, request, session, or application scope.	No
debug	The debug flag. Logs debugging information of the bundle to the ServletContext.	No

- Examples

Define a bundle using browser preference to determine locale.

```
<i18n:bundle baseName="org.apache.taglibs.i18n.i18n-test"/>
```

Define a bundle using browser preference to determine locale, and declaring the scripting variable "bundle".



```
<i18n:bundle baseName="org.apache.taglibs.i18n.i18n-test"
  id="bundle"/>
```

Define a bundle using a scriptlet variable to specify the locale.

```
<i18n:bundle baseName="org.apache.taglibs.i18n.i18n-test"
  locale="<%= localeVar %>"/>
```

Define a bundle using a page, request, session, or application attribute to specify the locale

```
<i18n:bundle baseName="org.apache.taglibs.i18n.i18n-test" localeRef="userLocale"/>
```

Message

- Description

Format a message (using `java.text.MessageFormat`) from the value resulting from looking up the provided key within the default or specified resource bundle. Arguments to `MessageFormat` can be supplied as an object array or as subtags within the message tag body.

- Restrictions

Message tags require that a default bundle be provided (by placing a bundle tag prior to the message tag) or that a bundle or `bundleRef` attribute be supplied to the tag.

- Attributes

Name	Description	Required
id	Script variable id for use with standard <code>jsp:getProperty</code> tag and as an attribute to other tags in this tag library.	No
key	The key to use when retrieving the display message format from the <code>ResourceBundle</code> .	Yes
args	Provides an array of args for use with <code>java.text.MessageFormat</code> when formatting the display text. This is an alternative to using arg subtags.	No
bundle	An object reference to the <code>ResourceBundle</code> in which the key can be found.	No
bundleRef	The name of an attribute that contains a resource bundle. This attribute is usually defined/populated by <code>BundleTag</code> 's <code>id</code> attribute.	No
Debug	Set to "true" to log debugging messages.	No

- Examples

Display a plain message using the default (first defined) bundle.

```
<i18n:message key="column1.header"/>
```

Display a plain message using a specified bundle.

```
<i18n:bundle baseName="org.apache.taglibs.i18n.i18n-test"
```

```
id="bundle"/> <!-- the default -->
<i18n:bundle baseName="org.apache.taglibs.i18n.i18n-test2"
id="bundle2"/> <!-- the alternate -->
<i18n:message key="column1.header" bundle="<%= bundle2 %>" />
Display a message with arguments. In the example below, the English value for the key "datetxt" is
"Welcome, today is {0,date,short}."
<% Object msgArgs[] = {new Date()};%>
<i18n:message key="datetxt" args="<%= msgArgs %>"/>
```

2.8 Knowledge base access

The knowledge base of the REGNET Ontology system, is not directly connected with the end user, thus internationalisation is not necessary. However, the metadata that are stored in the knowledge base could be in multilingual format, if stored in this way by the other REGNET modules that are connected to the Ontology.

2.9 Topic Map Viewer

The Topic Map Viewer is part of the Ontology Node. The Topic Map Viewer was developed according to the "Topic Maps" standard (ISO13250). It supports most of the functionalities provided by the Topic Maps standard. Its purpose is to parse XTM files and present them in user-friendly way so that the users can navigate and effectively use the topic maps.

The tool was developed using Open source technologies (TCL programming language) and it requires the Apache Web Server on Unix to run.

Figure 3 shows the architecture of the Topic Map Viewer.

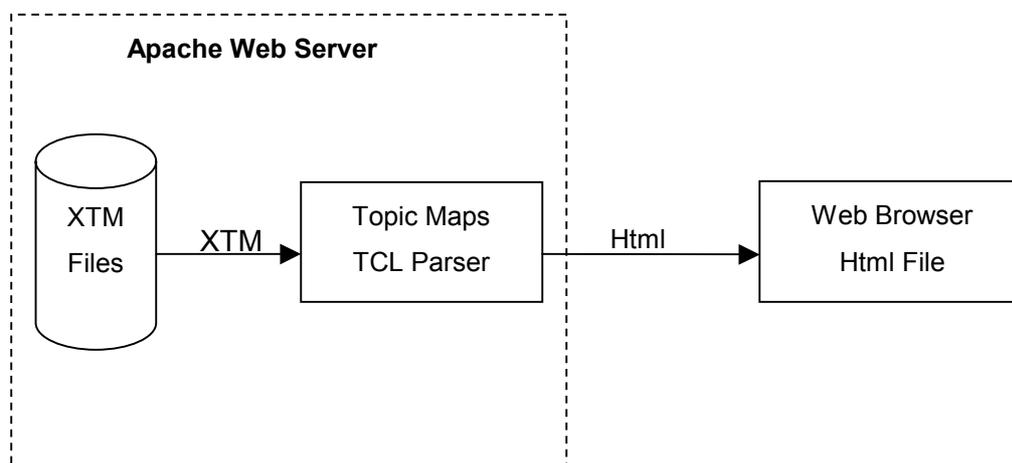


Figure 3: Topic Map Viewer architecture

When the user first operates the Topic Map Viewer, a list containing all available topic map files is being displayed. The user then can select the preferred topic map file to view and interact with (ie. Navigate). Once the file is being selected the TCL parser parses the xtm file and it dynamically transforms it to the appropriate HTML format. Parsing time is relative fast depending on the size of the selected topic map file. Then the user can see a list of all the topics contained in the topic map file and he/she can select a topic and then the information of the selected topic is displayed on a different frame (ie. Left frame at figure 2 displays all topics while the right frame displays information about a selected topic - in the above example the "Saint Angela" topic was selected). The Basename of the



topic can be displayed, the occurrences, associations, scopes etc. and generally all the data that a topic contains can be displayed and then the user can navigate via hyperlinks to related topics.

The following table displays the features supported by the developed topic map viewer.

Topic Map Viewer	Features Supported	Features not Supported
	XTM 1.0 ISO standard	Variants
	It allows the user to select a topic map file from a list in real time	topic map merging
	It resolves topic ids to basenames	
	- Occurrences - Associations - Basenames - Topic Ref - Subject Indicators - Resource Ref - Scopes - Instance of	
	It provides statistics about topics	
	parses the files relative fast	
	It allows to browse through the original xtm file	
	The html file is created dynamically	

The Topic Map Viewer was designed to be user-friendly and easy to use since the Topic Map standard by itself is a rather complicated standard and can easily confuse the users about its use. After researching on possible user interfaces we came up with an interface that is easy to use without leaving out any key features and functionalities of the standard. A screenshot of the developed interface can be seen at Figure 4.

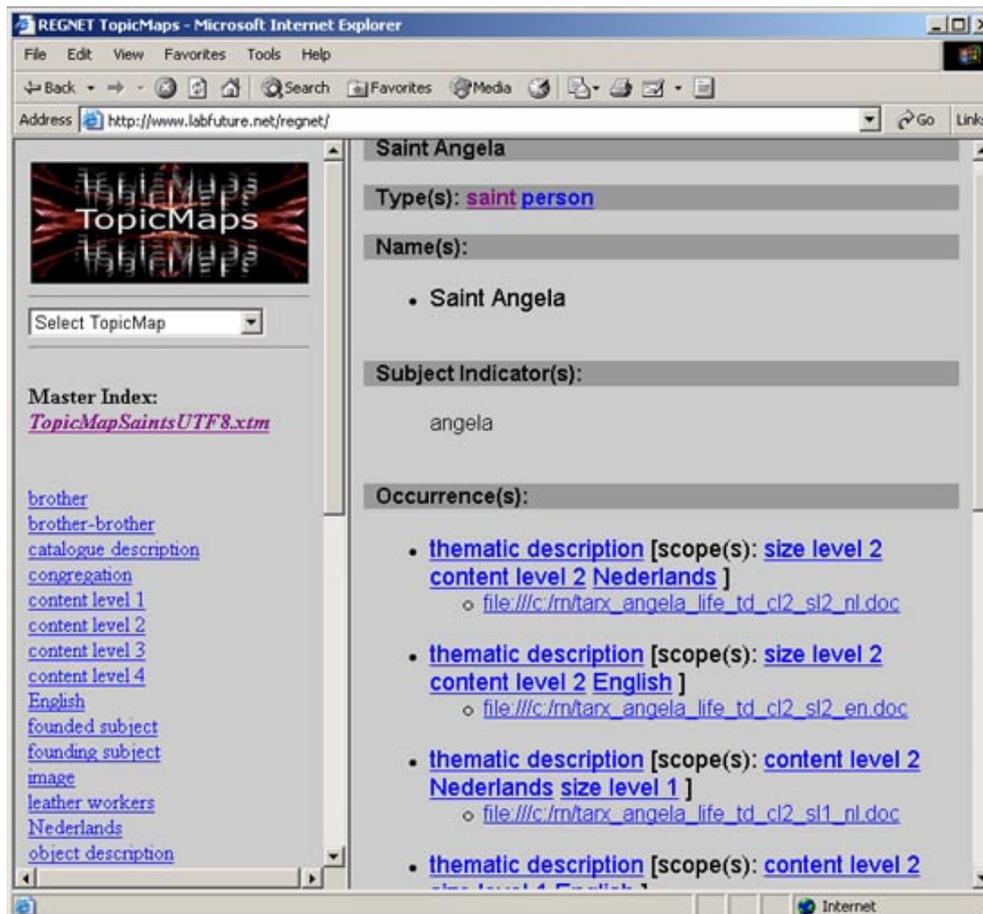


Figure 4: Topic Map Viewer interface

2.9.1 Internationalisation form

The Topic Map Viewer Interface is purely based on the Topic Maps standard and therefore it is not possible to translate the key concepts of the standard to different languages. By translating the standard there is great possibility of altering the meaning of topic map concepts, since most words are being used in a metaphorical way (ie. Scopes, Topic Ref etc.) and also there is a possibility to confuse the users by replacing the original words and phrases.

2.9.2 Form management

The authorized user can easily upload an xtm file to be displayed in the Topic Map viewer. This process is being handled by the Knowledge Base Access System. Uploading and deleting an xtm file is an easy and quick process that is achieved using forms online.

2.10 Publisher

2.10.1 Internationalisation form

The table below shows the translations of the English labels and text used in the Electronic Publishing Component:



English	German	Spanish	Bulgarian
Select a Publication to load	Wählen Sie die zu ladende Publikation	Seleccione la publicación a cargar	Избери публикация
Regnet Electronic Publishing Prototype	Regnet Electronic Publishing Prototyp	Prototipo de Publicación Electrónica de Regnet	REGNET Електронна публикация
or perform a query over the Regnet-Repositories and create a new Publication	oder führen Sie eine Suchanfrage über die Regnet-Datenbanken aus.	o realice una consulta al banco de datos de Regnet y cree una nueva publicación	или потърси в REGNET базата и създай нова публикация
Load Publication	Publikation laden	Cargar publicación	Зареди публикация
Perform Query	Suche ausführen	Realizar consulta	Изпълни търсенето
Continue	Weiter	Continuar	Продължи
Select a storyboard	Wählen Sie eine Storyboard:	Seleccionar patrón:	Избери тема
In this step of the publishing process a storyboard can be selected. Currently one storyboard is available. You can also proceed without choosing a storyboard. Storyboards can be used for grouping the data by certain criteria, which are called LATCH principles.	In diesem Schritt des Publikations Prozesses kann ein Storyboard ausgewählt werden. Storyboards bieten die Möglichkeit, die Suchergebnisse nach bestimmten Kriterien zu gruppieren. Diese Kriterien werden LATCH-Prinzipien genannt.	En este paso del proceso de publicación puede seleccionar un patrón. Actualmente sólo se encuentra disponible un patrón. También se puede continuar sin elegir ningún patrón. Los patrones pueden usarse para agrupar los datos bajo ciertos criterios, llamados principios de cierre (LATCH principles).	На тази стъпка от процеса на публикуване може да се избере тема. Текущо се обработва само една тема. Може да не изберете тема. Темата може да се използва за групиране на данни по определени критерии, наречени LATCH принципи
Submit	Abschicken	Enviar	Изпращане



Select a Layout:	Wählen Sie ein Layout aus:	Seleccionar estructura:	Избери оформление
In this step a layout can be selected. Currently three layouts are supported. Mention: not every layout is combinable with every fileformat.	In diesem Schritt kann ein Layout ausgewählt werden. Derzeit werden drei Layouts unterstützt. Nicht jedes Layout kann aber mit jedem Fileformat kombiniert werden.	En este paso puede seleccionar una estructura. Actualmente existen tres estructuras disponibles. Nota: no todas las estructuras se pueden combinar con todos los formatos de ficheros.	На тази стъпка може да се избере оформление. В момента са налични три типа оформления. Забележка: не всяко оформление може да се съчетае с всеки файлов формат
Catalogue	Katalog	Catálogo	Каталог
Table	Tabelle	Tabla	Таблица
Virtual Gallery	Virtuelle Ausstellung	Galería virtual	Виртуална галерия
Select a File Format:	Wählen Sie ein Datei-Format aus.	Seleccione un formato de fichero:	Избор на файлов формат
Three File Formats are supported: PDF, HTML, SMIL. In combination with the selected layout a proper stylesheet is applied to the resultset and transformed via XSL transformation and XSL:FO rendering	Drei Datei-Formate werden unterstützt: PDF, HTML, SMIL. In Kombination mit dem ausgewählten Layout wird das entsprechende Stylesheet auf das Suchergebnis angewandt und mittels XSLTransformation und XSL:FO rendering in Publikation verwandelt.	Hay tres formatos de ficheros disponibles: PDF, HTML, SMIL. En combinación con la estructura elegida, se aplicará a los resultados una hoja de estilo apropiada y se transformarán usando XSL y XSL:FO.	Поддържат се три файлови формата: PDF, HTML, SMIL. При избрано оформление системата прилага подходящ стил, реализиран чрез XSL трансформация и XSL:FO представяне на данните
The Publication has successfully been created!	Die Publikation wurde erzeugt!	¡La publicación ha sido creada con éxito!	Публикацията е създадена успешно !



Start the Publication	Starten der Publikation	Iniciar la publicación	Започни публикацията
Download the Publication	Download der Publikation	Descargar la publicación	Зареди публикацията
Restart	Neustart	Reiniciar	Рестарт
Produce a new Publication	Eine neue Publikation erzeugen	Producir un nueva publicación	Нова публикация
Refine this Publication	Diese Publikation bearbeiten	Refinar esta publicación	Усъвършенстване на публикацията
Go back to Regnet System	Zurück zum Regnet System	Volver al sistema Regnet	Връщане към REGNET системата
Map	Karte	Mapa	Карта
Select a Map	Eine Karte auswählen	Seleccionar mapa	Избери карта
Active Anchor	aktiver Anker	Anclaje activo	Текущо местонахождение
Delete Active	aktiven Löschen	Eliminar activo	Изтрий активния
Save	Speichern	Guardar	Съхрани
Name	Name	Nombre	Име
width	Breite	Anchura	ширина
height	Höhe	Altura	височина
anchorlist	Ankerliste	Lista de anclajes	списък на местонахожденията
next	weiter	Siguiente	следващ
New Recordgroup	neue Datengruppe	Nuevo grupo de registros	Нова група за запис
Assign Record	Datensatz zuweisen	Asignar registro	Предай записа
Remove Record	Datensatz entfernen	Eliminar registro	Отстрани записа
Assigned Records	Zugewiesene Datensätze	Registros asignados	Предадени записи
Back	Zurück	Volver	Назад
Available Records	Verfügbare Datensätze	Registros disponibles	Налични записи
Start Layout-Editor	Layout-Editor starten	Iniciar el editor de estructuras	Редактирай оформлението



The Layout Editor can be used to create a user defined layout of the publication

Der Layout Editor dient dazu, ein benutzerdefiniertes Layout zu erstellen.

El editor de estructura puede usarse para crear un layout definido por el usuario para la publicación

Редакторът за оформлението може да се използва за създаване на потребителско оформление на публикацията

2.10.2 Form management

The excel sheet provided for the translations are transformed into Java resource bundles for internationalisation. A resource bundle consists of labels used internally by an application mapped to language specific terms, e.g. the internally used label "dog" is mapped to "Hund" in the german resource bundle and to "dog" in the english one.

Below there is example for the German resource bundle file used:

SelectPub=Wählen Sie die zu ladende Publikation

RegnetPrototype=Regnet Prototyp für Electronic Publishing

query=oder führen Sie eine Suchanfrage über die Regnet-Datenbanken aus.

LoadPub=Publikation laden

Perform=Suche ausführen

Continue=Weiter

SelectStoryboard=Wählen Sie ein Storyboard:

StoryboardExplanation=In diesem Schritt des Publikations Prozesses kann ein Storyboard ausgewählt oder bestehende Publikationsdaten geladen werden. Storyboards bieten die Möglichkeit, die Suchergebnisse nach bestimmten Kriterien zu gruppieren. Diese Kriterien werden LATCH-Prinzipien genannt.

Submit=Abschicken

SelectLayout=Wählen Sie ein Layout aus:

LayoutExplanation=In diesem Schritt kann ein Layout ausgewählt werden. Derzeit werden drei Layouts unterstützt. Nicht jedes Layout kann aber mit jedem Fileformat kombiniert werden.

Catalogue=Katalog

Table=Tabelle

VirtualGallery=Virtuelle Ausstellung

SelectFileformat=Wählen Sie ein Datei-Format aus.

FileformatExplanation=Drei Datei-Formate werden unterstützt: PDF, HTML, SMIL. In Kombination mit dem ausgewählten Layout wird das entsprechende Stylesheet auf das Suchergebnis angewandt und mittels XSLTransformation und XSL:FO rendering in Publikation verwandelt.

PublicationSuccessful=Die Publikation wurde erzeugt!

Start=Starten der Publikation

Download=Download der Publikation

Restart=Neustart

Produce=Eine neue Publikation erzeugen

Refine=Diese Publikation bearbeiten

Goback=Zurück zum Regnet System



Map=Karte

SelectMap=Eine Karte auswählen

ActiveAnchor=aktiver Anker

DeleteAnchor=aktiven Anker löschen

Save=Speichern

Name=Name

Width=Weite

Height=Höhe

anchorlist=Ankerliste

next=weiter

NewRecordgroup=neue Datengruppe

AssignRecord=Datensatz zuweisen

RemoveRecord=Datensatz entfernen

AssignedRecords=Zugewiesene Datensätze

Back=Zurück

AvailableRecords=Verfügbare Datensätze

StartLayoutEditor=Layout-Editor starten

LayoutExplanation=Der Layout Editor dient dazu, ein benutzerdefiniertes Layout zu erstellen.

AdvancedSearch=Erweiterte Suche

Search=Suche

SimpleSearch=Einfache Suche

SelectDB=Datenbankauswahl

SelectDocTypes=Dokumenttypauswahl

Display=Anzeige

Matches=Treffer pro Seite

fulltext=Volltext

StartPublication=Start des Publikationsvorganges

ChooseStoryboard=Storyboardauswahl

ChooseLayout=Layoutauswahl

ChooseFileFormat=Dateiformatauswahl

PublicationEnd=Ende des Publikationsvorganges

SelectPubData=oder laden Sie gespeicherte Publikationsdaten:

NoStoryBoard=Kein Storyboard

Additionally to display the characters of each language correctly, different lookup tables with byte-to-character mappings have to be used.

The default table is specified by ISO-8859-1, also called Latin-1. It contains byte-to-character mappings for the characters most commonly used in Western European Languages. ISO-8859-5 defines a byte-to-character mapping for the characters of the Cyrillic alphabet, used in Bulgarian and Russian.



For standard implementation details of internationalisation in Java please see the references on the following website:

<http://developer.java.sun.com/developer/technicalArticles/Intl/>



3 Front End Integration

A first draft has been designed that can be used for the front end integration of the Regnet system. The specific template can be seen in the following picture (Figure 5).

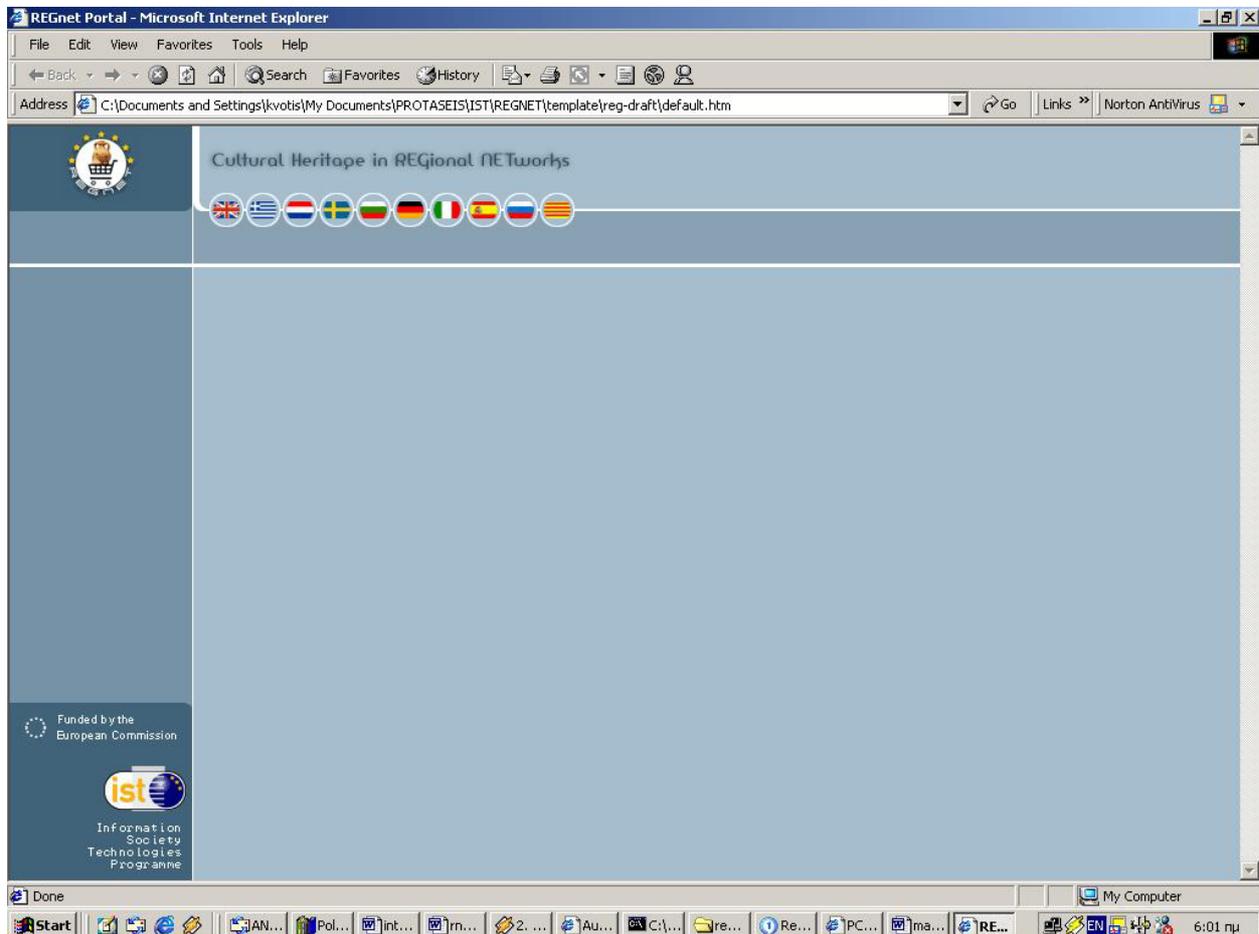


Figure 5: The front end integration template

This template has been used in order to adapted the portal and e-shop components. A first draft of portal can be viewed in the next figure (Figure 6).

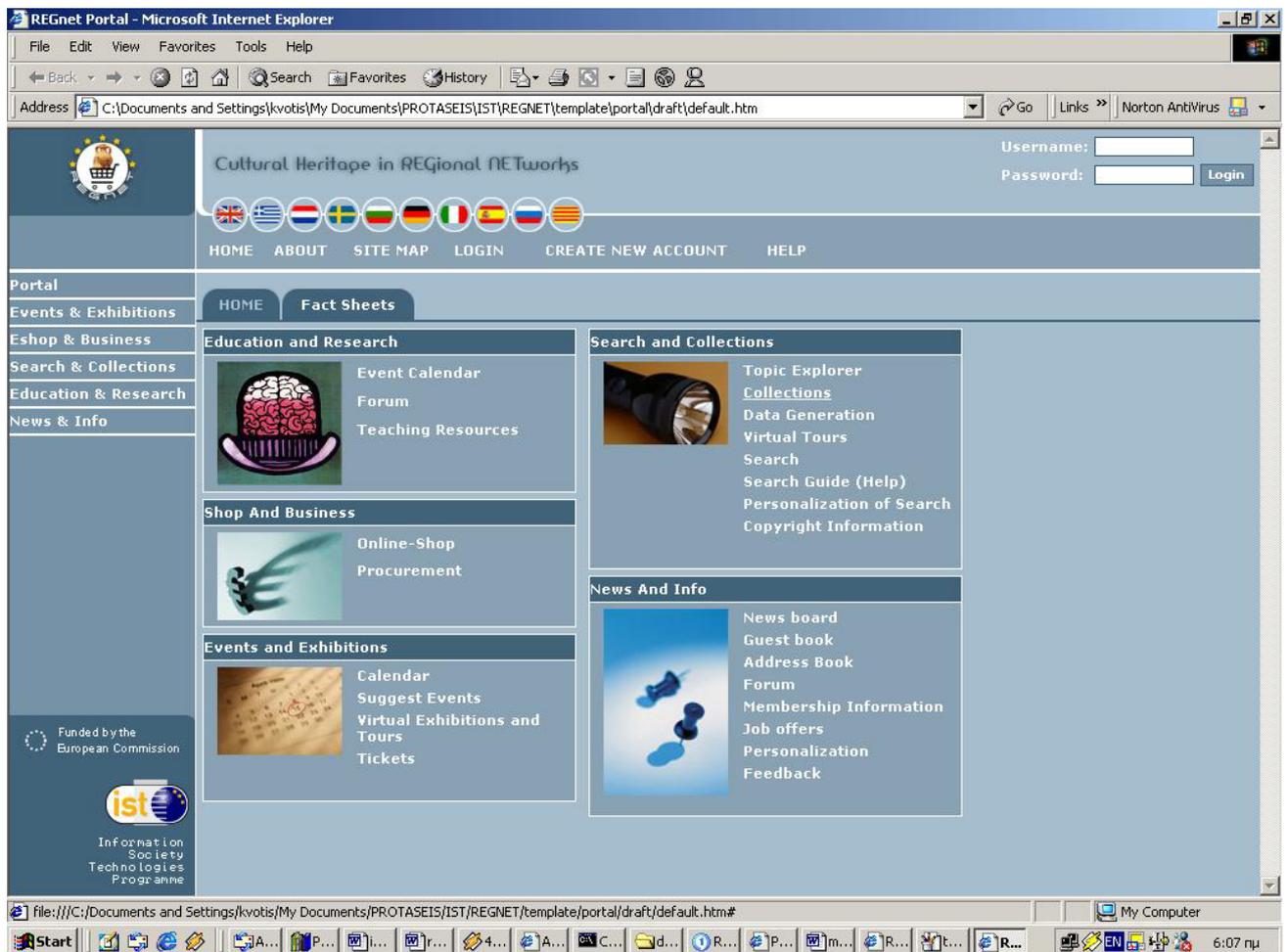


Figure 6: The interface of the portal using the specific tem

Furthermore in the next pictures (Figure 7, Figure 7Figure 8) we can see some samples from the e-shop component, after the adaption of the disigned template.

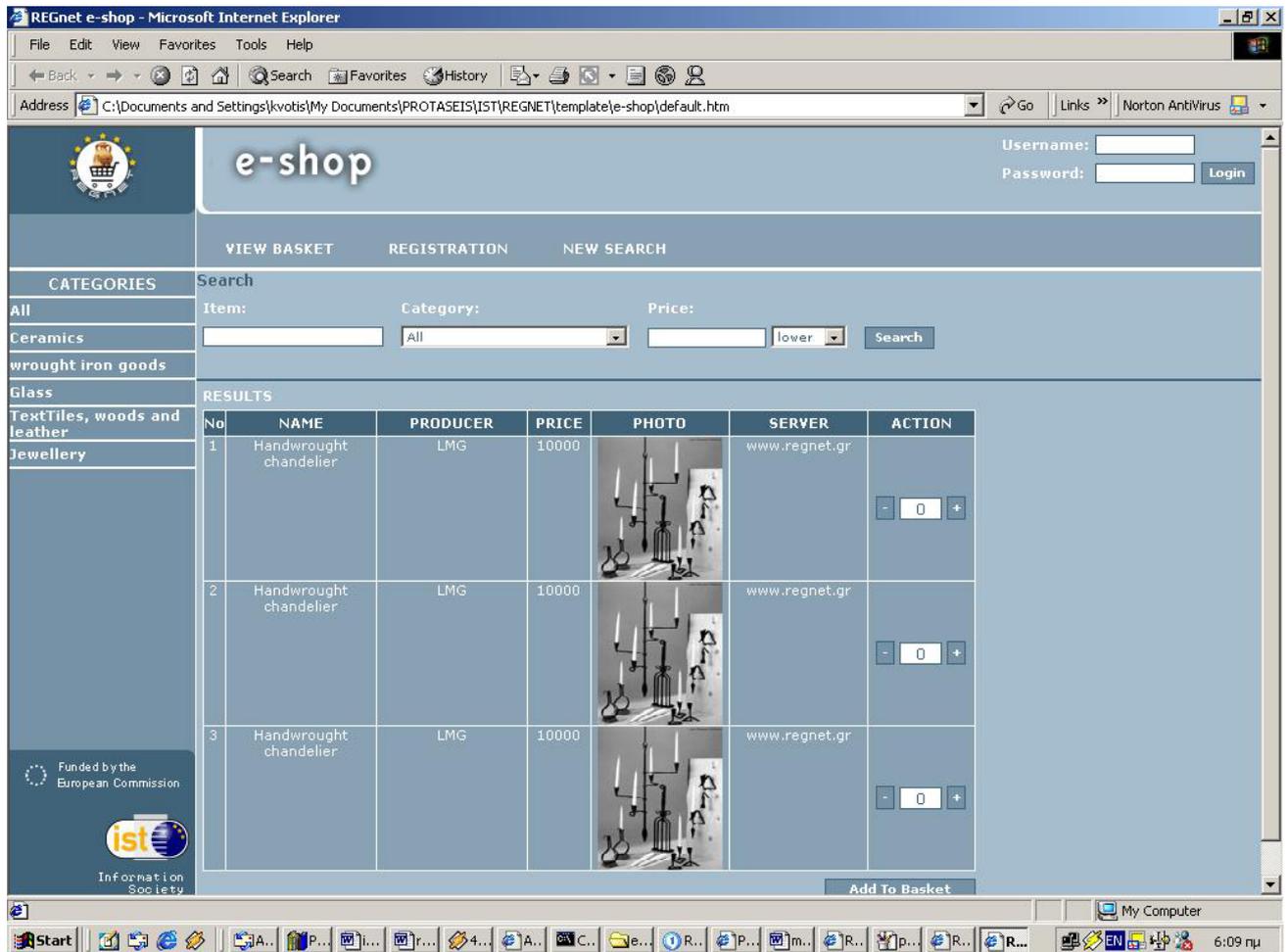


Figure 7: The default page of the e-shop interface using the specific template

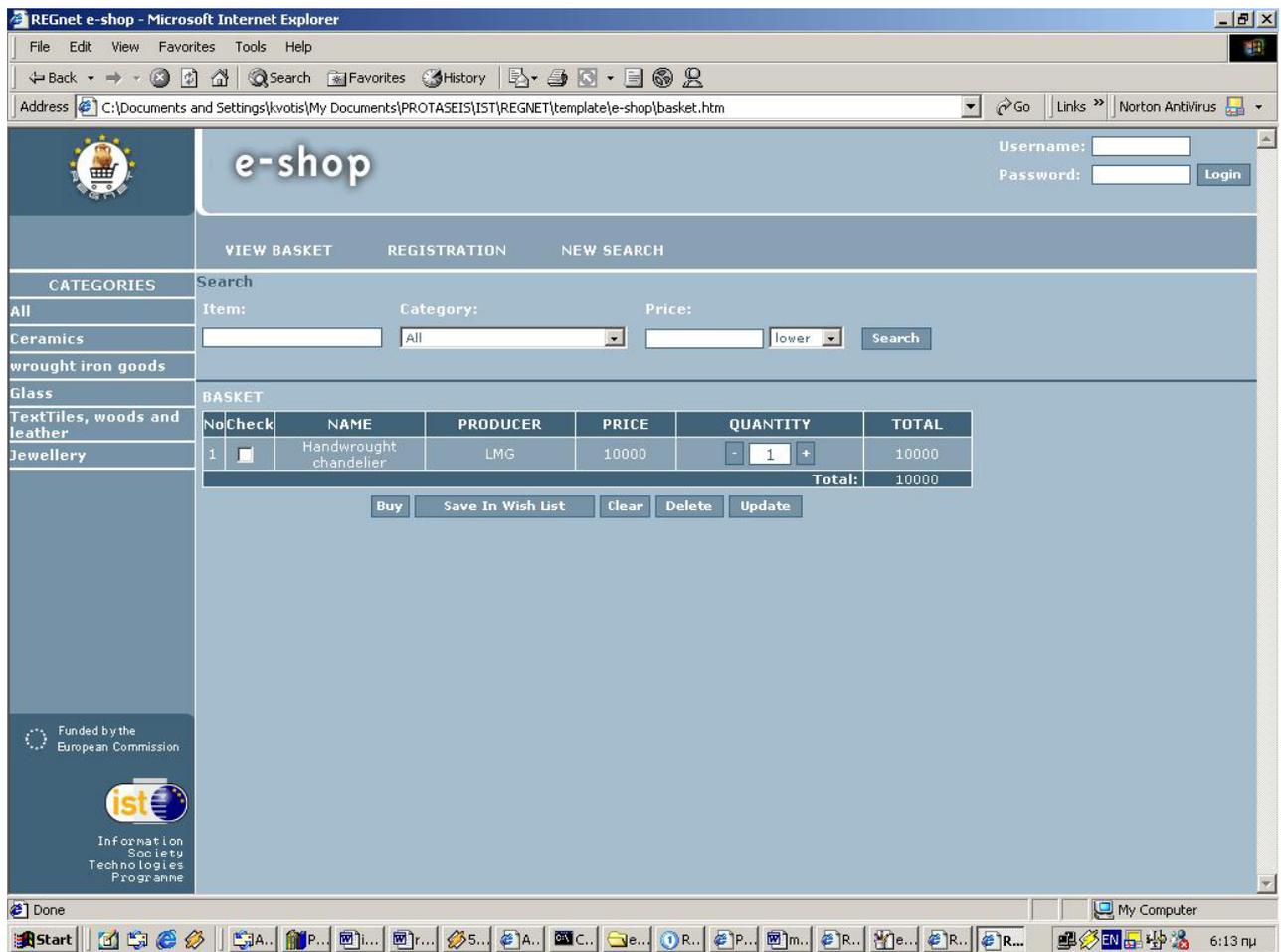


Figure 8: The interface of the e-shop's basket

4 Topic Map

The development of a web-based authoring tool for topic maps was essential since it facilitates one of the main tasks that the content providers have to undertake. This authoring tool gives to the content providers an easy way to produce XTM files and store them automatically in the Knowledge Base for immediate use by all REGNET modules.

4.1.1 Topic Map format

The new technology of Topic Maps provides a knowledge representation of real-world subjects, collected in indices, catalogues and taxonomies. Specifically for the REGNET project, Topic Maps provide the way to create metadata for the museum items as well as general information concerning the way these will be organized. These data are stored in the central node of the System, the Ontology, and are readily available for use by the interconnecting REGNET modules (publishing, business, portal, search system, etc). A Topic Map is constructed of various elements that describe real-world items like a painter or a sculpture (a “topic” element), and the association/relation between them (an “association”). Further elements include occurrences, subject descriptors, variants of the original name, instances of a topic, and scopes of various elements. The programming format used to map the Topic Map concept to a programming language is XTM, which stems from XML – it is well formed XML and with a specific DTD applied. The main advantage of XTM is that almost all types (topics, associations, occurrences, etc) are defined as topics. This design gives tremendous power to the model, allowing among other issues for the topic map (ISO/IEC 13250) to be self-documenting.

4.1.2 Architecture of the authoring tool

As already stated, the “Topic Maps Generator” tool is directly connected to the Knowledge Base and that is the reason, which makes the tool innovative. No other effort is required from both the user and the programmer point of view, in order to make the data accessible from every REGNET module.

The main difference between the first version and the second version of the tool, as far as it concerns the architecture design, is the use of SOAP protocol instead of the RMI as shown in Figure 9.

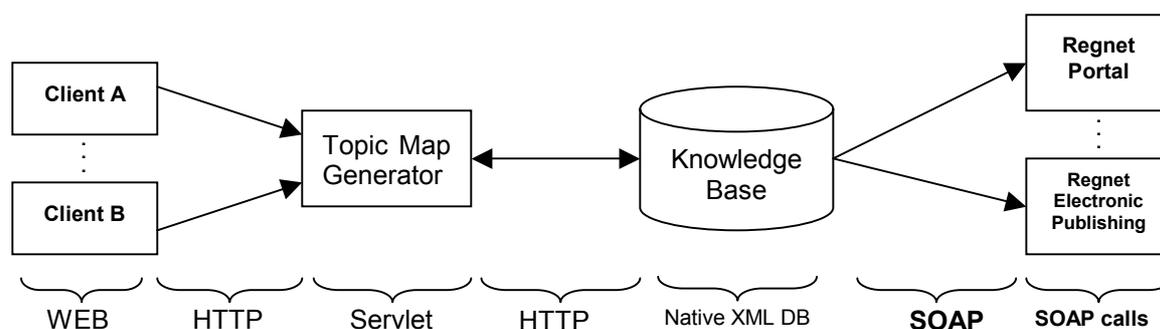


Figure 9: Architecture of the Topic Map Generator Tool (version 2)

The transition from RMI to SOAP was necessary, since the various modules of the REGNET system are based not only in Java (RMI supports interconnection with Java only) but in other programming languages (like PHP) as well. Also SOAP provides an uninterrupted connection between nodes without concerning the use of firewalls, as RMI does.

4.1.3 User interface of the authoring tool

The tool is user friendly since it is web-based and always prevents errors by checking the user's input. From the very beginning, Topic Maps Generator was developed to be used by the content providers of REGNET, and fully supports direct interconnection with the REGNET Ontology. In version 2.0, every user has the ability to create XTM files quickly and easily and download them to his/her computer. The

graphical user interface is completely new compared to version 1.0 and has many similarities with some of the most known commercial products (see Figure 10), thus the users don't have to learn a completely new interface. Another brand new feature is the uploading of XTM for further editing. The user could import an XTM file that is stored locally in the knowledge base from inside the tool.

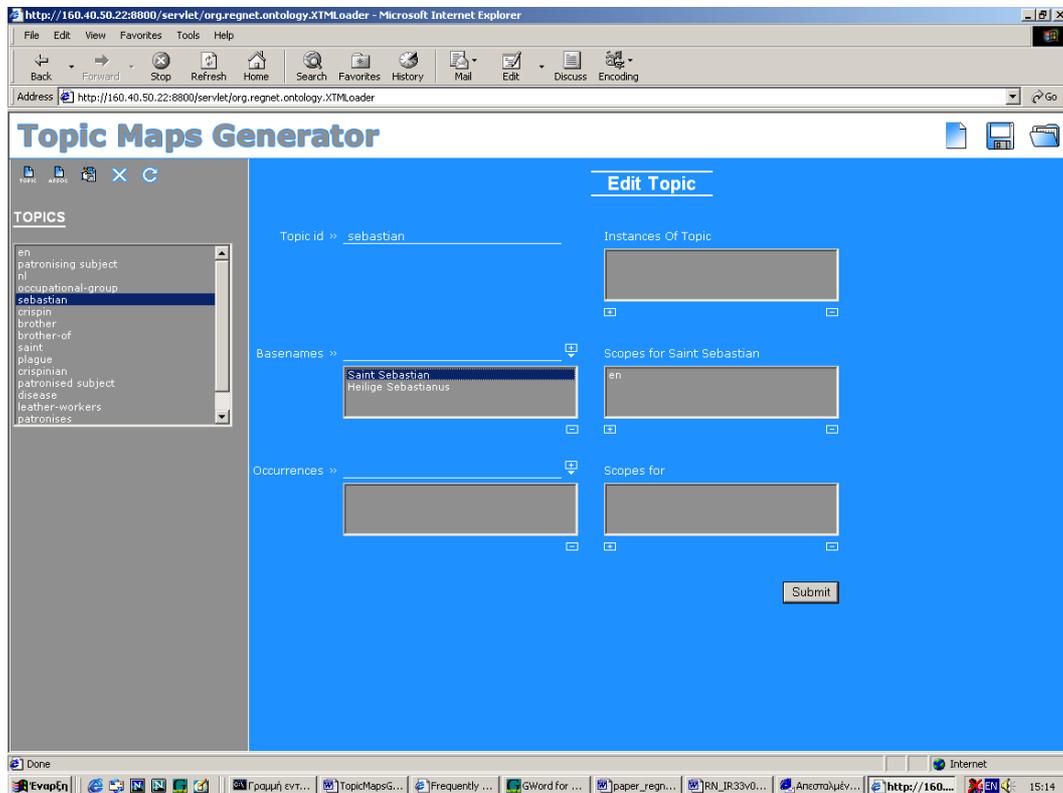


Figure 10: Screenshot of the “Topic Map Generator” tool

The features offered by the tool for creating an XTM file are the following:

<Topic> : It can have multiple <baseName> elements, each one with its own <scope>, and multiple <occurrence> elements, again each one having a unique <scope>.

<Association> : It can have multiple <member> elements, and multiple players, each one with its specific role. It can also have an <InstanceOf> and multiple <scope> elements.

Apart from these technical features the tool has got extra features to make the creation and storage of Topic Maps more convenient, since it has direct access to the Ontology Knowledge Base where Topic Maps are stored. The user is able to:

- Start the creation of a new XTM file. Concurrent requests are also supported since the tool makes use of sessions.
- Open an existing XTM from the Knowledge Base and edit it.
- Open multiple existing XTM files from the Knowledge Base in order to extract their topics and associations.
- Import a local XTM file for editing.
- Show the current XTM file produced onscreen
- Export the produced file for off-line viewing
- Save the created file to the Knowledge Base so that it can be readily accessible from the REGNET nodes.



The XTM standard has only a very limited set of mechanisms for defining constraints on the map [2]. The XTM Authoring Tool provides such mechanisms in order to avoid omissions and errors from users by creating automatically missing topics. Finally, it must be signified that the produced XTM file is fully compatible with the XTM 1.0 specifications "<http://www.topicmaps.org/xtm/1.0/>".

5 Distributed REGNET

5.1 Multi-Portal

REGNET infrastructure is constituted by a network of Cultural Service Centres (CSC) where each CSC has a Portal for user access and it could supply the whole set of available services or a part of them. Moreover, some services could reside locally to the CSC and other remotely. In general a service can reside also out of a CSC. In this case it is implemented by a subsystem accessed through a Portal of one of the CSCs.

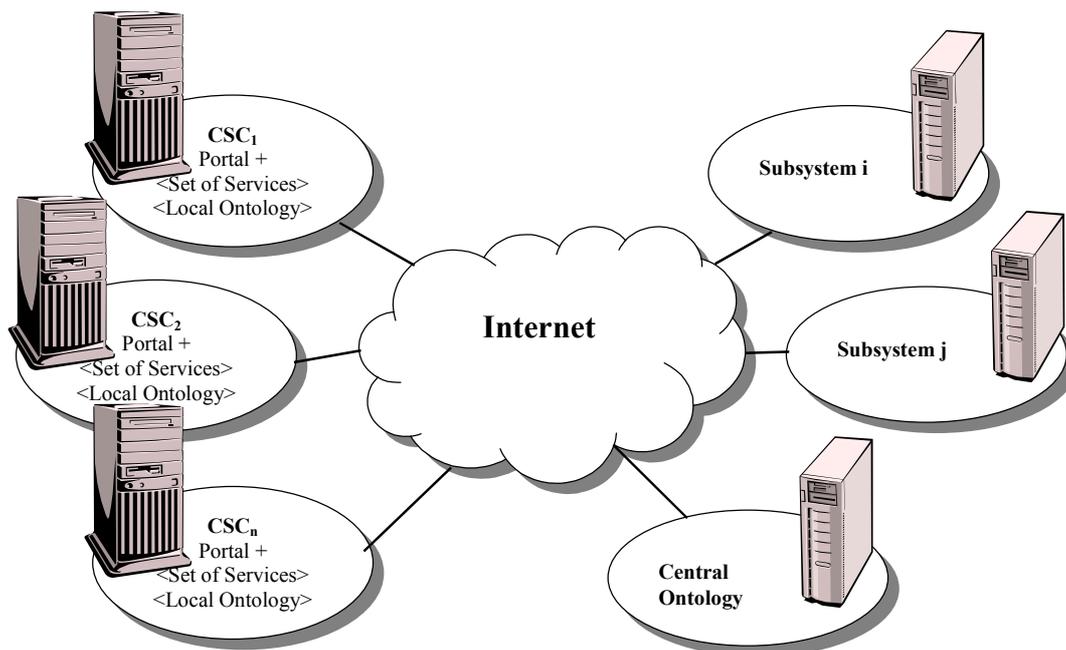


Figure 11: Multi-Portal architecture

A unique point of access redirects the user to the preferred CSC Portal but, in general, the user must be able to access the services he subscribed and his own profile from every Portal.

From the point of view of the management of Portals, the Ontology component of REGNET infrastructure acts as central repository to store and keep consistent all the information about the user. Each Portal can refer to the Ontology to retrieve these kinds of data. Nevertheless, each Portal has its own data management capability, based on a local database. This made necessary the development of opportune procedures of synchronization between the Portals and the Ontology.

Each CSC integrates a Portal and a local node of the Ontology. The local Ontologies are connected to a central Ontology node, so that each Portal, by connecting to the local Ontology, receives the same services as it was connected to the central one. Nevertheless, given that the interface between the Portal and the Ontology is SOAP based, the location of the Ontology respect to the Portal that refers to it is not a technical issue. Therefore, hereafter in this section we will refer to a unique Ontology connected to all the Portals.

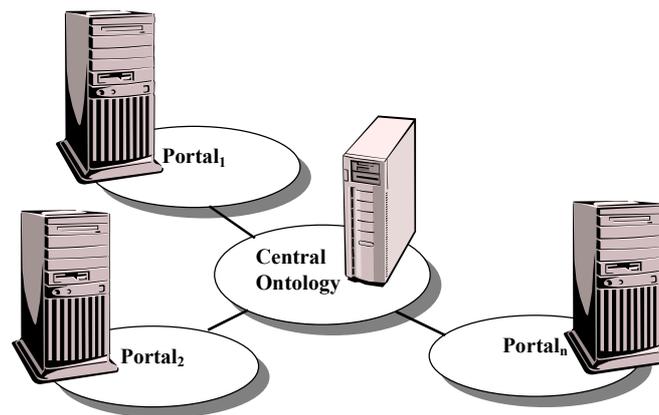


Figure 12: Logical architecture for portals and ontology

Jetspeed Portals supply a lot of functionalities and development API to access user information based on a local data management. For the sake of REGNET customisation, a trade-off was established in order to meet the best multi-Portal synchronization with a reasonable development effort, to avoid vanishing the support offered by the tool. Therefore, it was identified a set of functionalities that require the implementation of a synchronization procedure, while the rest of the functionalities rely on the native support offered by Jetspeed. This set includes:

- Creation of a new Account
- Editing of data related to an existent Account
- Login
- Deletion of an existent Account

5.1.1 Creation of a new Account

This is the simplest functionality in the context of multi-Portal synchronization, because it does not exist any information local to other Portals or in the central repository that must be kept consistent with the new account to be created.

An account consists in the couple username/password to authenticate the user, the mandatory fields first name, last name and email, and other information according to the REGNET User Profile.

Therefore, the creation of a new account consists in the gathering of the user profile from a web form, the creation of a XML structure to be posted to the central repository, and the creation of the structure in the local database. This was obtained by extending the class

org.apache.jetspeed.modules.actions.CreateNewUserAndConfirm

as Figure 13 shows:

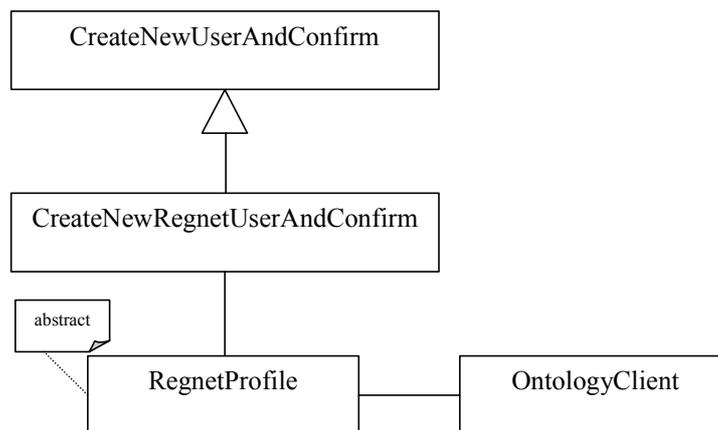


Figure 13: Editing an existent account: class diagram

5.1.2 Editing of data related to an existent Account

The update of data related to an existent Account required the customisation of the class

org.apache.jetspeed.modules.actions.EditAccount

in charge of preparing the form to collect the user information. Natively, this form is filled out with data coming from the local database. This piece of data must be considered potentially not coherent with the profile stored in the Ontology, because some information could be changed during a previous access via another Portal. The REGNET version of this class ignores the local data and retrieves the user profile from the central repository.

The next steps, as for the creation of a new account are the gathering of the user profile from the web form, the creation of a modified XML structure to be posted to the central repository (identified by a unique *userid*), and the update of the local database. This was obtained by extending the class

org.apache.jetspeed.modules.actions.UpdateAccount

as Figure 14 shows:

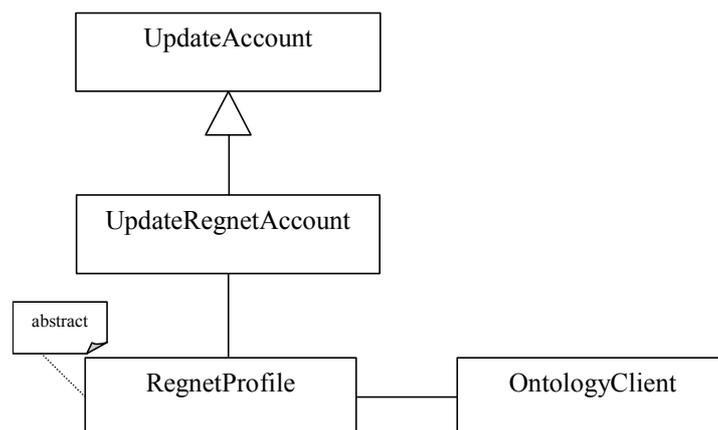


Figure 14: Creation of a new account: class diagram

5.1.3 Login

The customisation of the login functionality was the most sensitive. The login includes user authentication and the loading of the user profile.

First, the user is sought in the central repository by the username. If the user is found, user's inserted password is authenticated respect to the one stored in the Ontology profile.

Once authenticated respect to the Ontology, the user must be authenticated locally. This raises the problem of a potential change of password operated previously from another Portal. In this case, the local user password could be different from the one inserted by the user and verified in the Ontology. Detected the condition, the local password is forced to the new password, leaving to the next step the alignment.

The last step is to align the local information with data retrieved from the Ontology. This includes also the password.

It could happen that the user authenticated in the central repository does not exist locally. This is the case of the first login to a Portal for a user already registered from another Portal. If so, before the local authentication, a user account must be created transparently to the user, by the information retrieved from the Ontology.

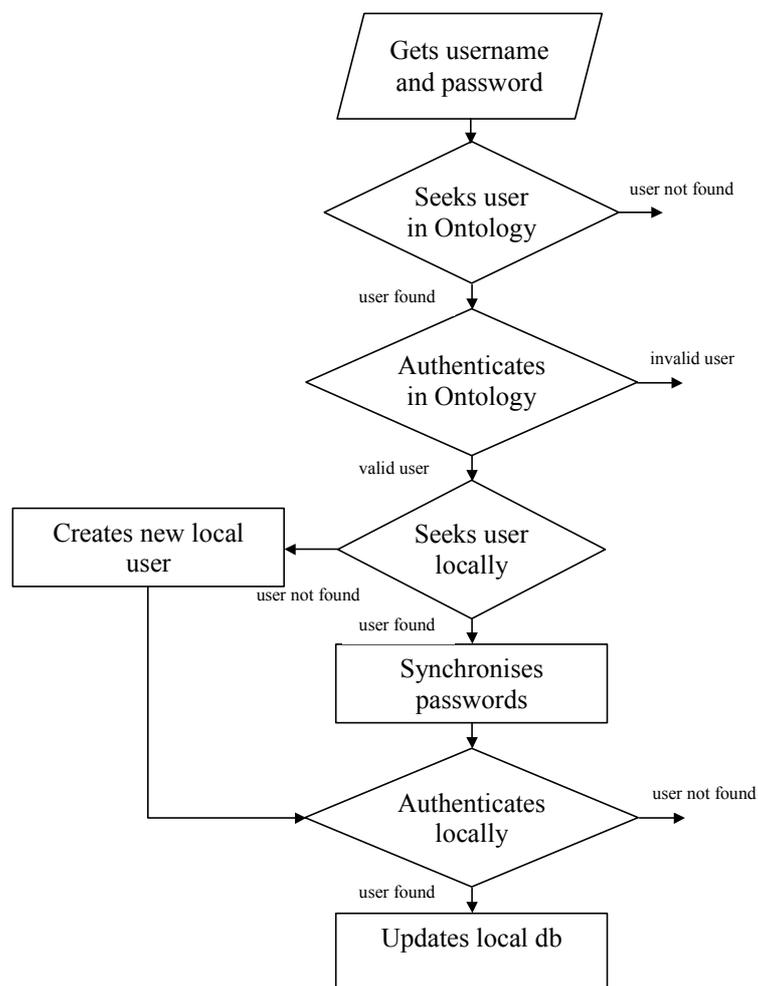


Figure 15: Authentication algorithm

Figure 15 summarizes the algorithm described above. All the implementation was obtained modifying the class

org.apache.jetspeed.modules.actions.JloginUser

The basic installation of Jetspeed includes an administration account and a simple user. While the simple user was deleted, the administrator was kept locally, but not in the Ontology. Therefore, each Portal has an administration account independent from the others.



5.1.4 Deletion of an existent Account

This functionality is reserved to the administrator. It is possible to delete completely the registration of the user, but it is also possible to delete only the local account, allowing the user to enter the system from another Portal.

5.2 REGNET connector

The regnet connector is used in order to set up B2B interaction between many REGNET CSCs. In order to achieve enough quality of service and standardisation, we choose to use the ebXML (www.ebxml.org) framework for this task.

In order to set up this ebXML connector, as explain below, many components are necessary:

- Public services modelised as Business Processes (BP) are stored into an ebXML registry. These business processes as been established during the previous work-packages. We choosed to implement at first the BP dealing with the establishment of a (virtual) exhibition.
- In order to manage reliable communication between CSCs we need to set-up a secure communication channel between them.

As far as ebXML framework is a new standard, softwares to implement it are very few so we decided to developed our implementation for ebXML messaging (based on JAXM: <http://java.sun.com/xml/jaxm/>) and to use the open source registry ebXMLrr (<http://ebxmlrr.sourceforge.net/>).

This chapter is organised into two parts: the first one deals with a description of ebXML (more detail that the one done for deliverable D2 of WP1), the second one describe the developed connector and necessary registry.

5.3 EbXML at work

5.3.1 Quick ebXML description

ebXML (Electronic Business using eXtensible Markup Language), sponsored by UN/CEFACT and OASIS, is a modular suite of specifications that enables enterprises of any size and in any geographical location to conduct business over the Internet. Using ebXML, companies now have a standard method to exchange business messages, conduct trading relationships, communicate data in common terms and define and register business processes.

5.3.2 The need for ebXML

Up until now, the technology available for most businesses to exchange data was electronic data interchange or EDI, which made significant contributions to productivity and inventory control. Many companies, however, find EDI expensive and difficult to implement. ebXML, using the economies of scale presented by the Internet, breaks through these obstacles.

The conventions established by ebXML are available publicly. These conventions encourage software developers to build packaged applications based on the common structure and syntax of ebXML messages and dramatically lower the cost of exchanging business data.

One of the technical foundations of ebXML is the Extensible Markup Language (XML) that allows parties to exchange structured data, like the information kept in databases, over the Internet. XML is an open and freely available document from the World Wide Web Consortium and has the support of the world's leading technology companies. XML also supports Unicode that enables the display and exchange of most of the world's written languages.

- **ebXML Value**

- Provides the only globally developed open XML-based Standard built on a rich heritage of electronic business experience.
- Creates a Single Global Electronic Market Enables all parties irrespective of size to engage in Internet-based electronic business. Provides for plug and play shrink-wrapped solutions.
- Enables parties to complement and extend current EC/EDI investment expand electronic business to new and existing trading partners.
- Facilitates convergence of current and emerging XML efforts.
- **ebXML delivers the value by**
 - Using the strengths of OASIS and UN/CEFACT to ensure a global open process.
 - Developing technical specifications for the open ebXML infrastructure.
 - Creating the technical specifications with the world's best experts.
 - Collaborating with other initiatives and standards development organizations.
 - Building on the experience and strengths of existing EDI knowledge.
 - Enlisting industry leaders to participate and adopt ebXML infrastructure.
 - Realizing the commitment by ebXML participants to implement the ebXML technical specifications.

5.3.3 ebXML Architecture

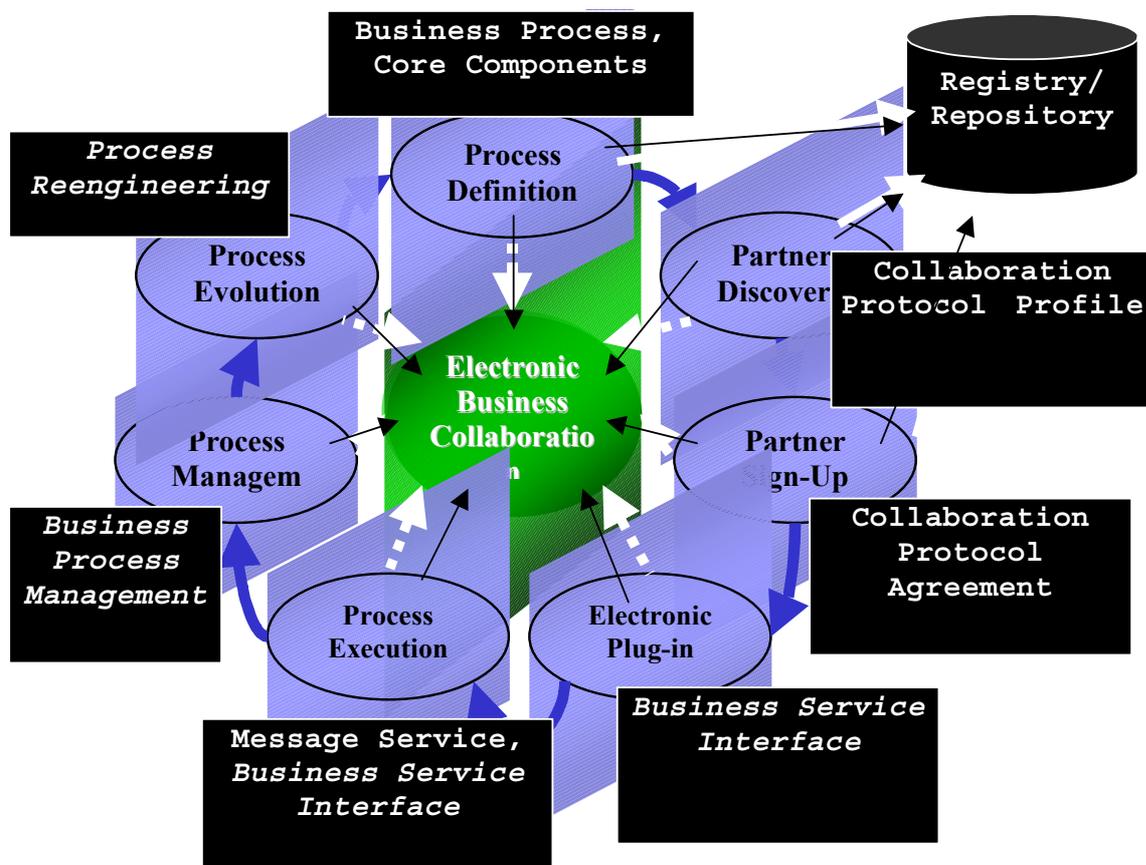


Figure 16: Relationship between “classic” Electronic Business Collaboration phases and ebXML stack

The figure above shows the relationship between “classic” Electronic Business Collaboration phases and the ebXML stack.

We can see that an ebXML compliant system is composed of the following parts:

- a messaging service: ebMS
- a registry/repository: provides finding and storage services
- Business Processes: that describe how a company is doing a specific business
- Core Components : basic « bricks » that can be reused in order to build business documents
- Collaboration Protocol Profiles : what a company can do
- Collaboration Protocol Agreements: how to company can conduct business together

All these different part will be detailed further in this document.

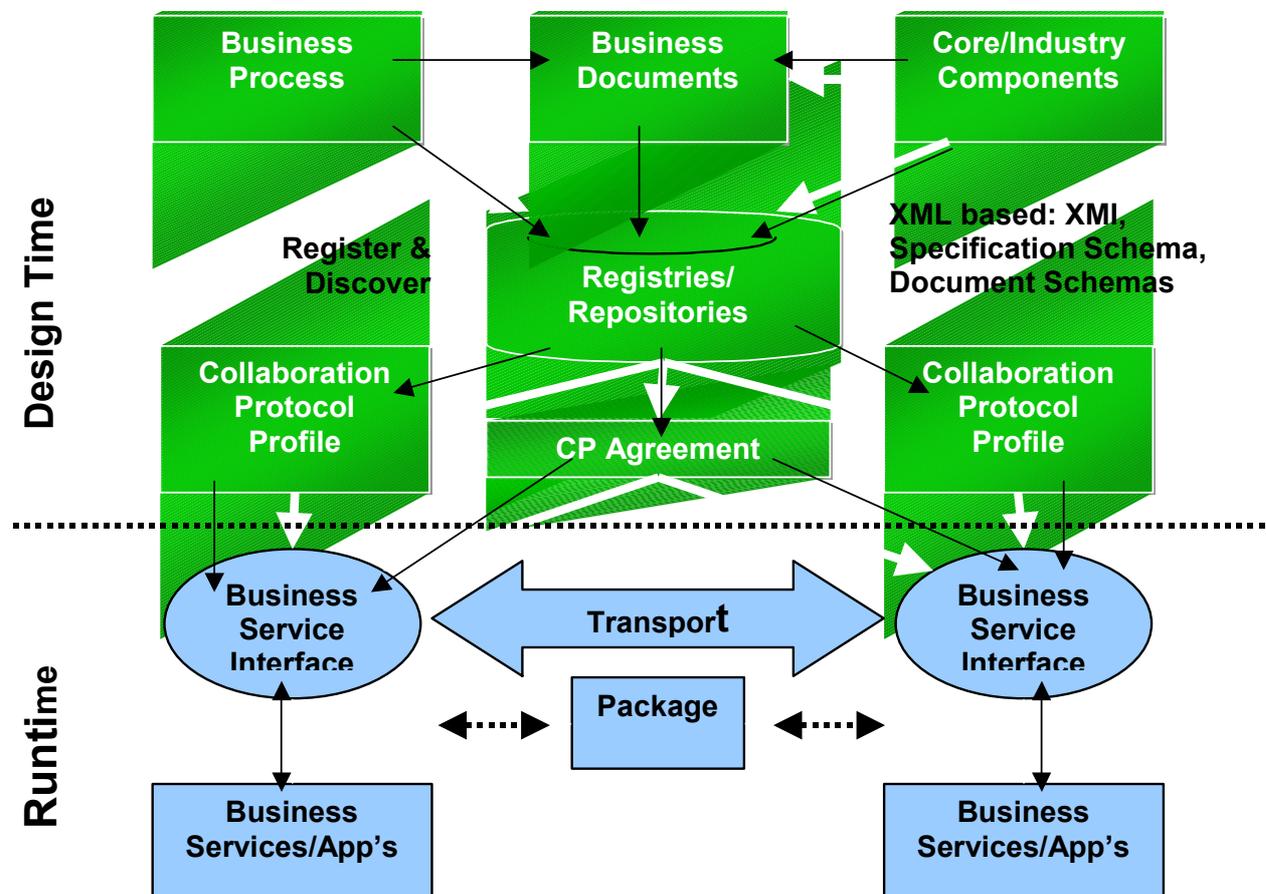


Figure 17: Architecture of an ebXML system

Figure 17 describes the architecture of an ebXML system.

During the design time a Registry/Repository is set up (or reused) in which core components are stored. This is at the design level also that Business Processes and collaboration Protocol Profiles are set up. Out of these two an agreement is defined (more explanation will be found later) that describe how two company can conduct business.

At the Run Time the Collaboration Profile Agreement is used in order to set up business interfaces that will communicate using the ebXML transport service: ebMS.

5.3.4 Example

In order to give a practical explanation of how ebXML is working let's take an example.

A company A willing to conduct business using ebXML must first modelize the business processes involved in the trade(1), implement an ebXML compatible system (2), then record its processes in a database (registry/repository) (3).

A company B already owning a system conformed to the ebXML specifications can discover and obtain information on company A (4).

According to these information the company B proposes a business exchange protocol (5) to the company A. If the agreement is validated, the companies conduct business together (6).

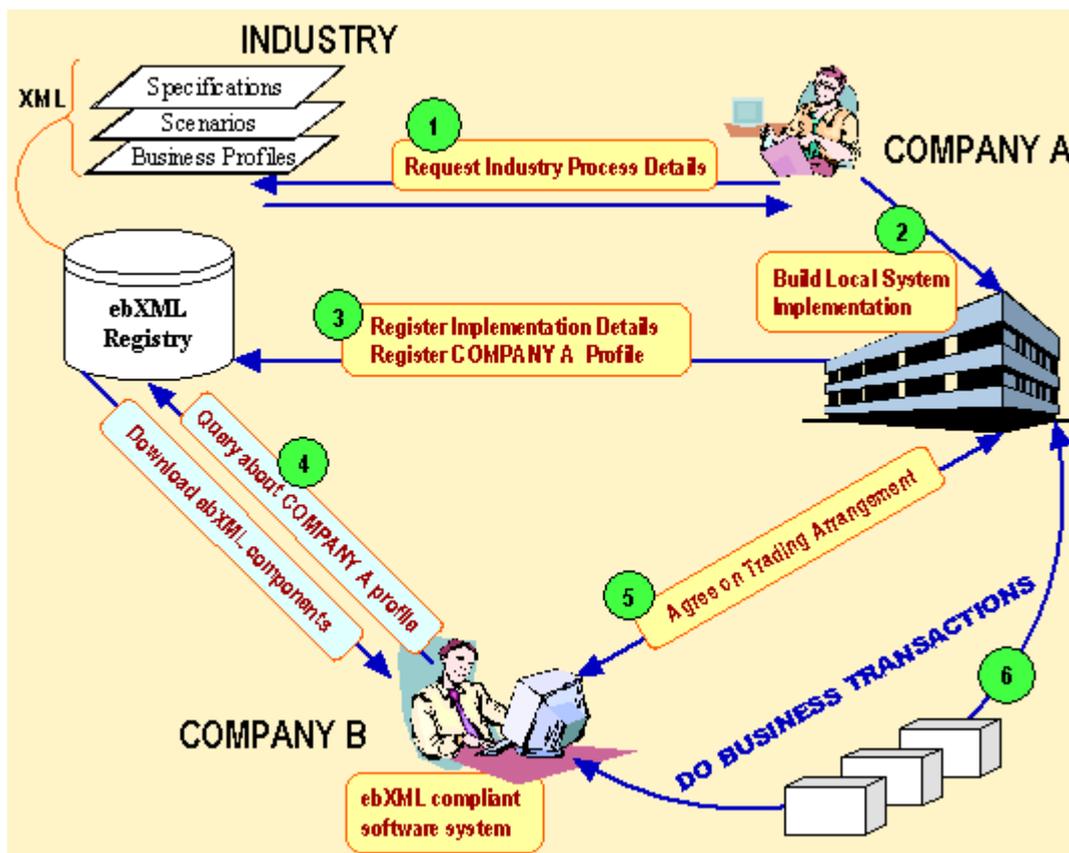


Figure 18: The ebXML system

5.3.5 ebXML messaging

5.3.5.1 Purpose

The purpose of the ebMS component is to provide the services needed for business document exchange and to take care only of the mechanics of message exchange, not the message payload.

Based on XML, an obvious choice for implementing ebMS is SOAP (Simple Object Access Protocol). Indeed, SOAP is an open XML protocol, which provides attachment capabilities (SWA: SOAP With Attachment), a key feature for building an ebMS compliant messaging mechanism.

In order to illustrate a possible implementation of ebMS, the code will simulate a communication between two museums, the first asking information to the other about an object. The museum context has been chosen because I am currently working on a cultural heritage European Project: www.regnet.org.

The following code is written in Java, and requires some knowledge about Servlets. In order to implement the SOAP part, JAXM has been chosen as Sun released a stable version of the API, which can be downloaded at <http://java.sun.com/xml/jaxm/>

5.3.5.2 ebMS specifics

In order to build an ebXML messaging service on SOAP, ebMS needs to extend the protocol to areas which are not covered by it: **security and reliability**. ebMS specifies additional mechanisms for routing, guaranteed and ordered delivery. It also relies strongly on the use of the digital signatures through which messages can be 'signed'.

In this implementation, only the basic ebXML messaging elements will be described.

In SOAP, the message is made of an envelope, itself composed by a header and a body, the header containing some routing information and the body the message itself.

An ebXML message (called a message package) is composed by one header container and zero or more payload containers. The header container is encapsulated in the SOAP envelope while the payload containers are SOAP attachments.

The purpose of the header container is to provide routing information, message behaviors, unique identifiers, and so on while the payload container holds the message content.

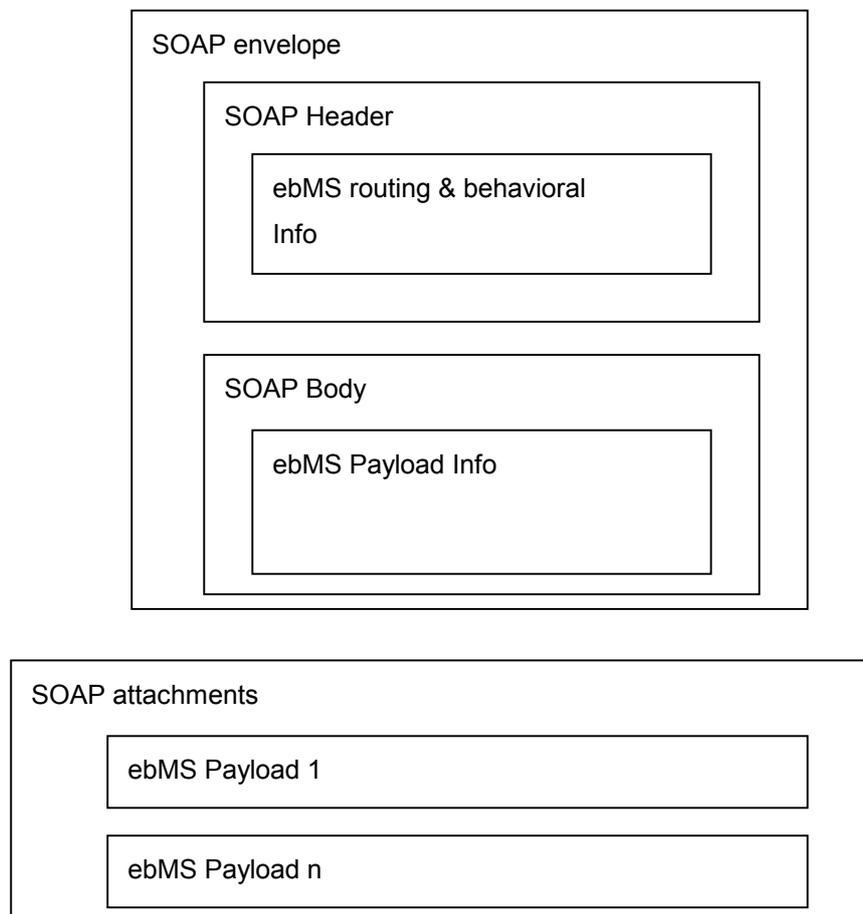


Figure 19: Description of ebMS specifics

Here is a list of elements that can be included in an ebXML header container, this list is not exhaustive but introduces the required fields:

To and From



As it can be deduced from their names, these two elements refer to a unique identifier for the sender and the receiver.

CPAId

This element refers to a unique ID associated with a Collaboration Protocol Agreement (CPA). A CPA defines the process by which the two parties interact.

ConversationId

Must be unique. This element is used to define a set a related message exchanges.

Service

This element is used to refer to the service used to process the message at the destination. In our example we have imagined a service named "ReceiveQuery" which has in charge the reception of all the external requests.

Action

Again the name of this element speaks for itself, as it specifies what action should be taken by the service that processes the message.

MessageId

This element is used to uniquely identify a particular message.

Timestamp

This element holds the date and time of the message's creation.

5.3.6 The Business Process Specifications Schema

One of the primary benefits of ebXML is to improve e-business capabilities of SMEs, focusing on interoperability. We can define interoperability by the way two companies with different way of working (business processes) can harmonize their differences and conduct business together.

Thus, BPSS (business process specification schemas) are a key-issue to address when one intends to implement an ebXML compatible system.

It is important to understand that BPSS can also be used independently of the ebXML framework to capture and communicate business processes in a way that can be understood, without ambiguity, by both involved parties.

The business process used to demonstrate the steps of modelling a BPSS is based on the activities required by a museum that intends to borrow an object to another museum (for an exhibition purpose, for instance). As the BPSS will be used in the ebXML context, it has to express collaboration between two business entities, i.e. to show the different business activities and their choreography involved in the process. That is why the BPSS will describe the exchanges made between the borrower and its counterpart, the lender.

5.3.6.1 BPSS fundamentals

The ebXML *Business Process Specification Schema* provides a standard framework for business process specification. As such, it works with the ebXML Collaboration Protocol Profile (CPP) and Collaboration Protocol Agreement (CPA) specifications to bridge the gap between Business Process Modeling and the configuration of ebXML compliant e-commerce software, e.g. an ebXML Business Service Interface as depicted in Figure 20.

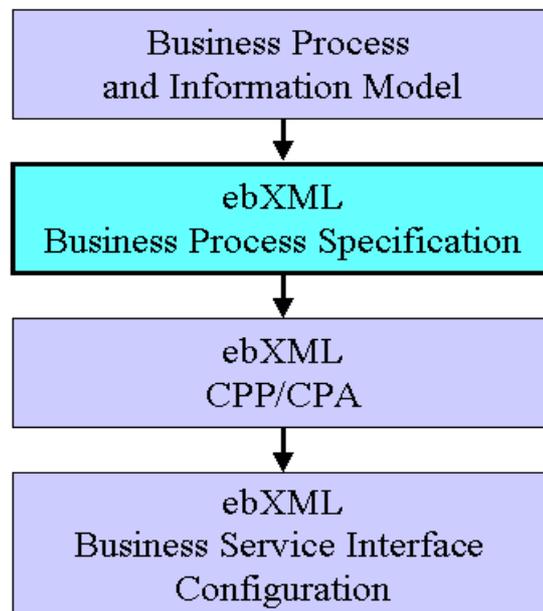


Figure 20: ebXML business service interface

Using Business Process Modeling, a user may create a complete Business Process and Information Model.

Based on this Business Process and Information Model and using the ebXML *Business Process Specification Schema* the user will then extract and format the nominal set of elements necessary to configure an ebXML runtime system in order to execute a set of ebXML business transactions. The result is an ebXML *Business Process Specification*.

Alternatively the ebXML *Business Process Specification* may be created directly, without prior explicit business process modeling.

An ebXML *Business Process Specification* contains the specification of Business Transactions and the choreography of Business Transactions into Business Collaborations.

This ebXML *Business Process Specification* is then the input to the formation of ebXML trading partner Collaboration Protocol Profiles and Collaboration Protocol Agreements.

These ebXML trading partner Collaboration Protocol Profiles and Collaboration Protocol Agreements in turn serve as configuration files for ebXML Business Service Interface software.

The architecture of the ebXML *Business Process Specification Schema* consists of the following functional components:

- UML version of the *Business Process Specification Schema*
- XML version of the *Business Process Specification Schema*
- Production Rules defining the mapping from the UML version of the *Business Process Specification Schema* to the XML version
- Business Signal Definitions

Together these components allow you to fully specify all the run time aspects of a business process model.

These components are shown (inside the dotted box) in the Figure 21.

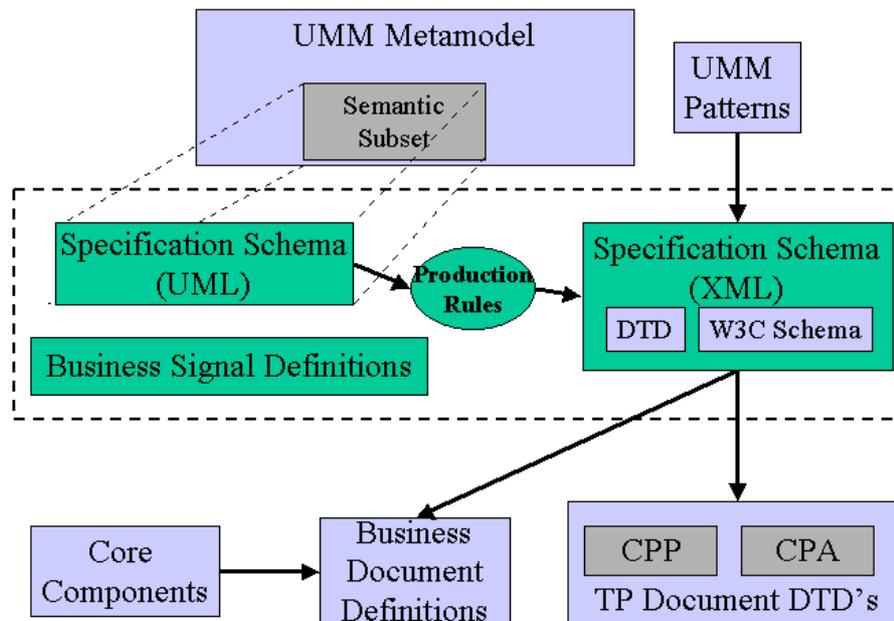


Figure 21: Architecture of the ebXML Business Process Specification Schema

The following provides a description of each of the components in the ebXML *Business Process Specification Schema* and their relationship to UMM, and ebXML CC and CPP/CPA:

UML version of Business Process Specification Schema

The UML version of the *ebXML Business Process Specification Schema* is a semantic subset of the metamodel behind UMM as specified in UN/CEFACT TMWG's N090R9.1

N090R9.1 is as of this writing not yet approved by UN/CEFACT. It is the intent to keep the *ebXML Business Process Specification Schema* and the UN/CEFACT TMWG's N090 semantically aligned.

The UML version of the *ebXML Business Process Specification Schema* is merely a UML Class Diagram. It is not intended for the direct creation of ebXML Business Process Specifications. Rather, it is a self-contained statement of all the specification elements and relationships required to be able to create an ebXML compliant Business Process Specification.

XML version of Business Process Specification Schema

The XML version of the *ebXML Business Process Specification Schema* provides the specification for XML based instances of ebXML Business Process Specifications, and as a target for production rules from other representations. Thus, a user may either create a *Business Process Specification* directly as an XML document, or may choose to use some other means of specification first and then apply production rules to arrive at the XML document version.

Any methodologies and/or metamodels used for the creation of ebXML compliant Business Process Specifications must at minimum support the production of the elements and relationships contained in the XML version of the *ebXML Business Process Specification Schema*.

Both a DTD and a W3C Schema is provided. Each is an isomorphic definition of the UML version of the *ebXML Business Process Specification Schema*.



UMM Business Process Interaction Patterns

ebXML Business Service Interfaces are configured to execute the business processes specified in a *Business Process Specification*. They do so by exchanging ebXML messages and business signals.

Each Business Transaction can be implemented using one of many available standard patterns. These patterns determine the actual exchange of messages and business signals between the partners to achieve the required electronic commerce transaction.

The Business Transaction Interaction Patterns set forth in Chapter 8 of the UMM N090R9.1 document illustrate recommended permutations of message sequences as determined by the type of business transaction defined and the timing policies specified in the transactions.

While the UMM patterns themselves are not part of the ebXML specifications, all the security and timing parameters required to express the pattern properties are provided as attributes of elements in the ebXML

Business Signal Definitions

Business signals are application level documents that 'signal' the current state of the business transaction. These business signals have specific business purpose and are separate from lower protocol and transport signals.

However, the structures of ebXML business signals are 'universal' and do not vary from transaction to transaction. Thus, they can be defined once and for all as part of the ebXML *Business Process Specification Schema* itself.

5.3.6.2 Key Semantics of a Business Transaction

A Business Transaction is the atomic unit of work in a trading arrangement between two business partners.

A business transaction consists of a Requesting Business Activity, a Responding Business Activity, and one or two document flows between them. A Business Transaction may be additionally supported by one or more Business Signals that govern the use and meaning of acknowledgements and related matters in the transaction.

Implicitly there is a requesting role performing the Requesting Business Activity and a responding role performing the Responding Business Activity. These roles become explicit when the transaction is used within a Business Transaction Activity within a Binary Collaboration.

There is always a Request document flow.

Whether a Response document is required is part of the definition of the Business Transaction. Some Business Transactions need this type of request and response, typically for the formation of a contract or agreement. Other Business Transactions are more like notifications, and have only a Request document flow.

5.3.6.3 Key Semantics of a Binary Collaboration

A Binary Collaboration is always between two roles. These two roles are called Authorized Roles, because they represent the actors that are authorized to participate in the collaboration.

A Binary Collaboration consists of one or more Business Activities. These Business Activities are always conducted **between** the two Authorized Roles of the Binary Collaboration. For each activity one of two roles is assigned to be the the InitiatingRole (from) and the other to be the RespondingRole (to).

A Business Activity can be either a Business Transaction Activity or a Collaboration Activity.

A Business Transaction Activity is the performance of a Business Transaction. Business Transactions are re-useable relative to Business Transaction Activity. The same Business Transaction can be performed by multiple Business Transaction Activities in different Binary Collaborations, or even by multiple Business Transaction Activities in the same Binary Collaboration.



A Collaboration Activity is the performance of a Binary Collaboration, possibly within another Binary Collaboration. Binary Collaborations are re-useable relative to Collaboration Activity. The same Binary Collaboration can be performed by multiple Collaboration Activities in different Binary Collaborations, or even by multiple Collaboration Activities in the same Binary Collaboration.

When performing a Binary Collaboration within a Binary Collaboration there is an implicit relationship between the roles at the two levels. Assume that Binary Collaboration X is performing Binary Collaboration Y through Collaboration Activity Q. Binary Collaboration X has Authorized roles Customer and Vendor. In Collaboration Activity Q we assign Customer to be the initiator, and Vendor to be the responder. Binary Collaboration X has Authorized roles Buyer and Seller and a Business Transaction Activity where Buyer is the initiator and Seller the responder. We have now established a role relationship between the roles Customer and Buyer because they are both initiators in activities in the related performing and performed Binary Collaborations.

Since a Business Transaction is atomic in nature, the performing of a single Business Transaction through a Business Transaction Activity is also atomic in nature. If the desired semantic is not atomic, then the task should be split over multiple transactions. For instance if it is desired to model several partial acceptances of a request, then the request should be modeled as one transaction within a binary collaboration and the partial acceptance(s) as separate transactions.

The CPA/CPP Specification requires that parties agree upon a Collaboration Protocol Agreement (CPA) in order to transact business. A CPA associates itself with a specific Binary Collaboration. Thus, all Business Transactions performed between two parties should be referenced through Business Transaction Activities contained within a Binary Collaboration.

5.3.6.4 Key Semantics of a Choreography

A Choreography is an ordering and sequencing of Business Activities within a Binary Collaboration.

The choreography is specified in terms of Business States, and transitions between those Business States.

A Business Activity is an abstract kind of Business State. Its two subtypes Business Transaction Activity and Collaboration Activity are concrete Business States. The purpose of a Choreography is to order and sequence Business Transaction Activity and/or Collaboration Activity within a Binary Collaboration, or across Binary Collaborations within a Multiparty Collaboration.

There are a number of auxiliary kinds of Business States that facilitate the choreographing of Business Activities. These include a Start state, a Completion state (which comes in a Success and Failure flavor), a Fork state and a Synchronization state. These are all equivalent to diagramming artifacts on a UML activity chart.

Transitions are between Business States. Transitions can be gated by Guards. Guards can refer to the status of the Document Envelope that caused the transition, the type of Document sent, the content of the document, or postconditions on the prior state.

A Transition can also be used to create nested BusinessTransactionActivities. A nested BusinessTransactionActivity is one where a first transition happens after the receipt of the request in the first transaction, and then the entire second transaction is performed before returning to the first transaction to send the response back to the original requestor. The flag 'onInitiation' in Transition is used for this purpose. Nested BusinessTransactionActivity are typically within a multiparty collaboration. In essence an Authorized Role in one Binary Collaboration receives a request, then turns around and becomes the requestor in an other Binary Collaboration before coming back and sending the response in the first Binary Collaboration.

isConcurrent is a parameter that governs the flow of transactions. Unlike the security and timing parameters it does not govern the internal flow of a transaction, rather it determines whether multiple instances of that transaction type can be 'open' at the same time as part of the same business transaction activity. IsConcurrent is the parameter that governs this. It is at the business transaction activity level.



5.3.7 CPP Definition

A *CPP* defines the capabilities of a *Party* to engage in electronic *Business* with other *Parties*. These capabilities include both technology capabilities, such as supported communication and messaging protocols, and *Business* capabilities in terms of what *Business Collaborations* it supports.

The **ProcessSpecification**, **DeliveryChannel**, **DocExchange**, and **Transport** elements of the *CPP* describe the processing of a unit of *Business* (conversation). These elements form a layered structure somewhat analogous to a layered communication model.

Process-Specification layer - The *Process-Specification* layer defines the heart of the *Business* agreement between the *Parties*: the services (*Business Transactions*) which *Parties* to the *CPA* can request of each other and transition rules that determine the order of requests..

Delivery Channels - A delivery channel describes a *Party's Message*-receiving characteristics. It consists of one document-exchange definition and one transport definition. Several delivery channels MAY be defined in one *CPP*.

Document-Exchange layer - The document-exchange layer accepts a *Business* document from the *Process-Specification* layer at one *Party*, encrypts it if specified, adds a digital signature for nonrepudiation if specified, and passes it to the transport layer for transmission to the other *Party*. It performs the inverse steps for received *Messages*. The options selected for the document-exchange layer are complementary to those selected for the transport layer. For example, if *Message* security is desired and the selected transport protocol does not provide *Message* encryption, then it must be specified at the document-exchange layer. The protocol for exchanging *Messages* between two *Parties* is defined by the ebXML *Message Service Specification*[ebMS] or other similar messaging service.

Transport layer - The transport layer is responsible for *Message* delivery using the selected transport protocol. The selected protocol affects the choices selected for the document-exchange layer. For example, some transport-layer protocols might provide encryption and authentication while others have no such facility.

It should be understood that the functional layers encompassed by the *CPP* have no understanding of the contents of the payload of the *Business* documents.

5.3.7.1 CollaborationProtocolProfile element

The **CollaborationProtocolProfile** element is the root element of the *CPP* XML document.

The **CollaborationProtocolProfile** element SHALL consist of the following child elements:

One or more REQUIRED **PartyInfo** elements that identify the organization (or parts of the organization) whose capabilities are described by the *CPP*,

One REQUIRED **Packaging** element,

Zero or one **ds:Signature** elements that contain the digital signature that signs the *CPP* document,

Zero or more **Comment** elements.

A *CPP* document MAY be digitally signed so as to provide for a means of ensuring that the document has not been altered (integrity) and to provide for a means of authenticating the author of the document. A digitally signed *CPP* SHALL be signed using technology that conforms to the joint W3C/IETF XML Digital Signature specification[XMLDSIG].

5.3.7.2 PartyInfo Element

The **PartyInfo** element identifies the organization whose capabilities are described in this *CPP* and includes all the details about this *Party*. More than one **PartyInfo** element MAY be provided in a *CPP* if the organization chooses to represent itself as subdivisions with different characteristics. Each of the subelements of **PartyInfo** is discussed later. The overall structure of the **PartyInfo** element is as follows:

<PartyInfo>



```
<PartyId type="..."> <!--one or more-->
    ...
</PartyId>
<PartyRef xlink:type="...", xlink:href="..."/>
<CollaborationRole> <!--one or more-->
    ...
</CollaborationRole>
<Certificate> <!--one or more-->
    ...
</Certificate>
<DeliveryChannel> <!--one or more-->
    ...
</DeliveryChannel>
<Transport> <!--one or more-->
    ...
</Transport>
<DocExchange> <!--one or more-->
    ...
</DocExchange>
</PartyInfo>
```

The **PartyInfo** element consists of the following child elements:

One or more REQUIRED **PartyId** elements that provide a logical identifier for the organization.

A REQUIRED **PartyRef** element that provides a pointer to more information about the *Party*.

One or more REQUIRED **CollaborationRole** elements that identify the roles that this *Party* can play in the context of a *Process Specification*.

One or more REQUIRED **Certificate** elements that identify the certificates used by this *Party* in security functions.

One or more REQUIRED **DeliveryChannel** elements that define the characteristics of each delivery channel that the *Party* can use to receive *Messages*. It includes both the transport level (e.g. HTTP) and the messaging protocol (e.g. ebXML *Message Service*).

One or more REQUIRED **Transport** elements that define the characteristics of the transport protocol(s) that the *Party* can support to receive *Messages*.

One or more REQUIRED **DocExchange** elements that define the *Message*-exchange characteristics, such as the *Message*-exchange protocol, that the *Party* can support.

PartyId element

The REQUIRED **PartyId** element provides a logical identifier that MAY be used to logically identify the *Party*. Additional **PartyId** elements MAY be present under the same **PartyInfo** element so as to provide for alternative logical identifiers for the *Party*. If the *Party* has preferences as to which logical identifier is used, the **PartyId** elements SHOULD be listed in order of preference starting with the most-preferred identifier.

In a *CPP* that contains multiple **PartyInfo** elements, different **PartyInfo** elements MAY contain **PartyId** elements that define different logical identifiers. This permits a large organization, for example, to have different identifiers for different purposes.

The value of the **PartyId** element is any string that provides a unique identifier. The identifier MAY be any identifier that is understood by both *Parties* to a *CPA*. Typically, the identifier would be listed in a well-known directory such as DUNS or in any naming system specified by [ISO6523].

PartyRef element

The **PartyRef** element provides a link, in the form of a URI, to additional information about the *Party*. Typically, this would be the URL from which the information can be obtained. The information might be at the *Party's* web site or in a publicly accessible repository such as an ebXML Registry, a UDDI repository, or an LDAP directory. Information available at that URI MAY include contact names, addresses, and phone numbers, and perhaps more information about the *Business Collaborations*



that the *Party* supports. This information MAY be in the form of an ebXML Core Component [ccOVER]. It is not within the scope of this specification to define the content or format of the information at that URI.

CollaborationRole element

The **CollaborationRole** element associates a *Party* with a specific role in the *Business Collaboration* that is defined in the *Process-Specification* document [ebBPSS]. Generally, the *Process Specification* is defined in terms of roles such as "buyer" and "seller". The association between a specific *Party* and the role(s) it is capable of fulfilling within the context of a *Process Specification* is defined in both the *CPP* and *CPA* documents. In a *CPP*, the **CollaborationRole** element identifies which role the *Party* is capable of playing in each *Process Specification* documents referenced by the *CPP*.

To indicate that the *Party* can play roles in more than one *Business Collaboration* or more than one role in a given *Business Collaboration*, the **PartyInfo** element SHALL contain more than one **CollaborationRole** element. Each **CollaborationRole** element SHALL contain the appropriate combination of **ProcessSpecification** element and **Role** element.

The **CollaborationRole** element SHALL consist of the following child elements: a REQUIRED **ProcessSpecification** element, a REQUIRED **Role** element, zero or one **CertificateRef** element, and one or more **ServiceBinding** elements. The **ProcessSpecification** element identifies the *Process-Specification* document that defines such role. The **Role** element identifies which role the *Party* is capable of supporting. The **CertificateRef** element identifies the certificate to be used. Each **ServiceBinding** element provides a binding of the role to a default **DeliveryChannel**. The default **DeliveryChannel** describes the receive properties of all *Message* traffic that is to be received by the *Party* within the context of the role in the identified *Process-Specification* document. Alternative **DeliveryChannels** MAY be specified for specific purposes, using **Override** elements as described below.

ProcessSpecification element

The **ProcessSpecification** element provides the link to the *Process-Specification* document that defines the interactions between the two *Parties*. It is RECOMMENDED that this *Business-Collaboration* description be prepared in accord with the ebXML Business Process Specification Schema [ebBPSS]. The *Process-Specification* document MAY be kept in an ebXML Registry.

The **ProcessSpecification** element has a single REQUIRED child element, **ds:Reference**, and the following attributes:

a REQUIRED **name** attribute, with type ID,

a REQUIRED **version** attribute,

a FIXED **xlink:type** attribute,

a REQUIRED **xlink:href** attribute.

The **ds:Reference** element relates to the **xlink:type** and **xlink:href** attributes as follows. Each **ProcessSpecification** element SHALL contain one **xlink:href** attribute and one **xlink:type** attribute with a value of "simple", and MAY contain one **ds:Reference** element formulated according to the XML Digital Signature specification [XMLDSIG]. In case the document is signed, it MUST use the **ds:Reference** element. When the **ds:Reference** element is present, it MUST include a **ds:URI** attribute whose value is identical to that of the **xlink:href** attribute in the enclosing **ProcessSpecification** element.

Role element

The REQUIRED **Role** element identifies which role in the *Process Specification* the *Party* is capable of supporting via the **ServiceBinding** element(s) siblings within this **CollaborationRole** element.

The **Role** element has the following attributes:

a REQUIRED **name** attribute,

a FIXED **xlink:type** attribute,



a REQUIRED *xlink:href* attribute.

ServiceBinding element

The **ServiceBinding** element identifies a default **DeliveryChannel** element for all of the *Message* traffic that is to be sent to the *Party* within the context of the identified *Process-Specification* document.

The **ServiceBinding** element SHALL have one child **Service** element and zero or more **Override** child elements.

The **ServiceBinding** element has the following attributes:

a REQUIRED *channelId* attribute,

a REQUIRED *packageId* attribute.

Service element

The value of the **Service** element is a string that SHALL be used as the value of the **Service** element in the ebXML *Message Header* [ebMS] or a similar element in the *Message Header* of an alternative *message service*. The **Service** element has an IMPLIED *type* attribute.

If the *Process-Specification* document is defined by the ebXML Business Process Specification Schema [ebBPSS], then the value of the **Service** element is an overall identifier for the set of *Business Transactions* associated with the authorized role corresponding to the role identified in the parent **CollaborationRole** element.

Override element

The **Override** element provides a *Party* with the ability to map, or bind, a different **DeliveryChannel** to *Messages* of a selected *Business Transaction* that are to be received by the *Party* within the context of the parent **ServiceBinding** element.

Each **Override** element SHALL specify a different **DeliveryChannel** for selected *Messages* that are to be received by the *Party* in the context of the *Process Specification* that is associated with the parent **ServiceBinding** element. The **Override** element has the following attributes:

a REQUIRED *action* attribute,

a REQUIRED *channelId* attribute,

a REQUIRED *packageId* attribute,

an IMPLIED *xlink:href* attribute,

a FIXED *xlink:type* attribute.

Certificate element

The **Certificate** element defines certificate information for use in this *CPP*. One or more **Certificate** elements MAY be provided for use in the various security functions in the *CPP*. An example of the **Certificate** element is:

The **Certificate** element has a single REQUIRED attribute: *certId*. The **Certificate** element has a single child element: *ds:KeyInfo*.

DeliveryChannel element

A delivery channel is a combination of a **Transport** element and a **DocExchange** element that describes the *Party's Message*-receiving characteristics. The *CPP* SHALL contain one or more **DeliveryChannel** elements, one or more **Transport** elements, and one or more **DocExchange** elements. Each delivery channel MAY refer to any combination of a **DocExchange** element and a **Transport** element. The same **DocExchange** element or the same **Transport** element MAY be referred to by more than one delivery channel. Two delivery channels MAY use the same transport protocol and the same document-exchange protocol and differ only in details such as communication addresses or security definitions. Figure 22 illustrates three delivery channels.

Three Delivery Channels

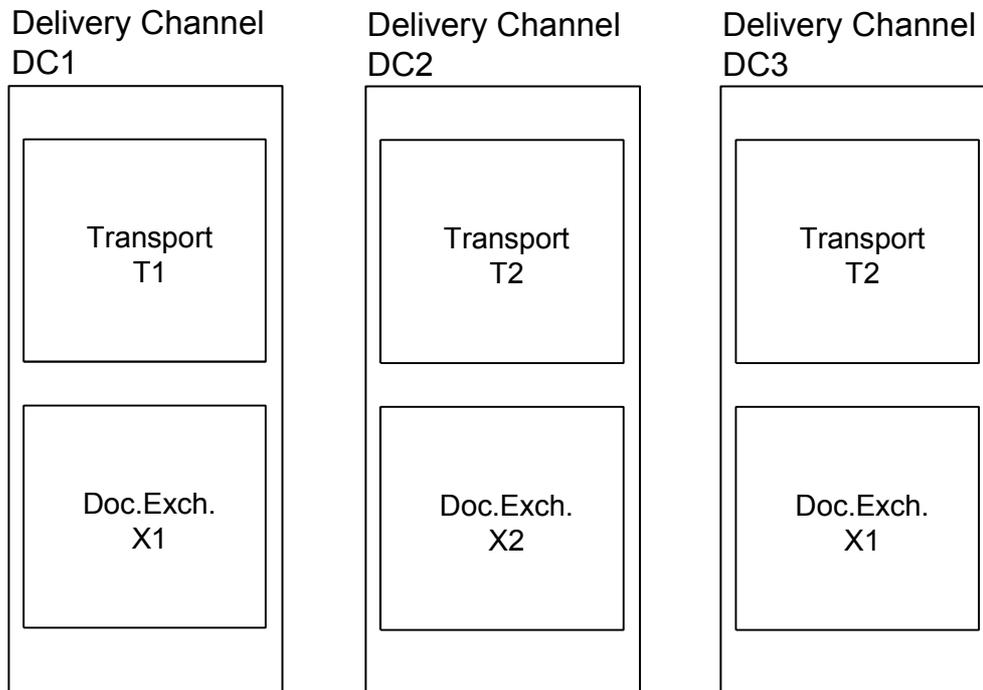


Figure 22: Delivery channels

The delivery channels have ID attributes with values "DC1", "DC2", and "DC3". Each delivery channel contains one transport definition and one document-exchange definition. Each transport definition and each document-exchange definition also has a name as shown in the figure. Note that delivery-channel DC3 illustrates that a delivery channel MAY refer to the same transport definition and document-exchange definition used by other delivery channels but a different combination. In this case delivery-channel DC3 is a combination of transport definition T2 (also referred to by delivery-channel DC2) and document-exchange definition X1 (also referred to by delivery-channel DC1).

Each **DeliveryChannel** element identifies one **Transport** element and one **DocExchange** element that make up a single delivery channel definition.

The **DeliveryChannel** element has the following attributes:

- a REQUIRED **channelId** attribute,
- a REQUIRED **transportId** attribute,
- a REQUIRED **docExchangeId** attribute.

The **DeliveryChannel** element has one REQUIRED child element, **Characteristics**.

Characteristics element

The **Characteristics** element describes the security characteristics and other attributes of the delivery channel. The attributes of the **Characteristics** element, except **syncReplyMode**, MAY be used to override the values of the corresponding attributes in the *Process-Specification* document.

The **Characteristics** element has the following attributes:

- An IMPLIED **syncReplyMode** attribute,
- an IMPLIED **nonrepudiationOfOrigin** attribute,
- an IMPLIED **nonrepudiationOfReceipt** attribute,



an IMPLIED **secureTransport** attribute,
an IMPLIED **confidentiality** attribute,
an IMPLIED **authenticated** attribute,
an IMPLIED **authorized** attribute.

Transport element

The **Transport** element of the *CPP* defines the *Party's* capabilities with regard to communication protocol, encoding, and transport security information.

The overall structure of the **Transport** element is as follows:

```
<Transport transportId = "N05">
  <!--protocols are HTTP, SMTP, and FTP-->
  <SendingProtocol version = "1.1">HTTP</SendingProtocol>
  <!--one or more SendingProtocol elements-->
  <ReceivingProtocol version = "1.1">HTTP</ReceivingProtocol>
  <!--one or more endpoints-->
  <Endpoint uri="http://example.com/servlet/ebxmlhandler"
    type = "request"/>
  <TransportSecurity> <!--0 or 1 times-->
    <Protocol version = "3.0">SSL</Protocol>
    <CertificateRef certId = "N03"/>
  </TransportSecurity>
</Transport>
```

Synchronous Responses

One distinguishing characteristic of transport protocols is whether a given transport protocol supports synchronous replies.

Transport protocol

Supported communication protocols are HTTP, SMTP, and FTP. The *CPP* MAY specify as many protocols as the *Party* is capable of supporting.

SendingProtocol element

The **SendingProtocol** element identifies the protocol that a *Party* can, or will, use to send *Business* data to its intended collaborator. The IMPLIED **version** attribute identifies the specific version of the protocol. For example, suppose that within a *CPP*, a **Transport** element, containing **SendingProtocol** elements whose values are SMTP and HTTP, is referenced within a **DeliveryChannel** element. Suppose, further, that this **DeliveryChannel** element is referenced for the role of Seller within a purchase-ordering process. Then the party is asserting that it can send purchase orders by either SMTP or HTTP. In a *CPP*, the **SendingProtocol** element MAY appear one or more times under each **Transport** element.

ReceivingProtocol element

The **ReceivingProtocol** element identifies the protocol by which a *Party* can receive its *Business* data from the other *Party*. The IMPLIED **version** attribute identifies the specific version of the protocol. For example, suppose that within a *CPP*, a **Transport** element is referenced within a **DeliveryChannel** element containing a **ReceivingProtocol** element whose value is HTTP. Suppose further that this **DeliveryChannel** element is referenced for the role of seller within a purchase ordering *Business Collaboration*. Then the party is asserting that it can receive *Business* responses to purchase orders over HTTP.

Within a *CPA*, the **SendingProtocol** and **ReceivingProtocol** elements serve to indicate the actual agreement upon what transports will be used for the complementary roles of the collaborators. For example, continuing the earlier examples, the seller in a purchase-order *Business Collaboration* could specify its receiving protocol to be SMTP and its sending protocol to be HTTP. These collaborator capabilities would match the buyer capabilities indicated in the *CPP*. These matches support an

interoperable transport agreement where the buyer would send purchase orders by SMTP and where the responses to purchase orders (acknowledgements, cancellations, or change requests, for example) would be sent by the seller to the buyer using HTTP.

To fully describe receiving transport capabilities, the receiving-protocol information needs to be combined with URLs that provide the endpoints (see below).

Endpoint element

The REQUIRED *uri* attribute of the **Endpoint** element specifies the *Party's* communication addressing information associated with the **ReceiveProtocol** element. One or more **Endpoint** elements SHALL be provided for each **Transport** element in order to provide different addresses for different purposes.

5.3.7.3 DocExchange Element

The **DocExchange** element provides information that the *Parties* must agree on regarding exchange of documents between them. This information includes the messaging service properties (e.g. ebXML Message Service [ebMS]).

The **DocExchange** element of the *CPP* defines the properties of the messaging service to be used with *CPAs* composed from the *CPP*.

docExchangeld attribute

The **DocExchange** element has a single IMPLIED **docExchangeld** attribute that is an [XML] ID that provides a unique identifier that MAY be referenced from elsewhere within the *CPP* document.

ebXMLBinding element

The **ebXMLBinding** element describes properties specific to the ebXML Message Service [ebMS]. The **ebXMLBinding** element is comprised of the following child elements:

zero or one **ReliableMessaging** element which specifies the characteristics of reliable messaging,

zero or one **NonRepudiation** element which specifies the requirements for signing the *Message*,

zero or one **DigitalEnvelope** element which specifies the requirements for encryption by the digital-envelope [DIGENV] method,

zero or more **NamespaceSupported** elements that identify any namespace extensions supported by the messaging service implementation.

5.3.8 The Registry/Repository

The ebXML registry/repository is a service that stores CPP's, CPA's, ebXML core components and any other ebXML documents or fragments including WSDL documents, JAR files with Java code and even audio and video etc.

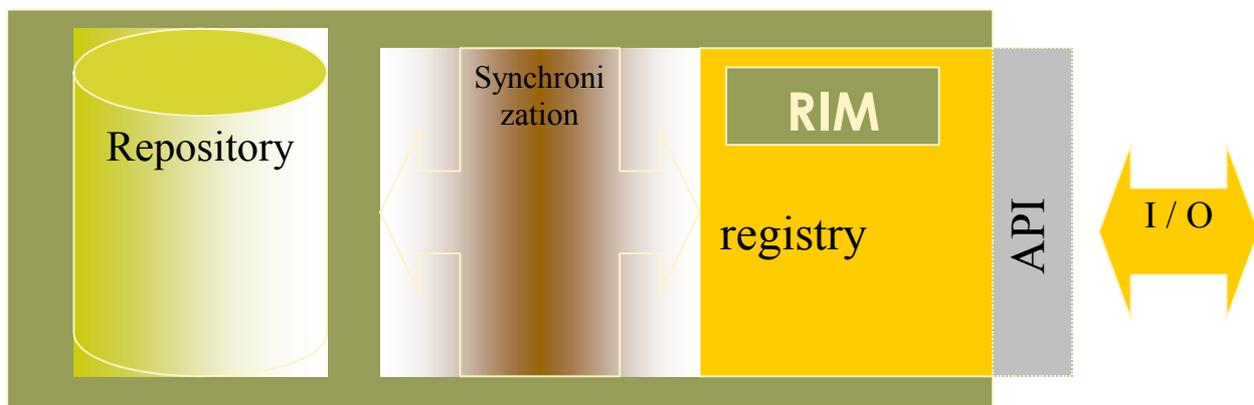


Figure 23: ebXML registry/repository



5.3.8.1 Role of ebXML Registry

The *Registry* provides a stable store where information submitted by a *Submitting Organization* is made persistent. Such information is used to facilitate ebXML-based *Business to Business* (B2B) partnerships and transactions. Submitted content may be XML schema and documents, process descriptions, *Core Components*, context descriptions, UML models, information about parties and even software components.

5.3.8.2 Registry Services

A set of *Registry Services* that provide access to *Registry* content to clients of the *Registry* is defined in the ebXML Registry Services Specification [ebRS]. This document does not provide details on these services but may occasionally refer to them.

5.3.8.3 What the Registry Information Model Does

The Registry Information Model provides a blueprint or high-level schema for the ebXML *Registry*. Its primary value is for implementers of ebXML *Registries*. It provides these implementers with information on the type of metadata that is stored in the *Registry* as well as the relationships among metadata *Classes*.

The Registry information model:

- Defines what types of objects are stored in the *Registry*
- Defines how stored objects are organized in the *Registry*
- Is based on ebXML metamodels from various working groups

5.3.8.4 How the Registry Information Model Works

Implementers of the ebXML *Registry* MAY use the information model to determine which *Classes* to include in their *Registry Implementation* and what attributes and methods these *Classes* may have. They MAY also use it to determine what sort of database schema their *Registry Implementation* may need.

5.3.8.5 Where the Registry Information Model May Be Implemented

The Registry Information Model MAY be implemented within an ebXML *Registry* in the form of a relational database schema, object database schema or some other physical schema. It MAY also be implemented as interfaces and *Classes* within a *Registry Implementation*.

5.3.8.6 Conformance to an ebXML Registry

If an *Implementation* claims *Conformance* to this specification then it supports all required information model *Classes* and interfaces, their attributes and their semantic definitions that are visible through the ebXML *Registry Services*.

5.4 Regnet connector

As described previously, in order to set up the REGNET connector we need both:

- EbXML messaging in order to implement transaction between partners.
- EbXML registry in order to store CSCs characteristics, CPP and CPA.

5.4.1 Implementing an ebXML messaging service

An implementation of the ebXML messaging service has been developed based on JAXM.

The Java™ API for XML Messaging (JAXM) Optional Package enables applications to send and receive document oriented XML messages using a pure Java API. JAXM implements Simple Object Access Protocol (SOAP) 1.1 with Attachments messaging so that developers can focus on building, sending, receiving, and decomposing messages for their applications instead of programming low level XML communications routines.



This version of the JAXM specification includes the notion of messaging Profiles. The intent is to establish a foundation for supporting a family of higher level standards-based messaging protocols. An example of a Profile would be an implementation of ebXML Transportation, Routing, and Packaging Message Handling Service or the W3C's XMLP layered on JAXM.

5.4.2 EbXML message example

This example illustrate communication between two museums.

Here is a message that are exchanged by the museums in the example:

```
--519137.1019573098130.JavaMail.administrator.regnetwk
Content-Type: text/xml

<Envelope xmlns="http://schemas.xmlsoap.org/soap/envelope/">
  <Header>
    <eb:MessageHeader xmlns:eb="http://www.ebxml.org/namespaces/messageHeader"
      mustUnderstand="1" version="1.0">
      <eb:From>
        <eb:PartyId>someone@museum2.com</eb:PartyId>
      </eb:From>
      <eb:To>
        <eb:PartyId>someone@museum1.com</eb:PartyId>
      </eb:To>
      <eb:CPAId>534</eb:CPAId>
      <eb:ConversationId>3246653</eb:ConversationId>
      <eb:Service>urn:services:ReceiveQuery</eb:Service>
      <eb:Action>ProcessQuery</eb:Action>
      <eb:MessageData>
        <eb:MessageId>1019573097499</eb:MessageId>
        <eb:Timestamp>Tue Apr 23 15:44:57 GMT+01:00 2002</eb:Timestamp>
      </eb:MessageData>
    </eb:MessageHeader>
  </Header>

  <Body>
    <eb:Manifest xmlns:eb="http://www.ebxml.org/namespaces/messageHeader">
      <eb:Reference xmlns:xlink="http://www.w3.org/1999/xlink"
        xlink:href="someone@museum1.com" xlink:type="simple">
        <eb:Description>Object Information</eb:Description>
      </eb:Reference>
    </eb:Manifest>
  </Body>
</Envelope>

--519137.1019573098130.JavaMail.administrator.regnetwk
Content-Type: text/plain; charset=ISO-8859-1
Content-Id: someone@museum1.com

<?xml version="1.0" encoding="UTF-8"?>

<MuseumObject:ObjectInfo
xmlns:MuseumObject="http://www.museum1.com/namespaces/object">

<MuseumObject:Name>La joconde</MuseumObject:Name>

<MuseumObject:ID>LOUVRE256</MuseumObject:ID>

<MuseumObject:Description>The famous painting from
Vinci</MuseumObject:Description>

</MuseumObject:ObjectInfo>
```

--519137.1019573098130.JavaMail.administrator.regnetwk--

Here we retrieve all the header elements presented on the ebMS chapter. The different ID's have been chosen arbitrarily, but in a real case implementation the proprietary application should take care of generating them.

Let's note that the SOAP body contains information about the ebXML payload. The link between the payload and header information is made via an xlink element. Note that the xlink points to someone@museum1.com, which is the content-id of the first SOAP attachment.

5.4.2.1 Architecture of the example:

The example involves 2 Servlets communicating via the SOAP protocol. A JSP is used to illustrate the client side.

The IDE used to implement this example is Borland JBuilder 4.0, and it's native application server, Tomcat 3.1.

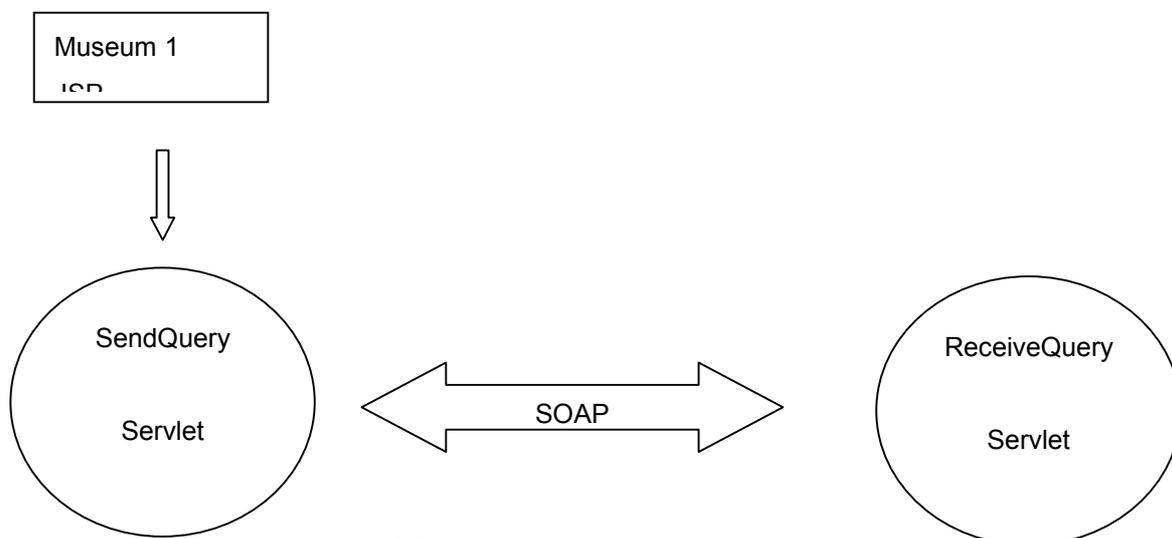


Figure 24: ebXML message example: architecture

5.4.3 Modelling business process

One of the primary benefits of ebXML is to improve e-business capabilities of SMEs, focusing on interoperability. We can define interoperability by the way two companies with different way of working (business processes) can harmonize their differences and conduct business together.

Thus, BPSS (business process specification schemas) are a key-issue to address when one intends to implement an ebXML compatible system.

It is important to understand that BPSS can also be used independently of the ebXML framework to capture and communicate business processes in a way that can be understood, without ambiguity, by both involved parties.

The business process used to demonstrate the steps of modelling a BPSS is based on the activities required by a museum that intends to borrow an object to another museum (for an exhibition purpose, for instance). As the BPSS will be used in the ebXML context, it has to express collaboration between two business entities, i.e. to show the different business activities and their choreography involved in the process. That is why the BPSS will describe the exchanges made between the borrower and its counterpart, the lender.

The business processes used here come from previous work-packages, in particular tasks dealing with business processes re-engineering.

5.4.3.1 Use Case



Use case “Borrow an object”

A museum borrows an object for a specific period of time and for a specified purpose, normally display, but including research, education and photography.

Sometimes a collector or an institution asks museum for keeping his cultural objects.

During this period the museum is responsible for the object.

Name: Borrow an object
Description: The museum borrows an object for a certain purpose from another institution/person
Actors: Head of Collections, Lender, Collection manger, Conservator, Photographer, Registrar

Actor action	System response
Head of Collections writes loan request to Lender	
If Lender agrees Collection manager requests more info about object and any requirements	
Collection manager records object loan information	System generates new and unique loan-in number and registers object loan information
An loan agreement is drawn up and signed by Lender and Head of Collections	
Collection manager prepares and schedules for receipt of object and identifies a location	System registers location of object
proceed to use case 7.1.6B (receive and object)	
If necessary Collection manager extends loan	System registers new loan end date
Collection manager prepares for the return of loan	
Conservator completes final condition check and Photographer photographs object if necessary	System registers object condition information

Use case “Lend out an object”

An object is lent to another institution for a specific period of time and for a specific purpose.

Use case description 7.1.3A

Name: Lend out an object
Description: An object of the museum is lend to another institution
Actors: Collection manager, Borrower, Requester
Type:
Cross references:
Preconditions: object cannot be loaned



Alternatives:

Actor action	System response
Collection manager receives written request from Borrower	
Collection manager records request information	System registers request information
Collection manager checks possibility to lend out object	System displays availability
Collection manager sends rejection to Requester with explanatory reason	

Use case description 7.1.3BA

Name: Lend out an object
 Description: An object of the museum is lend to another institution
 Actors: Collection manager, Borrower, Requester
 Type:
 Cross references:
 Preconditions: object is refused for loan
 Alternatives:

Actor action	System response
Collection manager receives written request from Borrower	
Collection manager records request information	System registers request information
Collection manager checks possibility to lend out object	System displays availability
Collection manager sends conditions of loan to Borrower	
Collection manager decides to loan or not taking into account time schedule, charges, suitability of Borrower, etc.	
Collection manager sends rejection to Requester with explanatory reason	

Use case description 7.1.3BB

Name: Lend out an object
 Description: An object of the museum is lend to another institution
 Actors: Collection manager, Borrower, Requester, Head of Collections, Registrar, Conservator, Photographer



Type:

Cross references:

Preconditions: object can be loaned

Alternatives:

Actor action	System response
Collection manager receives written request from Borrower	
Collection manager records request information	System registers request information
Collection manager checks possibility to lend out object	System displays availability
Collection manager sends conditions of loan to Borrower	
Collection manager decides to loan or not taking into account time schedule, charges, suitability of Borrower, etc.	
Collection manager notifies Borrower of decision to loan and provides object loan information to Borrower	System generates object loan information report
Head of Collections and Borrower confirm loan and sign loan agreement	
Collection manager prepares loan, insurance, packing for shipping, if needed Conservator does conservation work, and Photographer takes photograph	System registers information in loan, insurance, shipping, conservation and photograph
Collection manager makes arrangements for return of the object	System registers return arrangements
Collection manager ensures loan conditions have been met	
Registrar updates loan/exhibition information	System registers loan/exhibition information

5.4.3.2 Defining the business transactions

Out of these use cases we can already define the business transaction activities that will occur during the process:

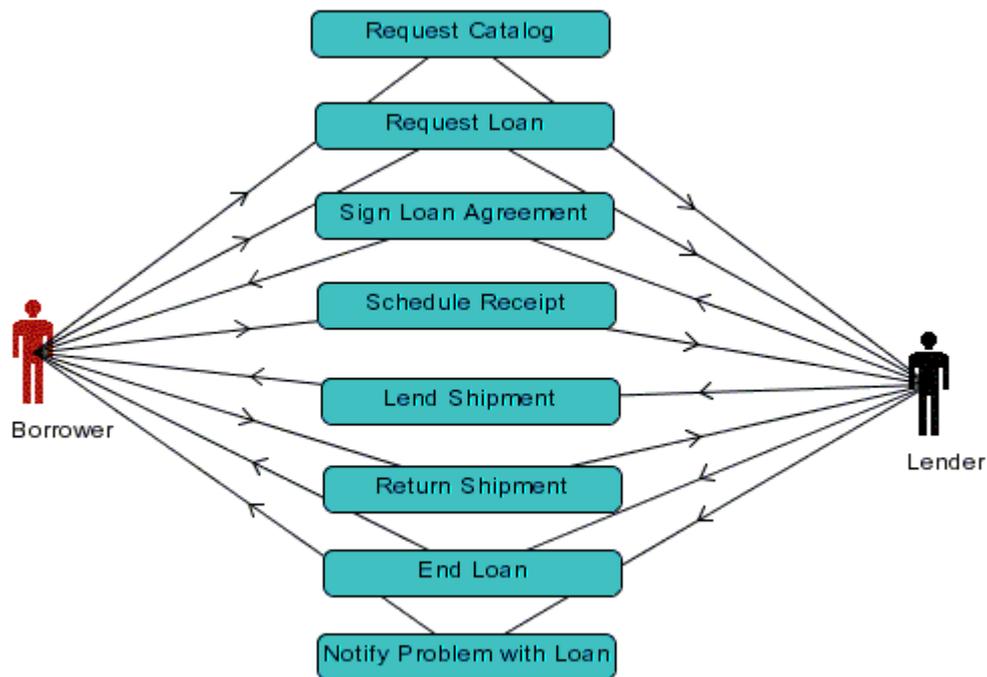


Figure 25: Business transactions scheme

It is important to note that at this point there is no reference to any kind of synchronization (or choreography). With each of these business transaction activities is associated a business transaction. Let's remind that each business transaction expresses an exchange of business documents.

The next step is so to define these business documents (two by business transaction, shown in *italic*) :

Request Catalog business transaction:

- Request : *Catalog request*
- Response : *Catalog*

Request Loan business transaction:

- Request : *Loan request*
- Response : *Loan acceptance*

Loan Agreement business transaction:

- Request : *Loan agreement*
- Response : *Loan agreement*

Receipt Schedule business transaction:

- Request : *Object receipt schedule*
- Response : *Schedule acceptance*

Shipment business transaction:

- Request : *Object sent notification*
- Response : *Object received notification*

End Loan business transaction:

- Request : *End loan ACK*
- Response : *End loan ACK*

Loan Problem business transaction:

- Request : *Loan problem description*
- Response : *Loan problem ACK*

The same business document can be used several times by the same, or different business transactions. Here it is the case for the Loan agreement business document, which is used twice by the Loan agreement business transaction.

5.4.3.3 Defining the synchronisation

We now have define what the business process is able to, how it can perform some messages exchanges (through business documents), but did not specify in what order can this transaction be processed.

It is time to add some sequentiality between all our business transaction activities.

Once again we can use the previously defined use cases to deduce Figure 26:

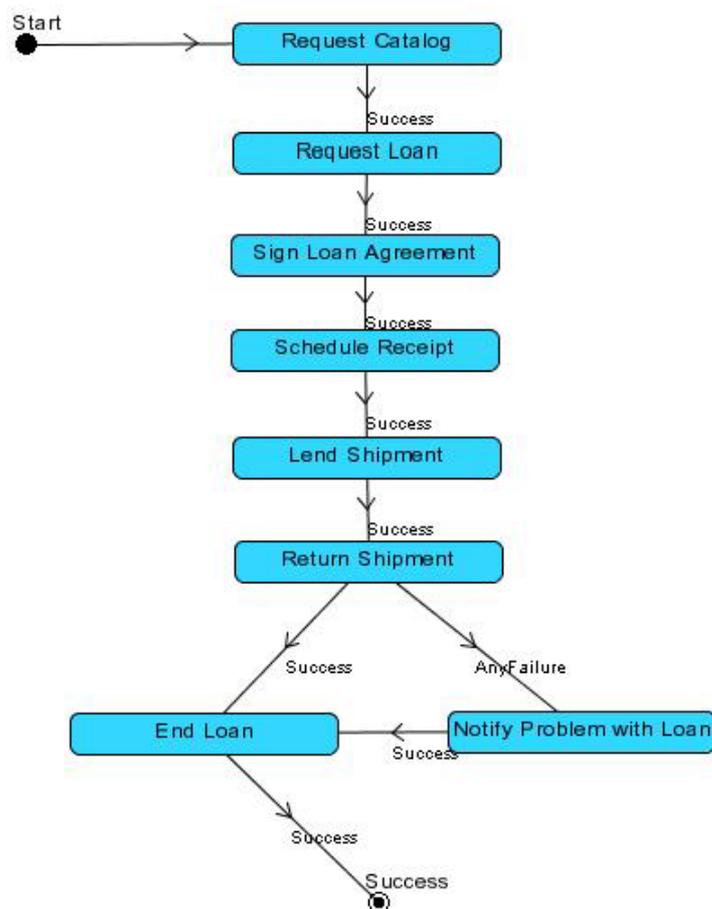


Figure 26: Use case diagram

For a better understanding purpose, this diagram is voluntary simplistic. But we could easy add a couple of failure tests in order to give it a more “realistic look”.

5.4.3.4 XML translation



It is now time to translate the work done into some XML code.

The XML schema specification for BPS can be found at: <http://www.ebxml.org/specs/ebBPSS.xsd>

And the dtd at: <http://www.ebxml.org/specs/ebBPSS.dtd>

It is recommended to use them in order to ensure the validity of the XML file to be produced.

The translation can be found in the annexes and is done accordingly with the XML elements described previously.

5.4.4 Installing the ebXML registry/repository

At the core of any ebXML compliant system is the registry/repository.

Main products available was:

- GoXML Registry from XMLGlobal (<http://www.xmlglobal.com>). GoXML™ Registry centralizes and organizes your business documents, processes, and services, then shares that information to enable business process integration. Its Lifecycle Manager controls the submission, classification, association, deprecation, and removal of registry items. A Query Manager permits discovery and retrieval of the registry's contents. A Repository Manager handles the physical storage and safekeeping of registry objects. A User Manager controls who can use the registry and their privileges. The ebXML Message Service module provides secure and reliable delivery of ebXML messages. A Web-based Administrator Workbench makes content management both intuitive and convenient. Web Services and ebXML interfaces let you plug into popular e-business infrastructures and provide superior XML interoperability. The main drawback of this tool is its price (around 35 000 \$).
- EbXMLrr from source forge (<http://ebxmlrr.sourceforge.net/>). The goal of the ebxmlrr project is to deliver a functionally complete reference implementation for the OASIS ebXML Registry V2.0 specification. As far as this tool is available freely with open source licence, we decided to use it.

5.4.4.1 EbXMLRR

EbXMLRR (<http://ebxmlrr.sourceforge.net/>) is based on Sun's try to develop an ebXML compliant registry/repository. When the specifications for the registry where passed to version 2.0, Sun stopped the development and released the sources of its product.

It has then been overtaken by a group of developers that extended it so that it can fit v2 specifications.

In order to install the product, the first task is to download all the source files. This can be done, using cvs. More information is found at https://sourceforge.net/cvs/?group_id=37074

It is then necessary to install java 1.3 (to compile the files), and tomcat 1.3.

Ant files are given, so it is wise to install this software as well to simplify the installation.

Example of interaction with registry is:

Query:

```
<?xml version = "1.0" encoding = "UTF-8"?>
<AdhocQueryRequest xmlns = "urn:oasis:names:tc:ebxml-regrep:query:xsd:2.0"
xmlns:xsi = "http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation
= "urn:oasis:names:tc:ebxml-regrep:query:xsd:2.0 query.xsd">
  <ResponseOption returnType = "LeafClassWithRepositoryItem"
returnComposedObjects="true"/>
  <SQLQuery>
    SELECT id FROM Organization o, Name nm WHERE nm.value LIKE '%Sun%'
AND o.id = nm.parent
  </SQLQuery>
</AdhocQueryRequest>
```

Answer:



```
<soap-env:Envelope
  xmlns:soap-env="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:SOAP-SEC="http://schemas.xmlsoap.org/soap/security/2000-12" soap-
env:actor="some-uri"
  soap-env:mustUnderstand="1">
  <soap-env:Header>
  </soap-env:Header>
  <soap-env:Body SOAP-SEC:id="Body">
    <RegistryResponse status="Success" xmlns="urn:oasis:names:tc:ebxml-
regrep:registry:xsd:2.1">
      <AdhocQueryResponse xmlns="urn:oasis:names:tc:ebxml-
regrep:query:xsd:2.1">
        <SQLQueryResult>
          ...
        </SQLQueryResult>
      </AdhocQueryResponse>
    </RegistryResponse>
  </soap-env:Body>
</soap-env:Envelope>
```

5.5 EbXML Glossary

TERM	ACRONYM	DEFINITION	SOURCE
ABSTRACT CLASS		A class that cannot be directly instantiated.	UML Glossary V1.3
ABSTRACTION		The essential characteristics of an entity that distinguish it from all other kinds of entities.	UML Glossary V1.3
ACTIVE CLASS		A class whose instances are active objects.	UML Glossary V1.3
ACTOR		Someone or something, outside the system or business that interacts with the system or business.	Rational Unified Process
AGGREGATE [CLASS]		A class that represents the "whole" in an aggregation (whole-part) relationship.	UML Glossary V1.3
AGGREGATE CORE COMPONENT		Defines a functional unit representation form that contains embedded information entities	CC/ Core Component Terminology
AGGREGATION		A special form of association that specifies a whole-part relationship between the aggregate (whole) and a component part.	UML Glossary V1.3
AGREEMENT		An arrangement between two partner types that specifies in advance the conditions under which they will trade (terms of shipment, terms of payment, collaboration protocols, etc.) An agreement does not imply specific economic commitments.	BP team Terminology



APPLICATION		An Application is software that may implement a Service by processing one or more of the Messages in the Document Exchanges associated with the Service.	WebXML Glossary
ARCHITECTURE		The architecture of a software system (at a given point in time) is its organization or structure of significant components interacting through interfaces	Rational Unified Process
AUTHORISATION		A right or a permission that is granted to a system entity to access a system resource.	IETF RFC 2828
AUTHORISATION PROCESS		A procedure for granting authorisation	IETF RFC 2828
BUSINESS ENTITY		Something that is accessed, inspected, manipulated, produced, and worked on in the business.	UMM
BASIC INFORMATION ENTITY		Defines a component which contains data but which does not have embedded information entities.	CC/ Core Component Terminology
BEHAVIOUR		The observable effects of an operation or event, including its results.	UML Glossary V1.3
BUSINESS		A series of processes, each having a clearly understood purpose, involving more than one organization, realized through the exchange of information and directed towards some mutually agreed upon goal, extending over a period of time.	(Open-edition Reference Model Standard - ISO/IEC 14662). (MoU)
BUSINESS LIBRARY		A repository of business process specifications and business information objects within an industry, and of common business process specifications and common business information objects that are shared by multiple industries.	TA Specification Terminology
BUSINESS ACTIVITY		A business activity is used to represent the state of the business process of one of the partners.	BP Team Terminology
BUSINESS COLLABORATION		An activity conducted between two or more parties for the purpose of achieving a specified outcome.	TA Specification Terminology
BUSINESS COLLABORATION KNOWLEDGE		The knowledge involved in a collaboration	TA Specification Terminology
BUSINESS CONTEXT		Defines a context in which a business has chosen to employ an information	CC/ Core Component



CONTEXT		has chosen to employ an information entity	Component Terminology
BUSINESS DOCUMENT		The set of information components that are interchanged as part of a business activity.	CC/ Core Component Terminology
BUSINESS ENTITY		Something that is accessed, inspected, manipulated, produced, and worked on in the business.	UMM
BUSINESS INFORMATION GROUP		A set of basic and/or aggregate information entities that convey a single business function.	CC/ Core Component Terminology
BUSINESS OPERATIONAL VIEW (BOV)	BOV	A perspective of business transactions limited to those aspects regarding the making of business decisions and commitments among organisations, which are needed for the description of a business transaction.	TA Specification Terminology
BUSINESS PARTNER		An entity that engages in business transactions with another business partner(s).	
BUSINESS PROCESS		The means by which one or more activities are accomplished in operating business practices.	UMM
BUSINESS PROCESS INTERFACE		The definition of how to interact with one partner role in order to make partner perform a desired service.	BP Team Terminology
BUSINESS PROCESS SPECIFICATION SCHEMA		Defines the necessary set of elements to specify run-time aspects and configuration parameters to drive the partners' systems used in the collaboration.	BP Team Terminology
BUSINESS PROFILE		Describes a company's ebXML capabilities and constraints, as well as its supported business scenarios.	
BUSINESS RULE		Rules, regulations and practices for business.	UMM
BUSINESS SERVICE INTERFACE		An ebXML collaboration that is conducted by two or more parties each using a human or automated business service that interprets the documents and document envelopes transmitted and decides how to (or whether to) respond.	BP Team Terminology



BUSINESS TRANSACTION		A business transaction is a logical unit of business conducted by two or more parties that generates a computable success or failure state.	BP Team Terminology
CHOREOGRAPHY		A declaration of the activities within collaboration and the sequencing rules and dependencies between these activities.	
CLASS		A description of a set of objects that share the same attributes, operations, methods, relationships, and semantics.	Rational Unified Process
CLASS DIAGRAM		A graphical representation that shows static structure of concepts, types, and classes.	UML Glossary V1.3
CODE		A character string (letters, figures or symbols) that for brevity and / or language independency may be used to represent or replace a definitive value or text of an attribute.	ebXML CC Dictionary Naming Conventions
COLLABORATION		Describes a pattern of interaction among objects; it shows the objects participating in the interaction by their links to each other and the messages they send to each other.	Rational Unified Process Terminology
COLLABORATION DIAGRAM		A graphical representation of collaboration.	Rational Unified Process Terminology
COLLABORATION PROTOCOL		The protocol that defines for a Collaborative Process:	CPA Specification Terminology



COLLABORATION PROTOCOL AGREEMENT	CPA	Information agreed between two (or more) Parties that identifies or describes the specific Collaboration Protocol that they have agreed to use.	CPA Specification Terminology
COLLABORATION PROTOCOL PROFILE	CPP	Information about a Party that can be used to describe one or more Collaborative Processes and associated Collaborative Protocols that the Party supports.	CPA Specification Terminology
COLLABORATIVE PROCESS		A shared process by which two Parties work together in order to carry out a process.	CPA Specification Terminology
COMMITMENT		An obligation to perform an economic event (that is, transfer ownership of a specified quantity of a specified economic resource type) at some future point in time. Order line items are examples of commitments.	BP Team Terminology
COMMON BUSINESS LIBRARY	CBL		
COMMON BUSINESS PROCESS		A business process that is used with reasonable frequency in a business community.	BP Team Terminology
COMMUNICATION PROTOCOL ENVELOPE		The outermost envelope of an ebXML Message.	Messaging Service Specification Terminology
CONCRETE CLASS		A class that can be directly instantiated.	UML Glossary V1.3
CONFORMANCE		Fulfilment of a product, process or service of all requirements specified; adherence of an implementation to the requirements of one or more specific standards or technical specifications.	ISO Guide 2
CONSTRAINT		A condition or a restriction.	UMM
CONTROLLING AGENCY		Agency responsible for controlling the content of a basic information entity	CC/ Core Component Terminology



CORE COMPONENT		Generic term that covers Core Component Type, Aggregate Information Entity and Basic Information Entity.	CC/ Core Component Terminology
CORE COMPONENT TYPE		Any Core Component that has no business meaning on its own. When they are reused in a business context, Core Component Types become Basic Information Entities.	CC/ Core Component Terminology
CORE LIBRARY		Contains data and process definitions, including relationships and cross references, as expressed in business terminology that may be tied to accepted industry classification scheme or taxonomy.	
DATA TYPE		A type of data to be used to represent the content of an information entity.	CC/ Core Component Terminology
DIGITAL SIGNATURE		A digital code that can be attached to an electronically transmitted message that uniquely identifies the sender	Digital Signature Scheme ISO 9796
DISTRIBUTED REGISTRY		Federation of multiple registries that behaves logically as one registry.	TA Specification Terminology
DOCUMENT TYPE DEFINITION	DTD	Allows different instances of documents of the same type to be automatically processed in a uniform way.	XMI Glossary: OMG
DOMAIN		A district or area under someone's control, range of influence.	CC/ Core Component Terminology
ebXML INFRASTRUCTURE		The full compliment of technical specifications encompassed within the ebXML framework.	Technical Architecture terminology
ECONOMIC CONTRACT		A subtype of agreement between partner types that some actual economic exchanges will occur in the future.	BP Team Terminology
ECONOMIC EVENT		The transfer of control of an economic resource from one party to another party.	BP Team Terminology
ECONOMIC RESOURCE		A quantity of something of value that is under the control of an enterprise.	BP Team Terminology



ECONOMIC RESOURCE TYPE		An economic resource type is the abstract classification or definition of an economic resource.	BP Terminology Team
EDIFACT WORKING GROUP	UN/EDWG	To develop and maintain UN/EDIFACT support of harmonised implementations and the use of multi-lingual terminology.	UN/EDWG
ELECTRONIC BUSINESS	eBusiness	A generic term covering information definition and exchange requirements within and between enterprises by electronic means	(MoU)
electronic business XML	ebXML		
ELECTRONIC COMMERCE		Electronic Commerce is doing business electronically. This includes the sharing of standardised unstructured or structured business information by any electronic means.	UN/CEFACT SIMAC
ELECTRONIC DATA INTERCHANGE	EDI	The automated exchange of any predefined and structured data for business among information systems of two or more organizations.	(Open-edi Reference Model - Standard ISO/IEC 14662). (MoU)
ELEMENT		An atomic constituent of a model.	UML Glossary V1.3
ENCRYPTION		Cryptographic transformation of data (called "plaintext") into a form (called "ciphertext") that conceals the data's original meaning to prevent it from being known or used.	ETF RFC 2828
EXTENSIBLE MARKUP LANGUAGE	XML	XML is designed to enable the exchange of information (data) between different applications and data sources on the World Wide Web and has been standardized by the W3C.	UN/CEFACT SIMAC
FUNCTIONAL SERVICES VIEW	FSV	A perspective of business transactions limited to those information technology interoperability aspects of IT systems needed to support the execution of open-edi transactions.	Open-edi Reference Model, ISO/IEC 14662
FUNCTIONAL SET		A set of alternative representations for the same semantic concept.	CC/ Core Component Terminology
IMPLEMENTATION		An implementation is the realization of a specification.	NIST



INHERITANCE		The mechanism by which more specific elements incorporate structure and behaviour of more general elements related by behaviour.	Rational Unified Process Terminology
INSTANCE		An entity to which a set of operations can be applied and which has a state that stores the effects of the operations.	Rational Unified Process Terminology
INTERACTION DIAGRAM		Shows how several objects collaborate in single use case.	UML Glossary V1.3
MESSAGE ENVELOPE		A communication independent envelope, specifically MIME multipart/related, which contains the two main parts of an ebXML compliant message (the Header and Payload containers).	Messaging Service Specification Terminology
MESSAGE HEADER		A specification of the structure and composition of the information necessary for an ebXML Messaging Service to successfully generate or process and ebXML compliant message.	Messaging Service Specification Terminology
MESSAGING CAPABILITIES		The set of capabilities that support exchange of Documents between Parties.	Messaging Service Specification Terminology
MESSAGING SERVICE		A framework that enables interoperable, secure and reliable exchange of Messages between Trading Partners.	Messaging Service Specification Terminology
MESSAGING SERVICE LAYER		Enforces the "rules of engagement" as defined by two Trading Partners in a Collaboration Protocol Agreement (including, but not limited to security and Business Process functions related to Message delivery).	TA Specification Terminology
METHOD		The detailed, logically ordered plans or procedures followed to accomplish a task or attain a goal.	Rational Unified Process Terminology
OPEN-EDI		Electronic data interchange among multiple autonomous organizations to accomplish an explicit shared business goal.	(MoU) Check reference.
PACKAGE		A general-purpose mechanism for organizing elements into groups.	Rational Unified Process Terminology
PACKAGE DIAGRAM		Shows groups of classes and dependencies among them.	UML Glossary V1.3



PARTY		A Party is an entity such as a company, department, organisation or individual that can generate, send, receive or relay Documents.	CPP & CPA Specification Terminology
PARTY DISCOVERY PROCESS		A Collaborative Process by which one Party can discover CPP information about other Parties.	CPA Specification Terminology
PAYLOAD		A section of data/information that is not part of the ebXML wrapping.	Messaging Service Specification Terminology
PAYLOAD CONTAINER		An optional container used to envelope the real payload of an ebXML message.	Messaging Service Specification Terminology
PAYLOAD ENVELOPE		The specific MIME headers that are associated with a MIME part.	Messaging Service Specification
REGISTRY		A mechanism whereby relevant repository items and metadata about them can be registered such that a pointer to their location, and all their metadata, can be retrieved as a result of a query.	TA Specification Terminology
REGISTRY AUTHORITY		A super user who maintains registry.	Defined in ISO11179
REGISTRY CLIENTS		An ebXML application that makes use of services offered by a Registry using the messaging services.	TA Specification Terminology
REGISTRY ENTRY		Metadata that catalogs registry item.	
REGISTRY INFRASTRUCTURE PROVIDER		An entity which provides a registry/repository to store profiles, CPPs etc.	
REGISTRY INTERFACE		A set of Registry Services that provide access to Registry content to clients of the Registry is defined in the ebXML Registry Services Specification.	TA Specification Terminology
REGISTRY ITEM		The content registered in a repository.	TA Specification Terminology
REGISTRY SERVICE		A way of providing access to Registry content to clients of the Registry.	ebXML Registry Services Specification 222
REPOSITORY		A location or set of distributed locations where Repository Items, pointed at by the registry, reside and from which they can be retrieved.	TA Specification Terminology



REPRESENTATION TYPE		Type of data to be used to represent the content of an information entity	CC/ Core Component Terminology
ROLE		The named specific behaviour of an entity participating in a particular context.	UML Glossary V1.3
SCENARIO		A formal specification of a class of business activities having the same business goal.	(ISO 19735 part)
SECURITY MODEL		A schematic description of a set of entities and relationships by which a specified set of security services are provided by or within a system.	IETF RFC 2828
SECURITY POLICY		A set of rules and practices that specify or regulate how a system or organization provides security services to protect sensitive and critical system resources.	IETF RFC 2828
SEQUENCE DIAGRAM		A diagram that shows object interactions arranged in time sequence.	Rational Unified Process Terminology
SIMPLE ELECTRONIC BUSINESS (SEB)	SEB	Simple Electronic Business is the application of simplified business processes, using core application data, and new and existing standardised techniques that support paperless and efficient operations.	UN/CEFACT SIMAC
SIMPL-EDI		Subsets of UN/EDIFACT messages especially designed for SMEs. Simpl-EDI - Simple Electronic Business defines simplest processes and their required core data allowing the exchange of the minimum data to effect a business transaction electronically.	UN/CEFACT SIMAC
SPECIFICATION SCHEMA		An additional view of a meta model.	
SUBMITTING ORGANISATION		Any organisation that submits a repository item to be registered in a repository.	OASIS
SUPPLY CHAIN		A sequence of events, which may include conversion, movement or placement, which adds value to goods, products, or services.	UN/CEFACT SIMAC
UNIFIED MODELLING LANGUAGE (UML)	UML	A set of diagrams that communicate requirements regarding a business process.	



UNIQUE IDENTIFIER	UID	The abstract concept of utilizing a standard mechanism and process for assigning a sequence of alphanumeric codes to ebXML Registry items, including: Core Components, Aggregate Information Entities, and Business Processes	
UNIVERSALLY UNIQUE IDENTIFIER (UUID)	UUID	An identifier that is unique across both space and time, with respect to the space of all UUIDs.	DCE 1.1: Remote Procedure Call. Open Group Technical Standard. Document Number C706. The Open Group (Reading, UK: August, 1997).
USE CASE		Defines a sequence of actions a system performs that yields an observable result of value to a particular actor.	UML
USE CASE MODEL		A model that describes a system's functional requirements in terms of use cases.	UML Glossary V1.3
VULNERABILITY		A flaw or weakness in a system's design, implementation, or operation and management that could be exploited to violate the system's security policy.	IETF RFC 2828
WORKFLOW		The sequence of activities performed in a business that produces a result of observable value to an individual actor.	Rational Unified Process



6 Epublishing

6.1 Additional functionalities in the 2nd version of the Electronic Publisher

In the 2nd version of the Electronic Publishing tool the following changes have been applied:

- Internationalisation of User Interface
- Internationalisation of publication labels
- Integration of a timeline storyboard
- Integration of the ontology component
- Refinement of existing components

6.1.1 Internationalisation of User Interface

Currently English and German is supported. The language is selected at the entry page of the Regnet Publisher. By setting the language all labels throughout the publication process are changed to the selected language. The multilinguality of the user interface is implemented using the JAVA "ResourceBundle"- Class.

6.1.2 Internationalisation of the Publication

Internationalisation of the Publication is done by an XML-lookup. This works fine for PDF and HTML documents. The languages English, Spanish, German, Bulgarian, French are supported.

6.1.3 Timeline Storyboard

Additionally, a timeline storyboard was implemented. Via the timeline storyboard the retrieved records can be grouped by time. Records which do not contain information over creation time will reside in a separate group. For creating a period of time the user has to enter the begin and the end year of the desired period and all records within this period will be automatically assigned to it. Furthermore, periods of the remaining records are built by the system itself, which then can be added to the publication or split up again into more periods.

As in the SimpleLocation - Storyboard there will be created a Virtual Gallery in SMIL 1.0 format.

6.1.4 Integration of the Ontology

Information about the publication is now stored in the ontology subsystem. The connection is established using SOAP.

The ontology subsystem is contacted three times during a publication-process. Before presenting the Start-Page of the publisher, a list of the created publications is fetched from the ontology. After having performed a query, before the Choose Storyboard dialog the ontology is contacted again for a list of created publications. This is necessary because the user will need a list of publication objects. Once the publication process has been finished, a new publication object is stored into the ontology.

6.2 Publishing workflow

Due to the changes at the implementation side of the Electronic Publisher the publishing workflow looks as follows:

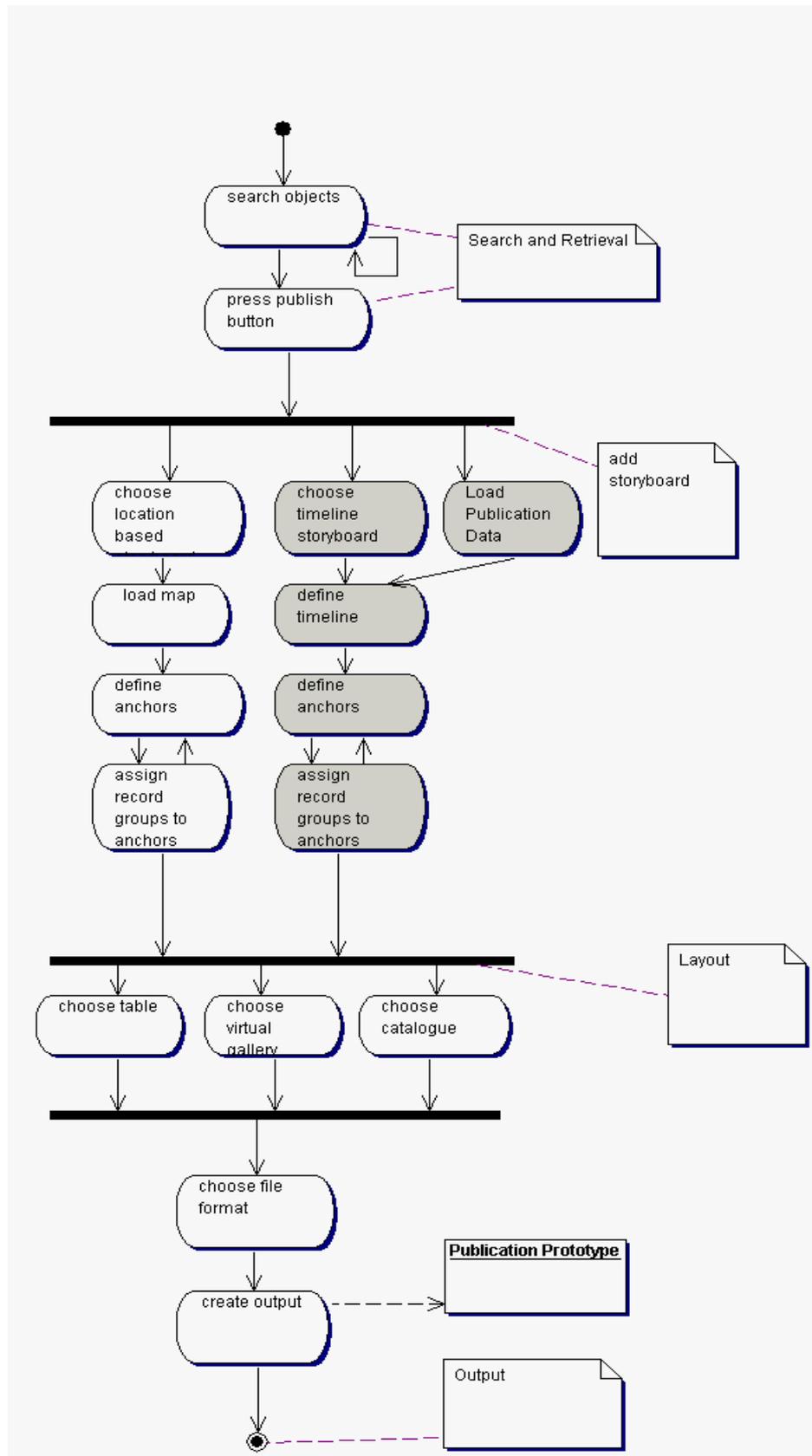


Figure 27: Electronic Publishing Workflow



Loading of Publications

The loading of a publication is now supported. The publication is identified by a ID-String which consists of the IP-address of the creating client PC and a timestamp. The publication can be selected by a dropdown list and is loaded after pushing the "load publication" button.

Loading a Publication Object

In difference to loading a publication, the publication object can be loaded, too. This is possible in the "Choose Storyboard" dialog after having performed a query (a new result set is needed). By pushing the "Load publication object" - button the publication object is loaded. The publisher will then choose the storyboard of the publication object, but load the new results. Further proceeding is as usual, the publication will be saved as new publication object.

6.3 Refinement of existing components

6.3.1 Location storyboard

Additionally to existing functionality, thumbnails are now supported in the SimpleLocation Storyboard. The URLs of the thumbnails are extracted from the records. In case of an invalid URL or if the record contains no URL the thumbnail will be a default one. The thumbnails are cached on the harddisk of the server for easier retrieve by the applet.

6.3.2 Integration of output formats

The output of publications is now available in PDF, HTML SMIL.

Not all of the formats make a useful combination with all of the layouts. The following table shows the the formats and the corresponding layouts available:

Layout/Format	PDF	HTML	SMIL
Table	Yes	Yes	-
Catalogue	Yes	Yes	-
Virtual Gallery	-	-	Yes



7 Auction

7.1 Overview

Auction system was build with a purpose to provide clients with the opportunity to bid for different products. Moreover, some clients may apply to the auction administrator in order to obtain the right to display their own items for sale. The administrator has the authority to change users' rights and the ability to forbid an item to take part in an auction.

Auction system has a mechanism, which allows moving the auction's scheduled time to an overtime, which is no more than 20 minutes after the actual time appointed for termination.

Users with different rights have a different set of options in the users' menu and as a result different functional abilities.

7.2 Structure and content of the total system

7.2.1 Functionalities

The functionalities of the auction system is different for different users' groups. The most important difference is that the ordinary user doesn't have a right and menu option to insert a new item to the auction. On the contrary users, who have this privilege, can also obtain statistics for bids made regarding their items. This menu option is provided in the profile options.

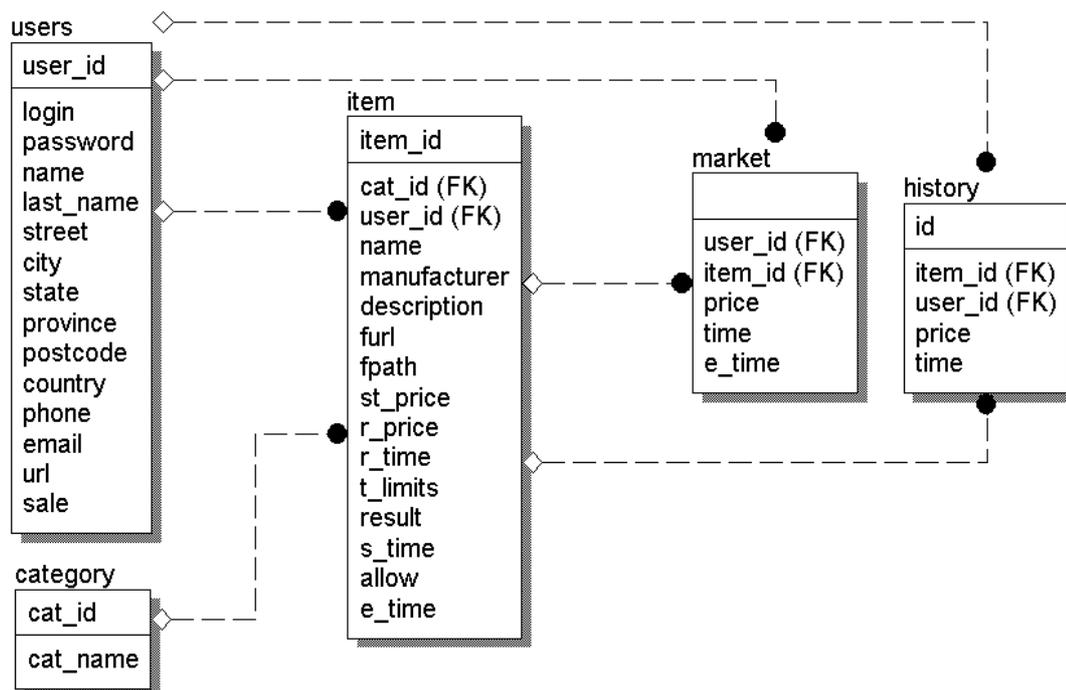
Ordinary users have the ability to see and make changes in their own profile data, and can also see their bids in different auctions. All users have the ability to get information related with current and future auctions.

The searching system provides the ability for a multi-criteria search. These criterions can be the name of the product, its description, category, start and finish auction time, providing the option to search before, after or at the moment of the appointed date.

Auction system has its own administrative system. However, this system is unavailable for ordinary users, while permission is available only by the administrator's login and password. This system allows the administrator to manage the items, users and add different categories to the database. Nevertheless, it does not allow the administrator to remove existing categories, since a great number of items may exist at the auction and any removal may destroy existing records.

7.2.2 Technical features

The database was designed according to the technical requirements. The structure is presented in Figure 28.

**Figure 28: Structure of the database**

The structure contains 5 tables with an optimal sets of attributes. Table **MARKET** is the main table, where current bids are stored. After over passing the time limits of the auction for a particular item, all the records related to this item are transferred to the table **HISTORY**, where auction's records are kept. These records can also provide statistics for bids and successful auctions.

Each item in table **ITEM** belongs to the owner (user_id). This table contains attributes such as start price (st_price), reserve price (r_price), time of registration (r_time), time to start the auction (s_time), time to end the auction (e_time), result (result), permission to start the auction (allow), and a standard set of attributes, which describe items.

Possible users' use cases are presented in Figure 29, while the aggregative algorithm of the auction is presented in figure 3. The algorithm exists only for the auction process, not for the registration or the process of managing users' profile or administrating the system.

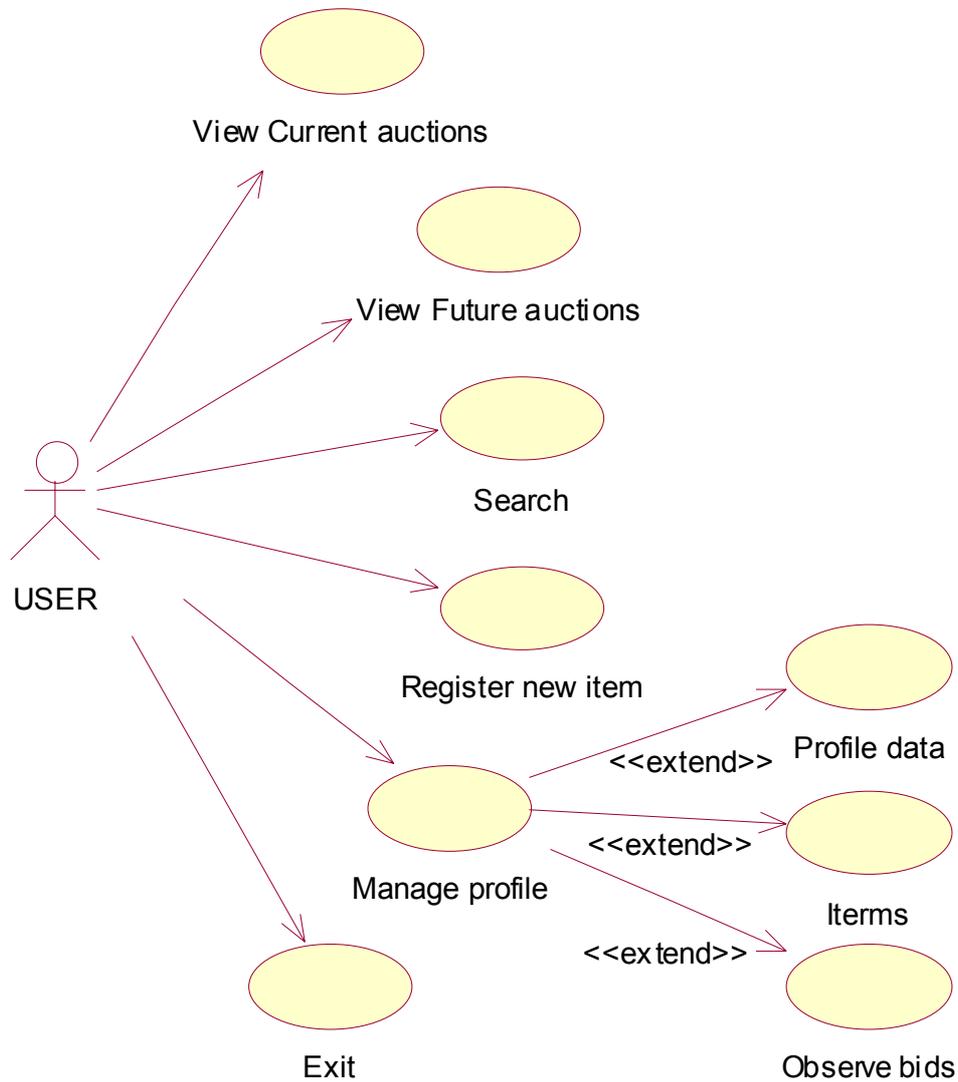


Figure 29: User use cases

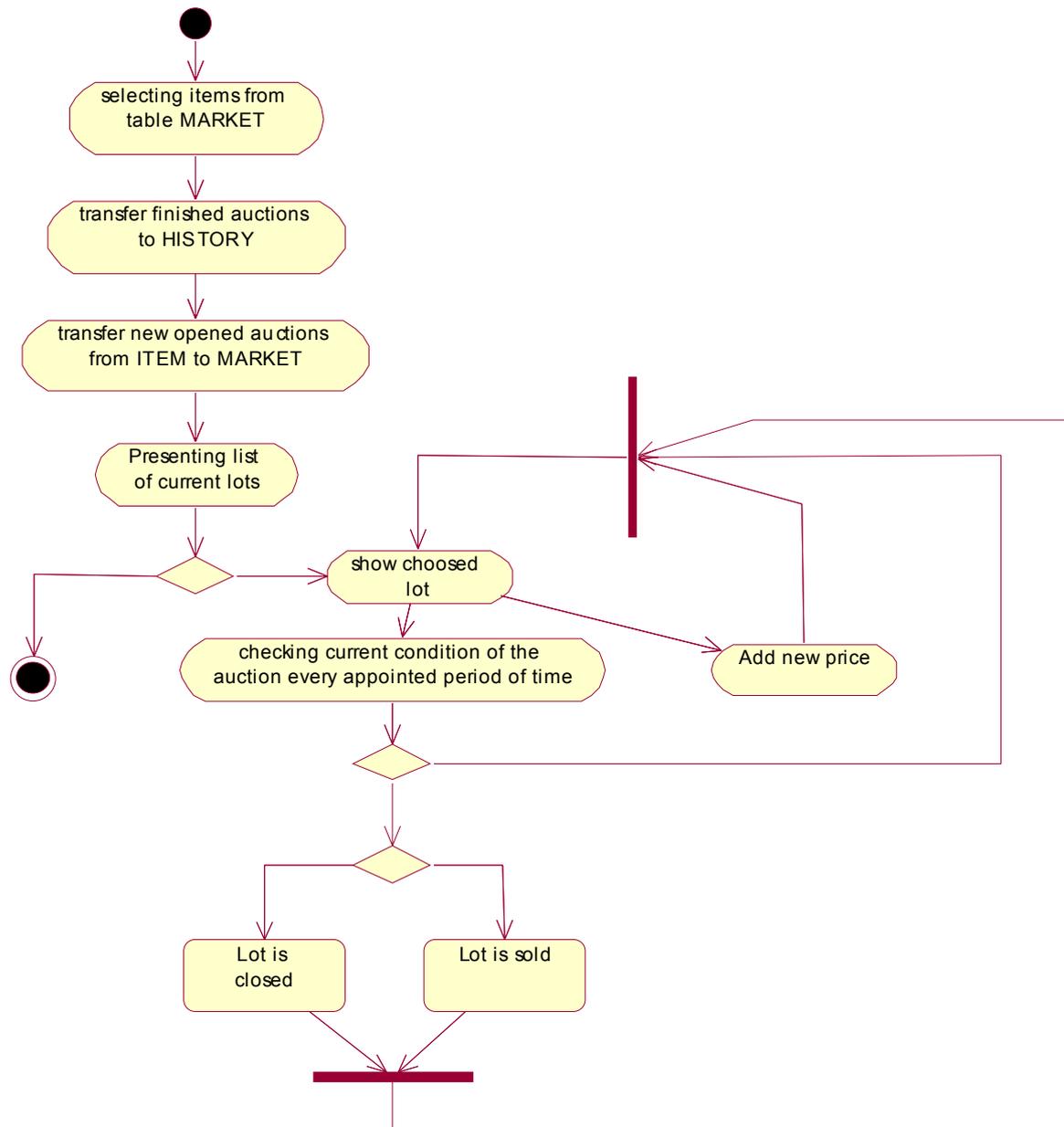


Figure 30: Algorithm of the auction

8 Delivery

8.1 Architecture

The goal of the Delivery system is to offer shipping cost computation services. This system is invoked by the procurement system when a client wants to get informed of the shipping price of her/his order.

Delivery system is implemented as a Web Service. Web services are software components that can be invoked across a network using XML. They are building blocks for building distributed applications faster, cheaper and more simply than ever before. Web services are language, platform and location independent, supported by all the major computing vendors, and quickly replacing older technologies such as CORBA, DCOM and EDI as the preferred way to tackle the challenges of EAI, B2B and B2C.

The Delivery web service is implemented using SOAP (service-oriented access protocol) standard. SOAP (service-oriented access protocol) is a general-purpose protocol for sending XML messages between endpoints, and may be used for remote procedure calls (RPC) or plain document transfer. SOAP messages can be send over any transport layer, HTTP is nevertheless the most common layer.

Here is a scheme of the architecture :

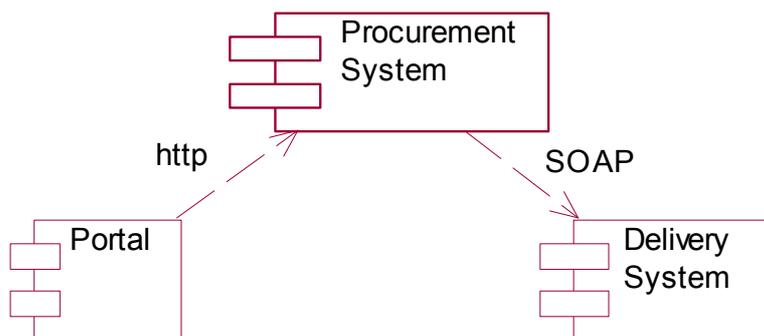


Figure 31: Architecture of the delivery system

8.2 GLUE functionalities

The Delivery subsystem is implemented with GLUE. GLUE is a Java platform for creating and deploying applications with web services, JSPs and servlets. GLUE can publish a Java object as a web service with a single line of code, and invoke web services as if they were local Java objects. GLUE generates proxies and WSDL dynamically, eliminating the need for command line tools.

8.2.1 Publishing and Unpublishing

To publish and unpublish Java objects as web services, use the following static methods defined in `electric.registry.Registry`:

void publish(String path, Object object) throws RegistryException

Publish the object to the specified path, exporting all of the object's public static and instance methods.

void unpublish(String path) throws RegistryException

Unpublish the object from the specified path.

In addition, the `electric.server.http.HTTP` class provides static methods for starting up an in-process GLUE web server that can accept incoming HTTP requests for the web services that you publish.



HTTPContext startup(String path) throws IOException

Start a web server on the specified path.

void shutdown()

Shutdown any web servers that were started using startup().

Because starting a web server causes some threads to be spawned, a program that calls HTTP.startup() will not terminate until HTTP.shutdown() is called.

The following example illustrates these methods by publishing an Exchange object that implements the following interface:

```
package examples.publish;
// method of interface ITime
public interface ITime {
// return current time
String getDate();
}
```

Here is the source code for the Exchange class:

```
Package examples.publish
Import
Public class TimeImp implements ITime
{
String getDate { return new CDate();}
}
```

The following server program starts up a web server on port 8004 of the local host to accept messages arriving on /glue. It then uses Registry.publish() to publish the Exchange object to the local path "exchange". Note that the server does not exit when the last line is reached, because the web server has threads running and Java does not halt until all threads have completed.

```
Package example.publish
Import electric.registry.Registry
Import electric.server.http.http
Pubic class Publish
{
public static void main(String[] args) throws Exception
{
// start a web server on port 8004 and accept messages via / glue
// publish an instance of Time
Registry.publish("time", new TimeImp());
}
}}
```



Here is the output from Publish:

```
>java examples.publish.Publish
```

```
[STARTUP] GLUE standard (c) 2001-2002 The Mind Electric
```

```
[STARTUP] http server started on http://199.174.24.278:8004/glue
```

At this point, the server is waiting for incoming HTTP requests to process. The next two sections show how the service may be invoked.

8.2.2 WSDL Generation

For a client to bind to a web service, it must have access to a WSDL (Web Service Description Language) file that describes the web service, including its location and a list of the operations that it can perform. GLUE dynamically generates and caches WSDL for any web service that it hosts. The `java2wsdl` utility can be used to create WSDL manually if desired, but we think you'll prefer dynamic generation.

WSDL includes argument names and documentation for each method. GLUE uses the following rules to obtain this information:

If a method is declared by an interface and the source code for the interface is available in the CLASSPATH, extract argument names and javadoc comments from the `.java` file, otherwise:

If the class was compiled with debug information, extract argument names from the `.class` file, otherwise:

Use the default argument names `arg0`, `arg1`, etc.

If your source code has a different root than your compiled classes and you want GLUE to locate the source code for your interfaces, be sure to add the root of the source directory to your CLASSPATH during the development phase. At deployment time, add the interfaces of the web services that you publish to the application's `.jar` file.

To see the WSDL for the service published above, enter the URL `http://localhost:8004/glue/exchange.wsdl` into your web browser.

8.2.3 Invoking

The most convenient way to invoke a web service is via a Java interface. The invoking web services section described how to create a Java interface using the `wsdl2java` interface. If the client already has the Java interfaces used by the web service on the server, you can bypass the use of `wsdl2java` and use the client interface directly. This allows GLUE to be used in a similar fashion to RMI for pure Java projects.

For example, the web service in the example shown above implements the `examples.publish.IExchange` interface. The following client uses the same interface to invoke the service, so there is no need to create additional client bindings using `wsdl2java`.

```
Packaga.examples.publish
```

```
Import electric.registry.publish
```

```
Public class Invoke
```

```
{
```

```
public static void main(String[] args) throws Exception
```

```
{
```

```
// bind to web service whose WSDL is at the specified URL
```

```
String url = http://localhost:8004/glue/exchange.wsdl;
```



```
ltime time = (ltime) Registry.bind(url, ltime.class);  
// invoke the web service as if it was a local java object  
String strTime = time.getTime();  
System.out.println("time is " + strTime);  
}  
}
```

Here is the output from Invoke.

```
>java examples.publish.Invoke
```

```
Date is = ..Wed Sep 04 14:31:51 CEST 2002
```

8.2.4 Invoking using HTTP GET/POST

GLUE includes support for HTTP GET/POST web service bindings. To invoke a web service using an HTTP GET, use the syntax `endpoint/method?arg1=value1&arg2=value2`. To try out this feature, run the `examples.publish.Publish1` server and then type the following URL into your web browser.

```
http://localhost:8004/glue/exchange/getTime
```

You will see the return result displayed in your browser window. If you invoke a web service in this manner using an HTTP POST, all parameters of the form `$x` are automatically replaced with their corresponding actual values from the POST form.

8.2.5 SOAP interface of Delivery system

8.2.5.1 Calculator interface

The Delivery system furnishes three main services which are specified in the following Java interface named Calculator :

```
public interface Calculator {  
  
    double getShippingCost(String departure, String destination, double parcelWeight) throws  
    InvalidDepartureException, InvalidDestinationException, InvalidWeightException,  
    InvalidDimensionException;  
  
    double getShippingCost(String departure, String destination, double parcelWeight, double  
    parcelLength, double parcelWidth, double parcelHeight) throws InvalidDepartureException,  
    InvalidDestinationException, InvalidWeightException, InvalidDimensionException;  
  
    double getShippingCost(String departure, String destination, double parcelWeight, double  
    parcelLength, double parcelWidth, double parcelHeight, double parcelValue) throws  
    InvalidDepartureException, InvalidDestinationException, InvalidWeightException,  
    InvalidDimensionException;  
}
```

```
double getShippingCost(String departure, String destination, double parcelWeight/*in kilo*/)  
throws InvalidDepartureException, InvalidDestinationException, InvalidWeightException,  
InvalidDimensionException;
```

This method calculates express shipping cost taking account only the weight of parcel

```
double getShippingCost(String departure, String destination, double parcelWeight/*in kilo*/,  
double parcelLength/*in cm*/, double parcelWidth/*in cm*/, double parcelHeight/*in cm*/)  
throws InvalidDepartureException, InvalidDestinationException, InvalidWeightException,  
InvalidDimensionException;
```



This method calculates express shipping cost taking account of the weight of parcel and its dimensions. If the parcel is too big, `InvalidDimensionException` is thrown

double getShippingCost(String departure, String destination, double parcelWeight/*in kilo*/, double parcelLength/*in cm*/, double parcelWidth/*in cm*/, double parcelHeight/*in cm*/, double parcelValue/*in euro*/) throws InvalidDepartureException, InvalidDestinationException, InvalidWeightException, InvalidDimensionException;

This method calculates express shipping cost taking account of only the weight of parcel, its dimensions and its price value. In order to calculate the cost of the insurance

8.2.5.2 Exceptions :

- `InvalidDepartureException`

This exception is thrown when a departure does not belong to the predefined list of departures

- `InvalidDestinationException`

This exception is thrown when a destination does not belong to the predefined list of destinations

- `InvalidWeightException`

This exception is thrown when a parcel weight exceeds the maximum weight allowed by the shipping company

- `InvalidDimensionException`

This exception is thrown when a parcel exceeds the maximum dimension defined by the shipping company.

8.2.6 Configuration process

The delivery uses three properties files.

- The first one is called `server.properties` and contains general properties, for instance the number of zones and the delta weight that stands for the weight step for shipping calculation.
- The second is named `countries.properties` and contains all the countries available. The key is the country code and the values are the label of the country in English and the zone number to which it is attached. For example : FR=France, 1.
- The last one is named `zones.properties` it contains the core information used for calculation.

Each key is composed of two parts separated by a dot. The first one is the zone name, it is always the word "zone" concatenated with the zone number. The second part is the weight range, it either the word "delta" and a number or the word "extra". When it is the word "delta", the number represents the range number. For instance, if the delta weight is set to 0,5 kg in the `server.properties` file, the range `delta1` starts from 0 kg to 0,5kg, the range `delta2` starts from 0,5kg to 1kg and so on. When the word is "extra" the key is used to store the cost of each delta weight that exceeds the maximum weight.

The value of each key is a float sequence separated by a comma. Each float represents the shipping cost in euro of a parcel from the zone of the key to the zone which number is the rank of this float in the sequence number. In the case of a key that contains the word "extra", each float is the cost in euro of each delta weight exceeding the maximum weight.

For instance the float 75.00 in second position in the following sequence number of the key `zone2.delta4` with a delta weight of 0,5 kg (`zone1.delta4=72.82, 75.00, 82.88, 79.88, 87.76`) means that the shipping cost for a parcel which weight is comprised between 1,5 kg and 2 kg from zone 2 to zone 2 is 75 euros.

9 Index+ gateway

9.1 Introduction

Index+ is a powerful software toolkit for creating systems to manage structured and unstructured text, data. It features fast searching, high storage capacity, a robust, network orientated, server-client architecture and a range of application development tools.

Index+ interfaces can be simple enough for inexperienced users, yet powerful enough for professional users. An Index+ database can be re-structured, copied or customised with a variety of application building tools. Index+ also provides interfaces to a range of programming environments.

Index+, created by System Simulation Ltd, has evolved as a toolset for building a range of applications for demanding clients. This range of applications and the continuous evolution of the Index+ system has ensured that it is very well suited to the requirements of the modern information economy. It enables the creation, development, management and exploitation of information and knowledge assets of any kind. System Simulation Ltd's business is now largely focused on partnerships with clients to optimise their use of information assets and maintain Index+ as a leading-edge technology.

Index+ has been designed to open systems principles. By meeting industry standards, Index+ provides users with flexibility in their choice of hardware and software components. Index+ is supported on a wide variety of hardware platforms and peripherals. Index+ can be integrated with third party hardware and software products, providing a consistent user interface across a wide range of environments.

Index+ is written in the C and C++ programming languages. Each major component of the Index+ software is implemented as a separate module with clearly defined interfaces and functions.

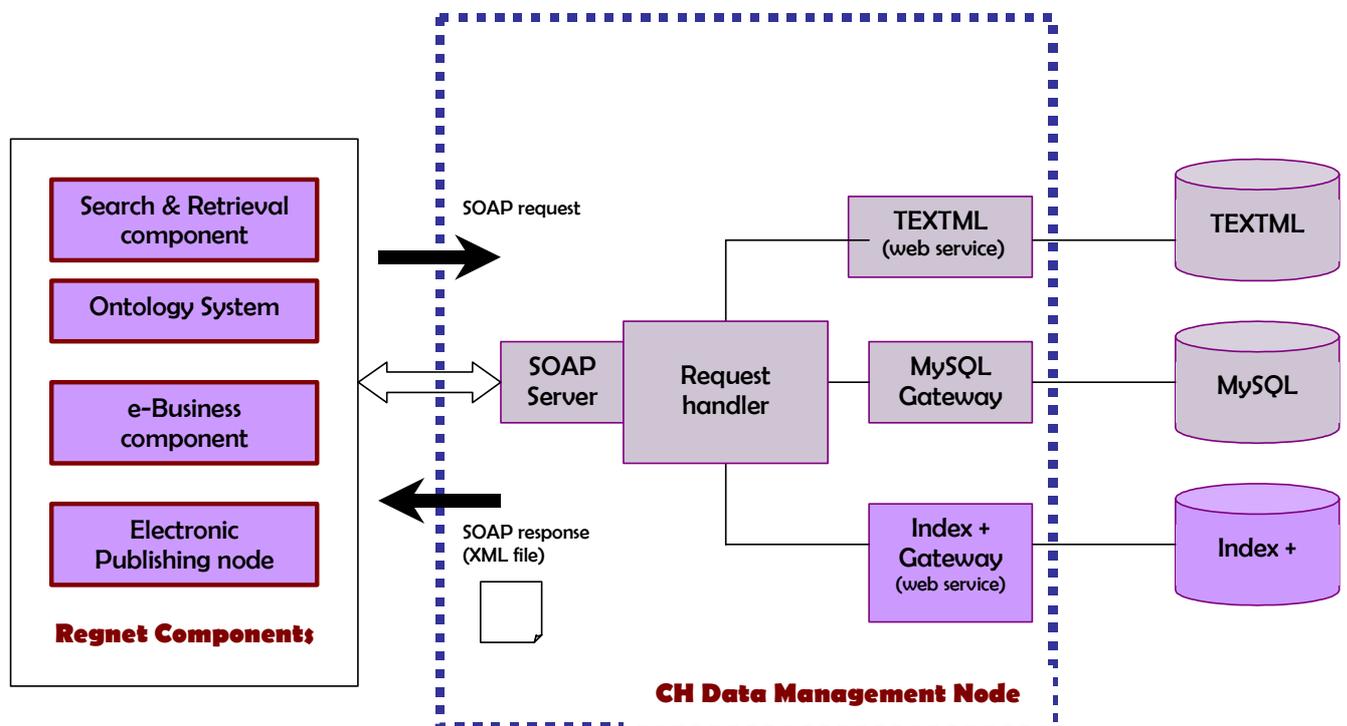


Figure 32: Overview Index+ gateway

The Index+ server is the database engine at the heart of the Index+ system. It is a single threaded process handling all storage, retrieval, and searching. This ensures integrity for information held in the database.

The Index+ Access module is a layer for accessing Index+ databases within the Reference System. It provide essentially consistent interface to the Search&Retrieve component.

Index+ gateway key technical features

- Full-text indexing and searching
- Server-client architecture with multi-platform operability, support for multiple sites and security features
- Designed to be flexible
- Suitable for applications of any size, Index+ is scalable and can adapt as information requirements change
- May be moved between different hardware platforms without having to be rebuilt
- Search and retrieve information from remote databases.

9.2 Index+ Gateway component

The target of Index+ gateway is to transform all the request for the Index+ repositories. The index+ connection will be usefull in the project because of the clustering of RegNet with the OpenHeritage project.

The Index+ gateway component for the CH Management Node is designed to link directly to the Regnet Web server so allowing users to access Index+ databases from any Web browser.

The Index+ gateway is a web service which allows the Cultural Heritage Management Node to search and retrieve information from remote Index+ databases.

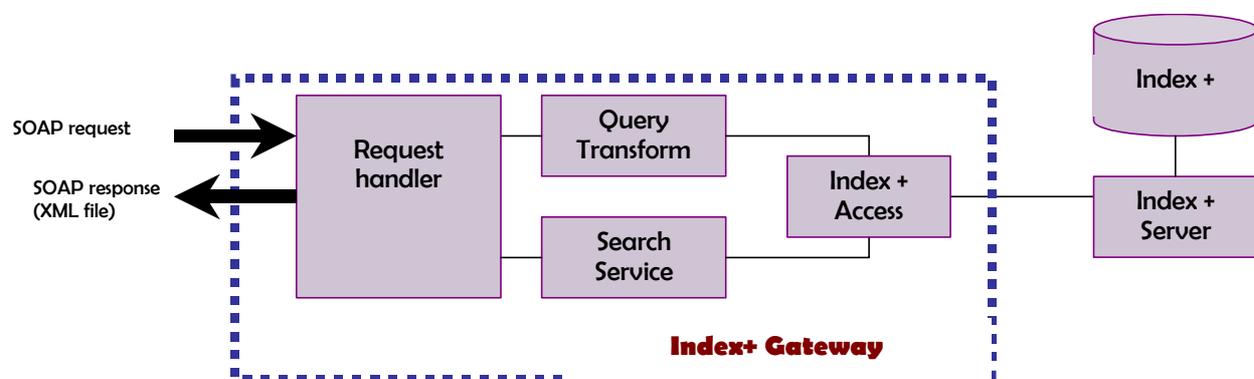


Figure 33: Index+ gateway component

The interface that the Index+ Gateway has to realize is the same of the component for access the TextML database. Fore the search service the component has to provide the following functionalities::

- SearchRetrieve
The main method of the Search Service. Returns query hits and the records found respectively.
- GetRecords
Returns the records specified in a record list filtered by the optional doctypes list.



The Index+ server is supported by a full range of clients and other utilities for database configuration and management.

The Index+ gateway provide the interface to the Index+ server for the RegNet platform through its programmable interface to the basic services in the C language.

The Index+ Access modul also provides a set of application libraries supporting higher-level capabilities in C, Visual Basic, C++ and an application developers language called IPI.

9.3 Index+ features

Text searches look for the presence of a given word or phrase in the documents being searched. They can include proximity and partial matching requirements. Proximity specifies that the words being searched for must occur within a given span of words and optionally in a given order. Partial matching allows incomplete words to be entered together with wildcard characters. Wildcard characters can occur anywhere in the search specification.

Expression searches look for numbers or strings within a range of values. They can search for the results of calculations based on the contents of fields in one or more records. This provides Index+ with functionality common to conventional numerically orientated relational databases.

By default Index+ searches across all the fields in a data record. However, if required, searches may be restricted to a particular field or group of fields. Clients can be configured to allow the user to create complex queries by simply filling in forms or to give the user full flexibility and power of the Index+ retrieval language in a command-line interface

Index+ does not impose any pre-defined structure on the application. It can store and index large bodies of text as single fields or records as well as supporting the highly structured organisation typical of a relational database application. Both approaches can be used in a single database allowing the data to be structured to suit the needs of the application rather than the dictates of the underlying database system.

Index+ Data structures

Index+ databases are configured from the following classes of objects - fields, links, record types and indexes. A field is a sub-category of information about a particular entity or object. For example, a record containing information about a particular artist is likely to have a field for the artist's name, a field for their date of birth, and a field giving information about the artist's work. Fields can hold variable length text, numbers or dates. The Index+ server can also handle structured data in the form of lists, arrays and arbitrarily nested structures.

Links provide one-to-one, one-to-many and many-to-many associations between records allowing complex data objects to be built up in the manner of object-oriented databases. Links can have an associated type so that retrievals can be made on the basis of link type. Index+ has different ways of implementing links to enable the most appropriate to be chosen for a specific link function.

Record types consist of collections of fields. They allow different kinds of records to be maintained separately in the same database. Each record type in a database can have a different set of fields and types of information or they can share fields, allowing users to search for field-specific information across multiple record types.

Text indexes

Text indexes divide text fields into a number of words or phrases using a word parser and then index the location of each word or phrase.

Text indexes are configurable. They can be set up to use particular character sets for parsing a body of text into individual words. This means that Index+ has no difficulty with non-English characters such as those with accents. Text indexes can be configured to be case sensitive or insensitive. They can contain information about which field contains the word and which words surround it to allow for field-specific or proximity searching. If there are words that do not need to be indexed, these can be held in a stoplist.



A database can contain many differently-configured text indexes. Typically a single field-specific text index may be used for the majority of the text fields with separate text indexes used for special cases such as part numbers, accession numbers and so on.

Expression indexes

Expression indexes support both range searching (retrieving all records with a field value between an upper and lower limit) and the retrieval of a set of records in a particular order. This can be used to index date, time, date/time and numeric fields, or to allow alphabetic browsing of structured text fields.

Expressions can be numerical or logical calculations and can utilise string manipulation operations. A versatile language allows the result of any expression which can be derived from a field or combination of fields to be indexed

Index+ has developed in response to the requirements of our users. It has many features to support the implementation of robust, easy to use, information systems.

Multuser

Index+ servers running on UNIX platforms are full multuser systems, using region locking on files. Index+ has been designed so that items in the database are kept locked for the shortest possible time, giving impressive performance statistics. The only limit on the number of users is the hardware available.

Multi-platform operability

Index+ can operate in a wide range of environments and configurations including:

- Single user MS-Windows
- UNIX server networked to MS-Windows PC clients
- UNIX server and clients with serially linked user terminals
- UNIX server and clients with LAN and user PCs in terminal emulation mode
- NT server configurations as for UNIX

The database files created by Index+ are portable between implementations on different platforms. This means that a database can be built on one machine and then distributed and run on a range of different systems without having to re-build the database.

The Index+ server is designed to be portable to any UNIX or POSIX compliant platform. It has been ported to a wide and growing range of platforms. We will port Index+ to new environments on request.

Network support

The Index+ server-client networking layer uses a network abstraction which means that Index+ can be run on a variety of networks. Several different network interfaces are supported on various platforms. In recent years there has been a strong trend among network software providers towards presenting applications with a single interface to networking, regardless of the network or the network software. This means that often a single Index+ installation can run on different underlying network platforms, just by changing the networking software: no change is necessary to Index+. This is particularly true of TCP/IP networks, the most common network for running server-client Index+.

Unless specially requested, Index+ networking clients for MS-Windows are now configured to use the WINSOCK.DLL v1.1 interface to TCP/IP. Index+ will run over any network software which conforms to this standard.

On UNIX systems, Index+ is built to use the 4.2/4.3BSD sockets interface, where available. This allows Index+ software to run alongside all the other standard TCP/ IP applications available under UNIX.

Multimedia support, electronic publishing

Index+ is a sophisticated tool for creating high quality presentations covering a large quantity of text, graphical and multimedia content served from an Index+ database.



Index+ can store any form of digital data, indexing it as required by use of an appropriate parser and index combination. This makes it ideal for supporting large scale repositories for managing digital data for electronic publishing on CD-ROM, online systems or other media. These repositories hold digital content in a general purpose form that can be re-used for multiple purposes thus offsetting the cost of digitisation.

Thesaurus support

The Index+ Thesaurus Editor supports the development and use of thesauri to BS 5723 (ISO 2788). Users can import standard thesauri such as the BSI Root thesaurus and then develop them further to meet their particular requirements.

The Index+ thesaurus provides a means of enforcing a controlled vocabulary in certain fields during data input as well as being an aid for searching. When searching, the thesaurus offers broader, narrower or related terms in order to refine a query.

Support and Training

We offer a comprehensive package of support arrangements tailored to user requirements, ranging from 24-hour pager support to telephone and email support. Training is provided for application builders, database administrators and end-users.

Migration tools

Tools for importing data from a wide range of third party files and databases to Index+ applications are available. We have extensive experience in this area.

Specialist requirements

Information management systems frequently require support for a range of specialist functions as well as the basic storage and retrieval facilities. The application building tools provided with Index+ are designed so that appropriately tailored systems are very easy to make, use and maintain. These facilities include:

- Interfaces to newswire systems
- Storage management for large text and image archives
- Interfaces to Point of Sale equipment
- Interfaces to EDI systems
- Interfaces to scanning and OCR software
- Script-based multimedia authoring
- Loan management
- Exhibition design

Index+ has been successfully integrated with RDBMS and other more specialised applications, for example workflow automation or accounting systems.



10 Cooperative thesauri

10.1 Functionalities

- select a term and add to document
- select a term and search for this or narrower term
- edit the thesaurus (inplace editor) You can select a term you can: rename, move up/down on same level, move up/down in hierarchy, copy, cut, paste (with all narrower terms)

10.2 Structure

The thesaurus structure is stored in a single xml file as specified in Thesaurus Doctype

10.3 Thesaurus Doctype

```
<!ELEMENT node (node*)>
<!ATTLIST node
  name CDATA #REQUIRED
>
<!ELEMENT thesaurus (node*)>
<!ATTLIST thesaurus
  name CDATA #REQUIRED
>
```



11 Annexe A: Installation manuel

11.1 Auction

Auction system doesn't support distributed databases, that's why the database and the files with a program code must be placed on the same server!

To install auction system correctly it is necessary to have the following software be installed:

Apache 1.3....(we have used Apache 1.3.20)

PHP 4.0....

Mysql 3.23

Apache web server should be configured to execute php scripts under the phtml extension of the files.

Then, we have to place the following files to the every place in the html documents tree of the Apache web server (it is preferable to create new folder, like auction or something like that):

add.phtml
appl.phtml
auction.phtml
auct.phtml
current.phtml
future.phtml
history.phtml
index.phtml
login.phtml
menu.phtml
passes.phtml
post.phtml
profile.phtml
regs.phtml
scripts.js
search.phtml
styles.css

and folder **/images** for images of the inserted items.

Also it is necessary to place the administrative part (folder **admin**) of the auction to some directory in the same web server. It is preferable to put it to the same directory as an auction system.

The folder admin must contain the following files:

adm.phtml
index.phtml
passes.phtml
styles.css



Also it will be necessary to insert the specific database of the auction system (auction is the name of specific database) in the right folder of mysql path

11.2 Delivery

11.2.1 Configuration

This is the required tools

Java Environment	Sun Java 1.3 runtime environment
Libraries (.jar)	GLUE-STD.jar Dom.jar Jnet.jar Servlet.jar
Prototype	delivery.zip

11.2.2 Requirements

REGNET portal shall be installed

11.2.3 Installation procedure

The steps to follow are the following:

1. Use compile file to compile java class file
2. Use startServer batfile to launch web server
3. Use startClient bat file to lanch a test client

11.3 Index+ gateway [SPACE]

The following is documentation on how to install the Index+ gateway web service on a Windows 2000 environment.

This section includes:

- Hardware, software, and database requirements
- Instructions for installations of entire environment
- Instructions for configuring and running webservice

11.3.1 Hardware & software requirements

Server hardware requirements depend on the entire RegNet platform.

Minimum Configuration

- 700 MHz Pentium II Processor
- 256MB RAM
- 4 GB hard disk with a minimum of 1 GB free space

Minimum Enterprise Configuration

- Dual 450 MHz Pentium II Processor



- 512MB RAM
- 4 GB hard disk with a minimum of 1 GB free space

Server Software Requirements

- Windows 2000 Professional
- Service Pack 2
- Microsoft, Internet Information Server (IIS)
- Windows Update
- .NET SDK environment

11.3.2 Instructions for installations of entire environment

Hereafter are listed all the actions executed to set up the server environment:

- Installation of Microsoft Windows 2000 operative system
- Installation of Service Pack 2 for Microsoft Windows 2000
- Installation of Microsoft, Internet Information Server (IIS). We installed Microsoft's web server, Internet Information Server (IIS) for free from the Windows 2000 Pro installation CD and configured it to run on our system by following the instructions below:
 - Place the Windows 2000 Professional CD-Rom into your CD-Rom Drive.
 - Open '*Add/Remove Windows Components*' found in '*Add/Remove Programs*' in the '*Control Panel*'.
 - Place a tick in the check box for '*Internet Information Services (IIS)*' leaving all the default installation settings intact.
 - Once IIS is installed on your machine you can configure IIS through the '*Personal Web Manager*' found in the '*Administration Tools*' in the '*Control Panel*'.
 - Double-click on the '*Personal Web Manager*' icon.
 - Once the Personal Web Manager is open you will see the Main dialog box where it will show your home page and home directory default values. Where the home page is shown below as '*http://My_Computer*', will be '*http://*' followed by the name of your computer. Clicking on each of these values will open your home page in your default web browser or open the default home directory in Windows Explorer.
 - To view your home page in a web browser type '*http://localhost*' (you can substitute '*localhost*' for the name of your computer) into the address bar of your web browser. If you are not sure of the name of your computer right-click on the '*My Computer*' icon on your desktop, select '*Properties*' from the shortcut menu, and click on the '*Network Identification*' tab.
 - Until you place your own web site in the default directory for the web servers home page you should now be looking at the documentation for IIS.
 - To place your own web site in place of the IIS documentation in your home page you need to place your own web page in the '*c:\inetpub\wwwroot*' directory making sure the page is called Default.htm or Default.asp. Now when you type '*http://* followed by your computer name', into your web browser you should see your own home page.
 - Additionally we have to be sure to get the latest service packs for IIS from Microsoft. It is known to have many security holes at release and many of them have been addressed through the service packs.



- Through Windows Update we installed all the new updated version suggested excepted for Service Pack3, DirectX and latest version of Media Player.
- Installation of MDAC 2.7
- Installation of .Net SDK environment

11.3.3 Instructions for configuring and running webservice

The computer is now ready to proceed with the webservice component installation.

- Create a new folder named for instance C:\IndexPlus where to copy DLL libraries of IndexPlus
- You have to add this folder to the Computer path
- Copy into the new folder all the Index+ libraries

Now we are able to compile the developed code (written in C#) in order to build our own library that allows the interaction between our component (Index+gateway web services) and the Index+ DLL.

iplusixdb.cs is the source file of the C# code that interacts with the native Index+ DLLs.

iplusixdb.netmodule is the intermediate version of the above

IndexPlus.key is the key file that is used to make the library.

- Copy files iplusixdb.cs, IndexPlus.key in a temporary directory. For instance C:\temp
- Compile and link the iplusixdb.cs file

```
c:  
cd temp  
csc /t:module /unsafe iplusixdb.cs  
al /keyfile:IndexPlus.key /out:iplusixdb.dll iplusixdb.netmodule
```

- Copy the created DLL in the bin sub-directory of webroot on IIS environment

```
cd \inetpub\wwwroot  
mkdir bin  
copy c:\temp\iplusixdb.dll bin
```

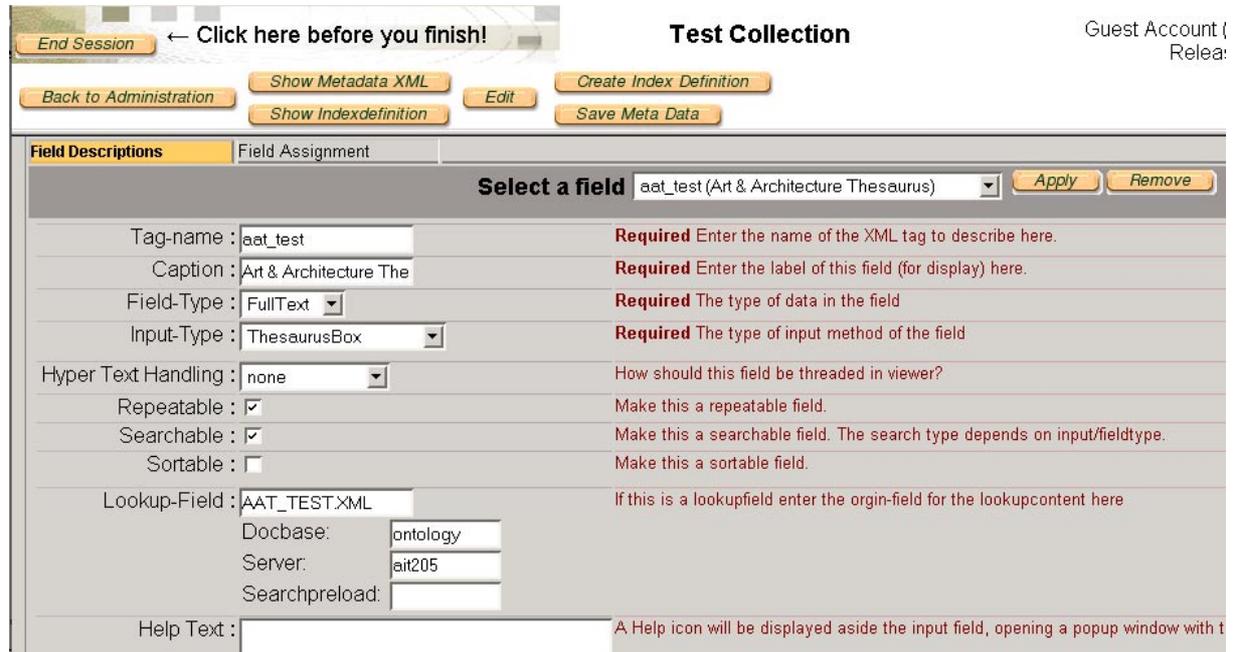
- Register the created DLL in .NET cache

```
cd \inetpub\wwwroot\bin  
gacutil /i iplusixdb.dll
```

At this point we are able to install the index.asmx file in the Web server root.

To test if everything is fine we can simply point the browser to <http://localhost/index.asmx>

11.4 Cooperative thesauri



End Session ← Click here before you finish! Test Collection Guest Account (Release:)

Back to Administration Show Metadata XML Edit Create Index Definition Show Indexdefinition Save Meta Data

Field Descriptions Field Assignment

Select a field aat_test (Art & Architecture Thesaurus) Apply Remove

Tag-name : aat_test Required Enter the name of the XML tag to describe here.

Caption : Art & Architecture The Required Enter the label of this field (for display) here.

Field-Type : FullText Required The type of data in the field

Input-Type : ThesaurusBox Required The type of input method of the field

Hyper Text Handling : none How should this field be threaded in viewer?

Repeatable : Make this a repeatable field.

Searchable : Make this a searchable field. The search type depends on input/fieldtype.

Sortable : Make this a sortable field.

Lookup-Field : AAT_TEST.XML If this is a lookupfield enter the orgin-field for the lookupcontent here

Docbase: ontology

Server: ait205

Searchpreload:

Help Text : A Help icon will be displayed aside the input field, opening a popup window with t

Figure 34: Screenshot “Properties of a thesaurus field”

The screenshot above shows the properties of a thesaurus field. The field input type has to be “Thesaurus Box” and the source for contents of this box, in this example this is a AAT test file, is stored on the ontology database.

In order to create a thesaurus there can on one hand be worked on contents via the interface as described in the previous chapter on the other hand a user can write an XML file defining the thesaurus and import it via the console of the TeXtML Server into the ontology.

The AAT test thesaurus file:

```
<?xml version="1.0"?>
<thesaurus name="AAT">
  <node name="multidisciplinary concepts" CN="B">
    <node name="Associated Concepts" CN="BM">
      <node name="concepts in the arts" CN="AFU">
        <node name="artistic concepts" CN="AFU">
          <node name="artistic devices" CN="AFU"/>
          <node name="concepts relating to the creative process" CN="ALO"/>
          <node name="formal concepts" CN="ARI"/>
          <node name="forms of expression" CN="AXC"/>
        </node>
      <node name="genres in the arts" CN="ALO">
        <node name="architecture genres" CN="BCW"/>
        <node name="art genres" CN="BIQ"/>
        <node name="erotica" CN="AFU"/>
        <node name="literary and oral genres" CN="BOK"/>
      </node>
    </node>
  </node>
</thesaurus>
```



```
<node name="pastoral" CN="ALO"/>
<node name="pornography" CN="ARI"/>
<node name="propaganda" CN="AXC"/>
</node>
<node name="historical, theoretical and critical concepts" CN="ARI">
  <node name="Antique, the" CN="AFU"/>
  <node name="appropriation" CN="ALO"/>
  <node name="art for art's sake" CN="ARI"/>
  <node name="connoisseurship" CN="AXC"/>
  <node name="critical theories" CN="BCW"/>
  <node name="decorum" CN="BIQ"/>
  <node name="Gesamtkunstwerk" CN="BOK"/>
  <node name="hermeneutics" CN="BUE"/>
  <node name="historical, theoretical and critical concepts: literature" CN="ELK"/>
  <node name="historical, theoretical and critical concepts: visual arts" CN="ERE"/>
  <node name="intermedia" CN="BZY"/>
  <node name="kitsch" CN="CFS"/>
  <node name="masterpiece" CN="CLM"/>
  <node name="paragone" CN="CRG"/>
  <node name="period" CN="CXA"/>
  <node name="provenance" CN="DCU"/>
  <node name="revival" CN="DIO"/>
  <node name="school (critical concept)" CN="DOI"/>
  <node name="style" CN="DUC"/>
  <node name="technique (concept)" CN="DZW"/>
</node>
</node>
<node name="culture and related concepts" CN="ALO">
  <node name="culture" CN="AFU">
    <node name="Museums / Library domain"/>
  </node>
  <node name="culture-related concepts" CN="ALO"/>
</node>
<node name="environmental concepts" CN="ARI">
  <node name="air quality" CN="AFU"/>
  <node name="drainage" CN="ALO"/>
  <node name="environmental impact" CN="ARI"/>
  <node name="green design" CN="BCW"/>
  <node name="humidity" CN="BIQ"/>
  <node name="infestation" CN="BOK"/>
  <node name="noise" CN="BUE"/>
  <node name="odors" CN="BZY"/>
</node>
```



```
<node name="pollution" CN="CFS"/>
  <node name="resources concepts" CN="CLM"/>
</node>
<node name="functional concepts" CN="AXC">
  <node name="access" CN="AFU"/>
  <node name="barrier-free design" CN="ALO"/>
  <node name="circulation (architecture)" CN="ARI"/>
  <node name="housing" CN="AXC"/>
  <node name="joint occupancy" CN="BCW"/>
  <node name="safety and related concepts" CN="BIQ"/>
  <node name="space utilization" CN="BOK"/>
  <node name="storage" CN="BUE"/>
  <node name="use" CN="BZY"/>
</node>
<node name="multidisciplinary concepts" CN="BCW">
  <node name="authenticity" CN="AFU"/>
  <node name="hybridity" CN="ALO"/>
  <node name="methodology" CN="ARI"/>
  <node name="standards (concepts)" CN="AXC"/>
  <node name="theory" CN="BCW"/>
  <node name="typology" CN="BIQ"/>
</node>
<node name="philosophical concepts" CN="BIQ">
  <node name="aesthetic concepts" CN="AFU"/>
  <node name="ethics (concept)" CN="ALO"/>
  <node name="ideology" CN="ARI"/>
  <node name="metaphysical concepts" CN="AXC"/>
  <node name="Nature" CN="BCW"/>
  <node name="philosophical movements and attitudes" CN="BIQ"/>
  <node name="truth" CN="BOK"/>
</node>
<node name="religions and religious concepts" CN="BOK">
  <node name="religions" CN="AFU"/>
  <node name="religious concepts" CN="ALO"/>
</node>
<node name="scientific concepts" CN="BUE">
  <node name="biological concepts" CN="AFU"/>
  <node name="computer science concepts" CN="ALO"/>
  <node name="contamination" CN="ARI"/>
  <node name="engineering concepts" CN="AXC"/>
  <node name="materials sciences concepts" CN="BCW"/>
  <node name="mathematical concepts" CN="BIQ"/>
</node>
```



```
<node name="physical sciences concepts" CN="BOK"/>
</node>
<node name="social science concepts" CN="BZY">
  <node name="communications concepts" CN="AFU"/>
  <node name="economic concepts" CN="ALO"/>
  <node name="legal concepts" CN="ARI"/>
  <node name="political concepts" CN="AXC"/>
  <node name="psychological concepts" CN="BCW"/>
  <node name="social and economic geography concepts" CN="BIQ"/>
  <node name="sociological concepts" CN="BOK"/>
</node>
<node name="technology and related concepts" CN="CFS">
  <node name="technology" CN="AFU"/>
  <node name="technology-related concepts" CN="ALO"/>
</node>
<node name="transportation and related concepts" CN="CLM">
  <node name="transportation" CN="AFU"/>
  <node name="transportation-related concepts" CN="ALO"/>
</node>
</node>
</node>
</thesaurus>
```



12 Annexe B: User manuel

12.1 Auction

12.1.1 GENERAL

Auction system was build with a purpose to provide clients with the opportunity to bid for different products. Moreover, some clients may apply to the auction administrator in order to obtain the right to display their own items for sale. The administrator has the authority to change users' rights and the ability to forbid an item to take part in an auction. To be able to bid on a lot, you must register as a user. This means choosing a username and a password, and also completing the other fields. The username must be unique.

As it was mentioned before, the functionality of the auction system is different for different users' groups. The most important difference is that the ordinary user does not have the right to insert a new item to the auction and does not have a menu option. On the contrary users, who have this privilege, can also obtain statistics made for bids regarding their items. (You can see the figure below)

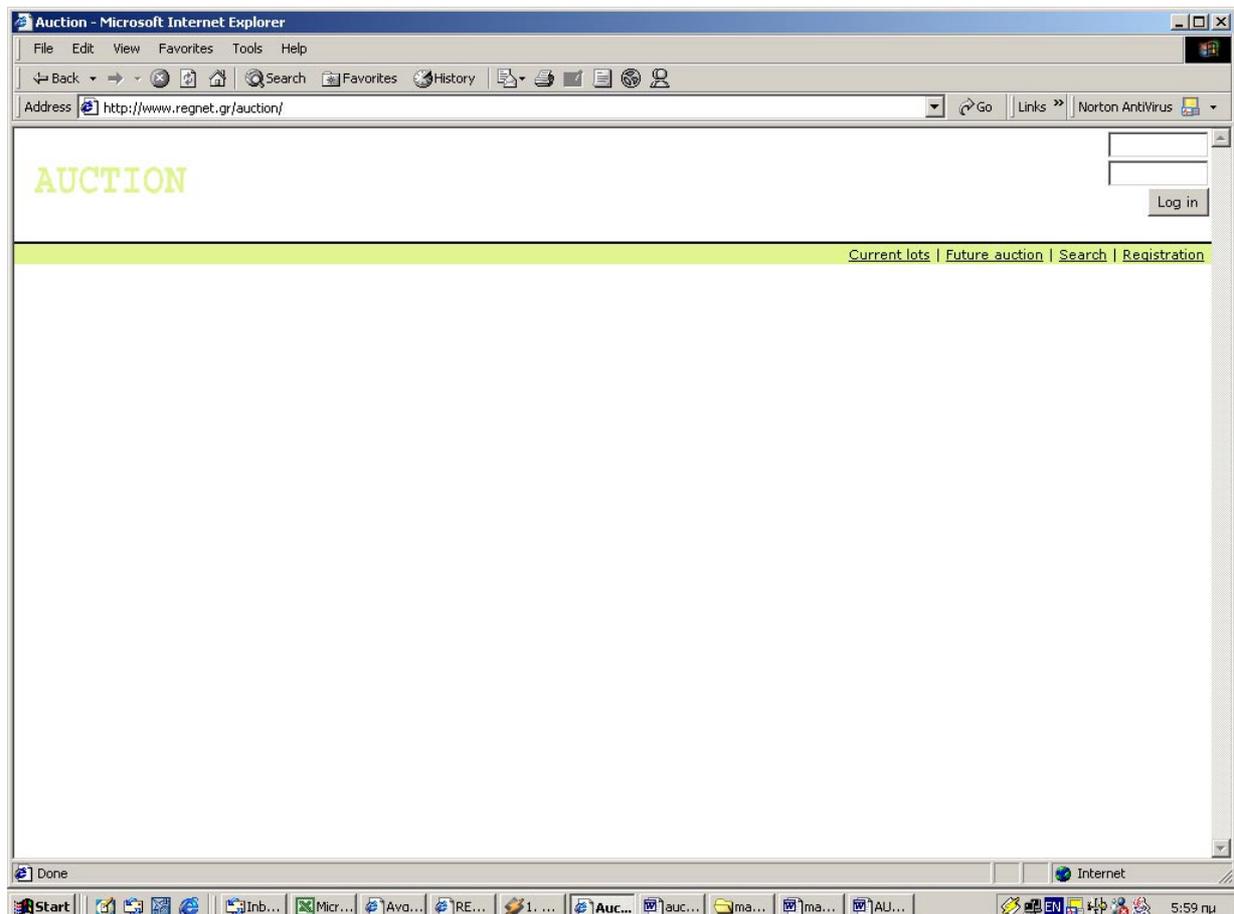


Figure 35: Auction system

12.1.2 USERS REGISTRATION

The system can be used not only for registered users as mentioned above, meanwhile the full functionality of the system can be used only by registered users. Thus, if a user wants to become a

registered user, firstly he/she has to go to the specific registration form in order to provide all his/her personal information and also to choose the preferable login name and his/her password. All the fields, which are necessary and should be provided by the user, are marked with a red star. (You can see the figure 2).

Furthermore if a registered user wants to sell his/her items, he/she has to contact with the system administrator in order to get a specific permission.

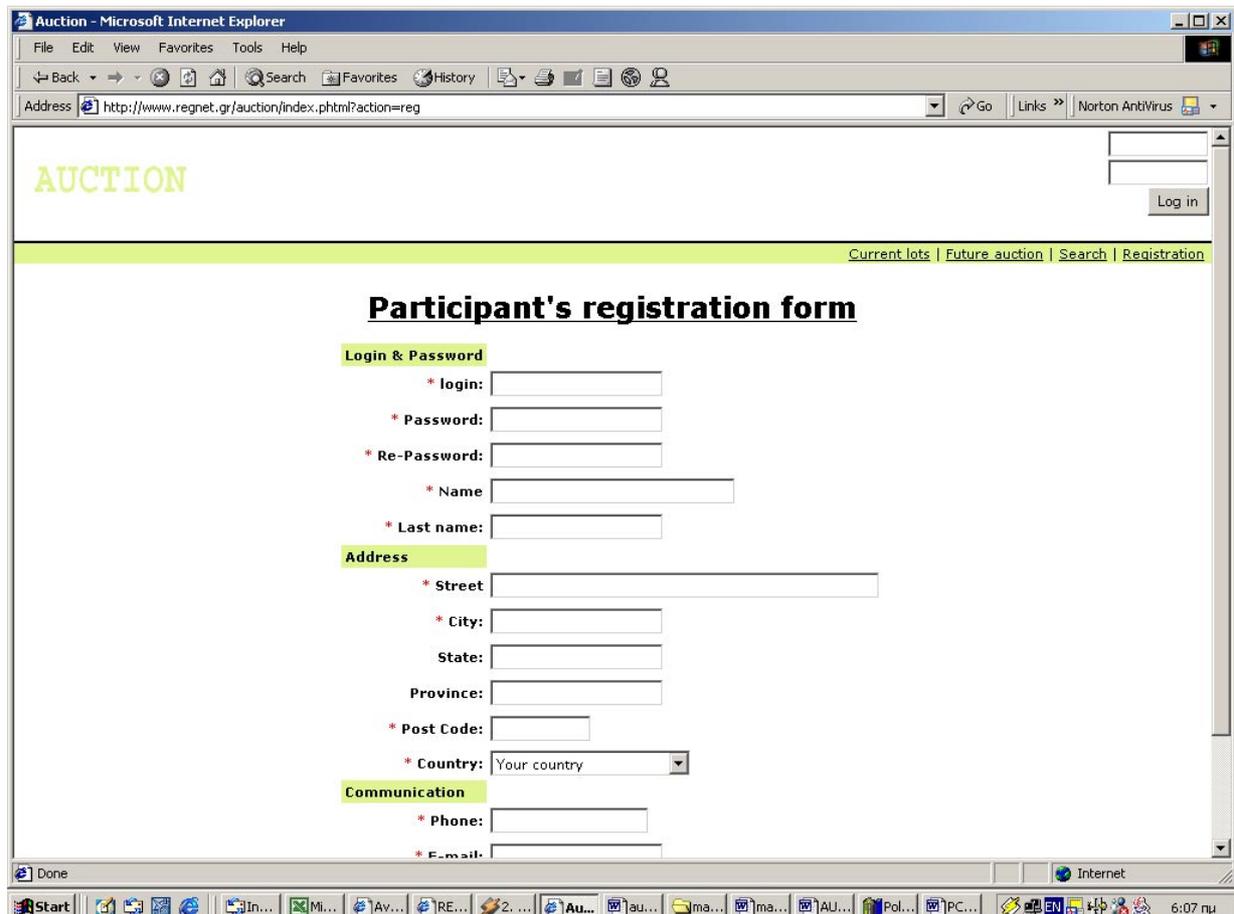


Figure 36: Registration form

12.1.3 SEEING THE LOTS

The system provides the option to the users to see all current lots, which mean the lots that are in progress by following the specific link "current lots" from the menu framework, provided by the system (you can see the figure 3).

Also users have the option to see future auctions, which are the auctions that have not started yet up to current date and time (see figure 4).



The screenshot shows a Microsoft Internet Explorer browser window displaying the REGNET auction page. The address bar shows the URL: <http://www.regnet.gr/auction/index.phtml?action=curr&b=0&sale=>. The page title is "AUCTION". There are search and login fields in the top right corner. Below the header, there are navigation links: [Current lots](#), [Future auction](#), [Search](#), and [Registration](#). The main content area is titled "Current Lots" and contains a table with two rows of auction items. The table has columns for Name, Description, Producer, Picture, Start price, Started from, and Bargain?. The first row lists "Equestrian Portrait of the Emperor Maximilian" by Hans Burgkmair, with a start price of 11.00 and a start date of 2002-06-01-01-00. The second row lists "A Tournament Scene" by Lucas Cranach, with a start price of 11.00 and a start date of 2002-03-01-01-00. Below the table, there is a pagination link: "Page: [1](#) [2](#) [3](#) [4](#) [5](#) [6](#)".

Name	Description	Producer	Picture	Start price	Started from	Bargain?
Equestrian Portrait of the Emperor Maximilian	Equestrian Portrait of the Emperor Maximilian Hans Burgkmair (German, 1473 - 1531)	Hans Burgkmair		11.00	2002-06-01-01-00	GO >>
A Tournament Scene	A Tournament Scene Lucas Cranach (German, 1472 - 1553)	Lucas Cranach		11.00	2002-03-01-01-00	GO >>

Figure 37: The view of the current lots

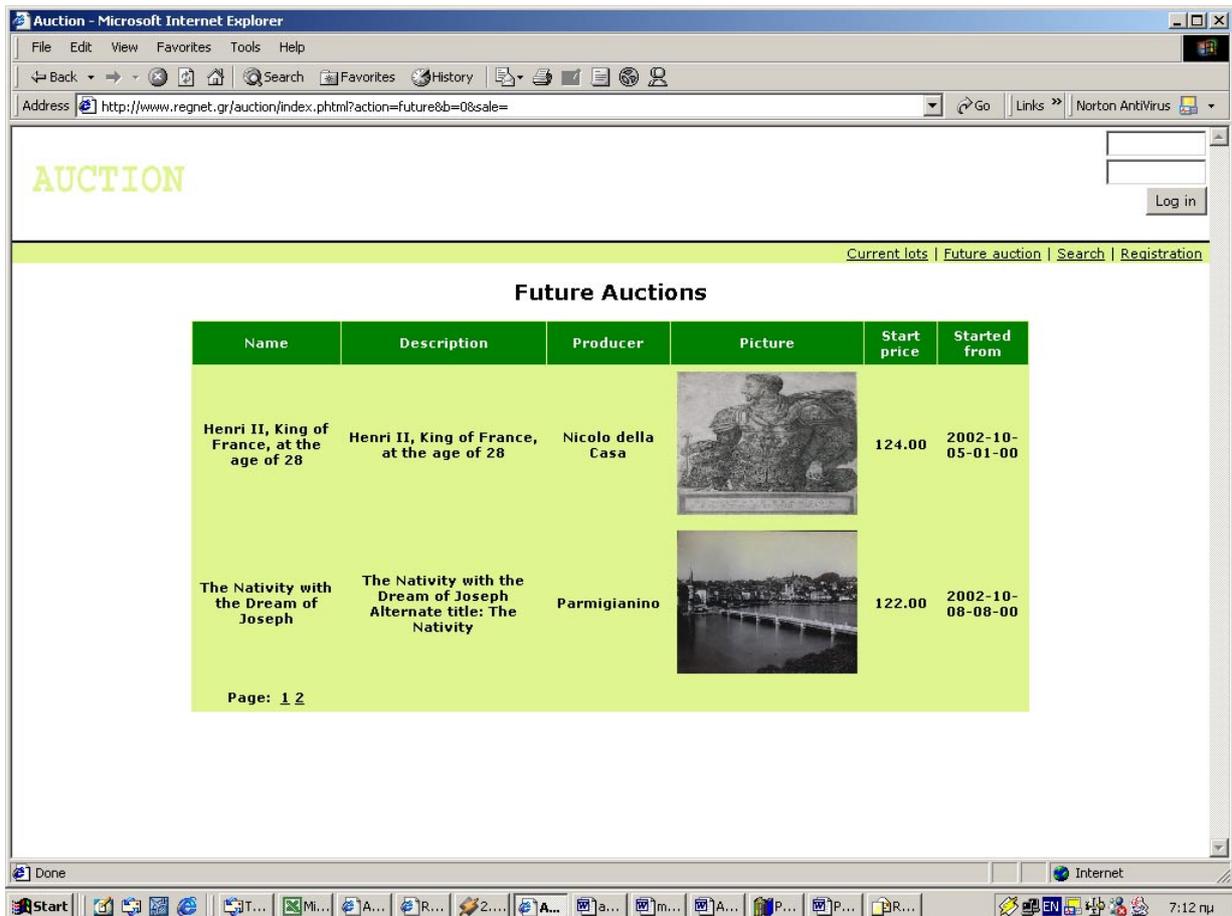


Figure 38: The future auctions

12.1.4 PRODUCTS SEARCH

The system's searching ability provides the option for a multi-criteria search. These criteria can be the name of the product, its description, category, start and finish auction time, the option to search before, after or at the moment of the appointed date. (You can see the figure 5)

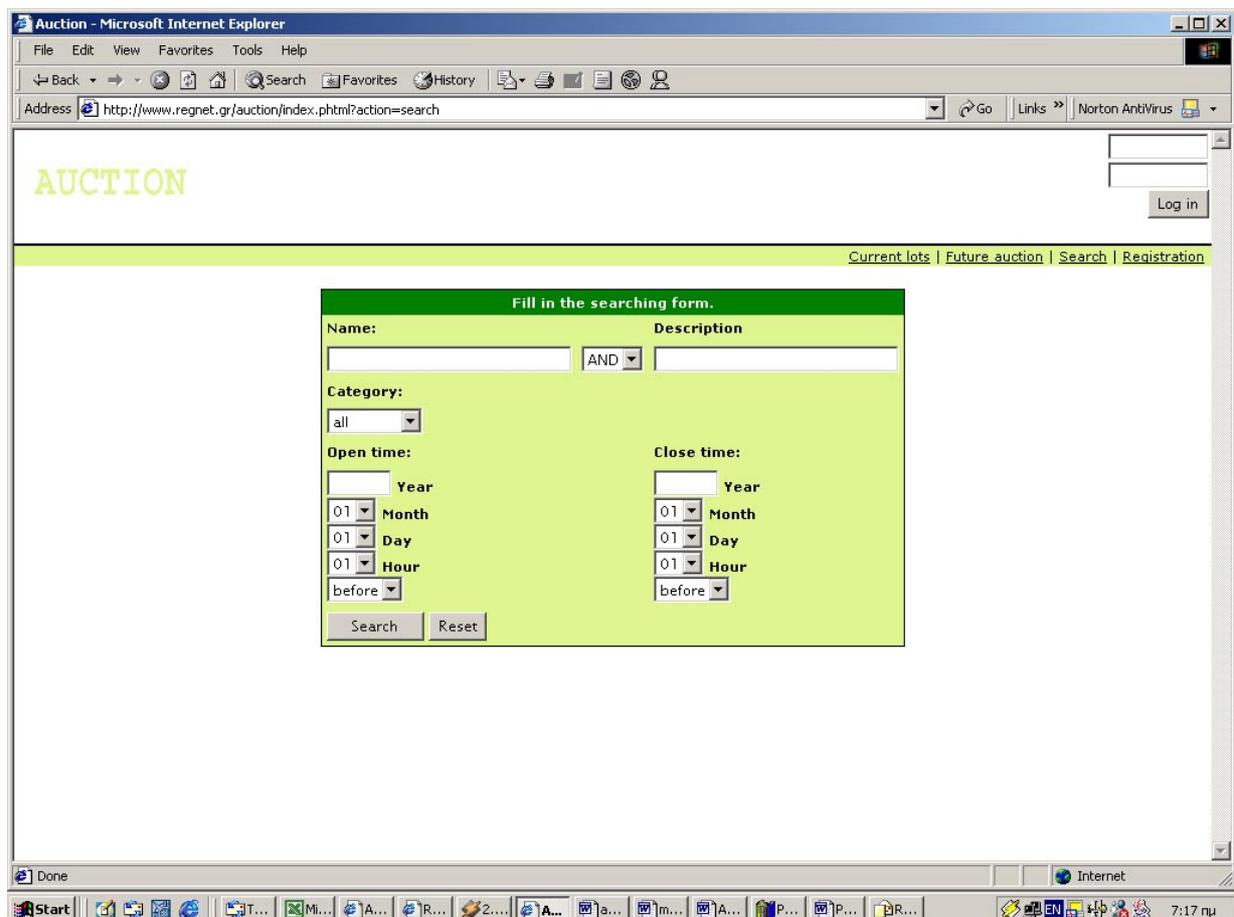


Figure 39: Searching form

12.1.5 1.3.5 PROFILE CHANGE

A Registered user has the option to change all of his registration data. Thus, he/she can update all the information of his/her profile such as the login, the password, the name, the address and other personal information. Also following the link "profile", users they can see, not only their profile information but also all the information concerning their bids (history table) and their items (if they have added any items for sale).



AUCTION

[Current lots](#) | [Future auction](#) | [Search](#) | [Register new item](#) | [Profile](#) | [Exit](#)

[Profile data](#)
[Your items](#)
[Your bids](#)

Edit your profile.

Login & Password

* Login:

* Password:

* Re-Password:

* Name:

* Last name:

Address

* Street:

* City:

State:

Province:

* Post Code:

* Country:

Communication

* Phone:

* E-mail:

URL:

Figure 40: Information about user's profile



The screenshot shows a Microsoft Internet Explorer window displaying the REGNET auction system. The address bar shows the URL: <http://www.regnet.gr/auction/index.phtml?action=prof&act=hist&sale=yes>. The page title is "AUCTION". A navigation bar at the top right contains links for "Current lots", "Future auction", "Search", "Register new item", "Profile", and "Exit". On the left side, there is a sidebar with links for "Profile data", "Your items", and "Your bids". The main content area is divided into two sections: "Your current bids!" and "History of your bids!".

Your current bids!	
You big with item A Tournament Scene	started from 2002/03/01 01:00
You big with item Equestrian Portrait of the Emperor Maximilian	started from 2002/06/01 01:00
You big with item OLYMPIC GAMES	started from 2002/06/06 19:00
You big with item test	started from 2002/06/28 01:00

History of your bids!	
You big with item Arm of Eve	started from 2002/06/01 01:00

Figure 41: User's bids

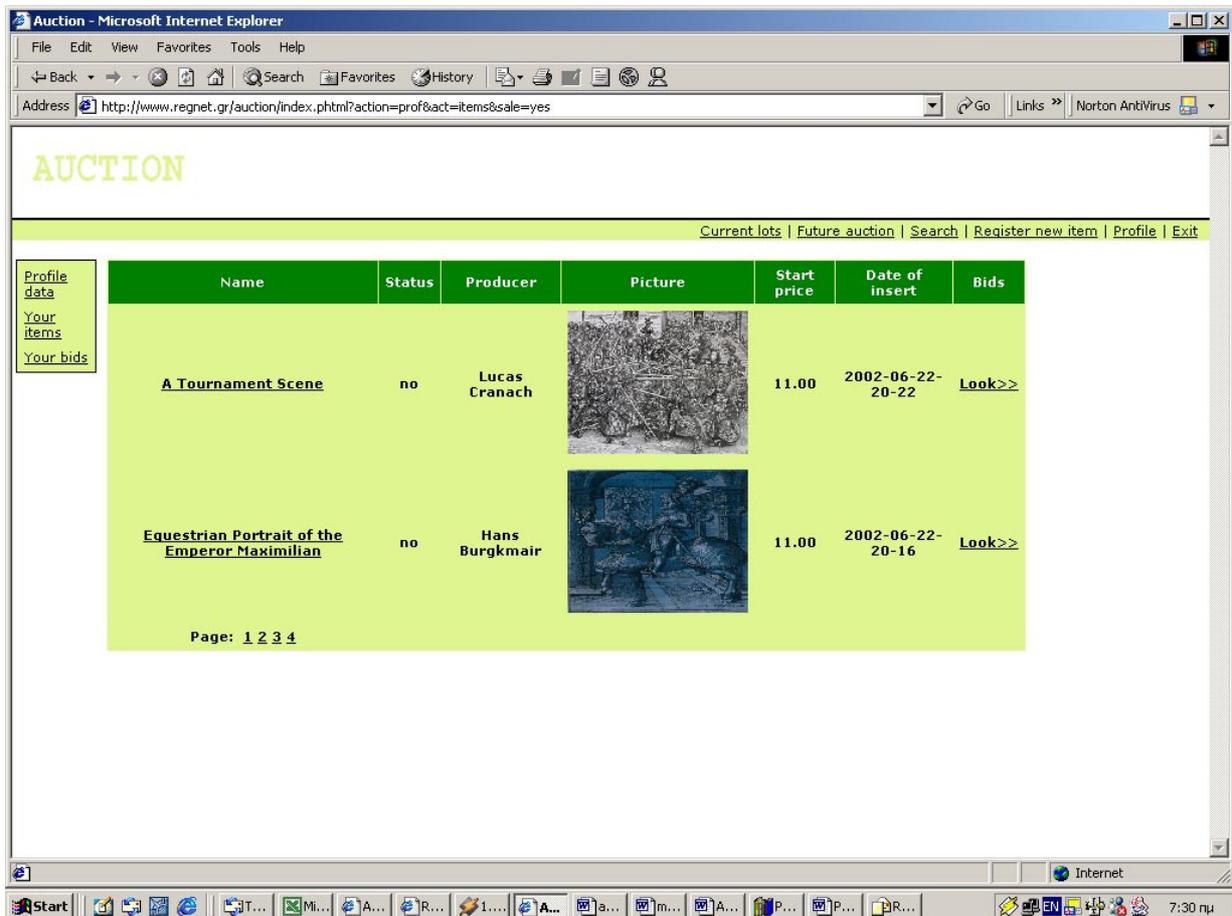


Figure 42: User's items

12.1.6 ADD ITEMS IN THE AUCTION SYSTEM

Furthermore, the system gives users the ability/ functionality to add the items they want and sell them through the auction system. Thus, if they want to add their items in the auction system they have to fill in a specific form, by following the link "register new items", which is provided by the system. (You can see the figure below). Also, here we have to mention that this functionality is available only for registered users who have contacted the administrator of the system and have obtained the necessary permissions.

Sellers are obligated to sell an item at the high bid price. The only exception here is reserve price auctions. Moreover, they cannot sell an item to anybody until the end of the auction. If the high bid does not meet seller's reserve price, then they are under no obligation to sell. In all other cases, they are obligated to sell the item to the highest bidder, and deliver the item within the time period specified.



The screenshot shows a web browser window titled "Auction - Microsoft Internet Explorer". The address bar shows the URL: <http://www.regnet.gr/auction/index.phtml?action=add&sale=yes>. The page content features a green header with the word "AUCTION" in large letters. Below the header is a navigation menu with links: "Current lots", "Future auction", "Search", "Register new item", "Profile", and "Exit". The main content area contains a form titled "Add item for auction!". The form has the following fields and controls:

- Name:** A text input field.
- Category:** A dropdown menu currently set to "all".
- Description:** A large text area with a vertical scrollbar.
- Producer:** A text input field.
- Start price:** A text input field.
- Reserve price:** A text input field.
- Time of start:** A group of four dropdown menus for Year, Month, Day, and Hour.
- Time of end:** A group of four dropdown menus for Year, Month, Day, and Hour.
- Picture:** A text input field followed by a "Browse..." button.

At the bottom of the form are two buttons: "Add item" and "Reset". The browser's taskbar at the bottom shows the Start button and several open applications, with the system clock displaying "8:02 πμ".

Figure 43: New item registration

12.1.7 AUCTION PARTICIPATION

If a user wants to bid an item he/she has to choose a specific item that he/she prefers by following the "GO" link (see figure) from the currents' lots page, in order to move to the next page where he/she should make his/her bids by giving the desirable price in the price field of the form with the lot's characteristics. (See figure). If a user is outbid, or the bid does not meet the seller's reserve price, users are under no obligation to buy.



AUCTION

[Current lots](#) | [Future auction](#) | [Search](#) | [Profile](#) | [Exit](#)

Current Lots

Name	Description	Producer	Picture	Start price	Started from	Bargain?
Poetry of Things	Contemporary Bulgarian Art, Oil/canvas, 40 x 50 cm, 1992	Kalia Zografova		600.00	2002-07-17-15-00	GO >>
Shell	Contemporary Bulgarian Art, Ceramics, 48x38x15cm, 1993	Anastasija Kmetova		1200.00	2002-07-17-16-00	GO >>

Page: [1](#) [2](#) [3](#) [4](#) [5](#) [6](#)

"GO" link

Figure 44: Current Lots



12.1.8 AUCTION SYSTEM ADMINISTRATION

Auction system has its own administrative system. However, this system is unavailable for ordinary users, while permission is available only to the administrator's login and password (see figure). This system allows the administrator to manage the items, users and add different categories to the database (see figure). Nevertheless, it does not allow the administrator to remove existing categories, since a great number of items may exist at the auction and any removal may destroy existing records.

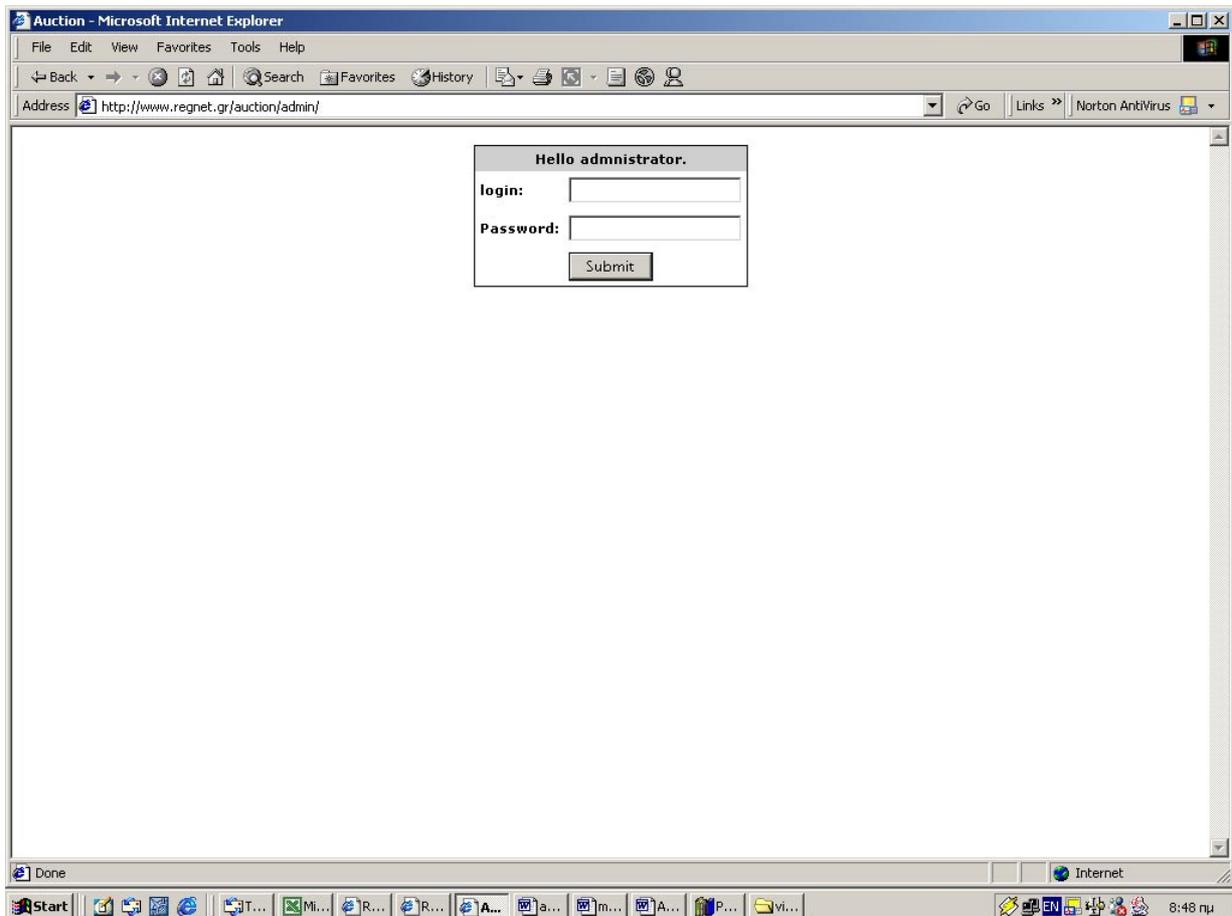


Figure 45: Administration system

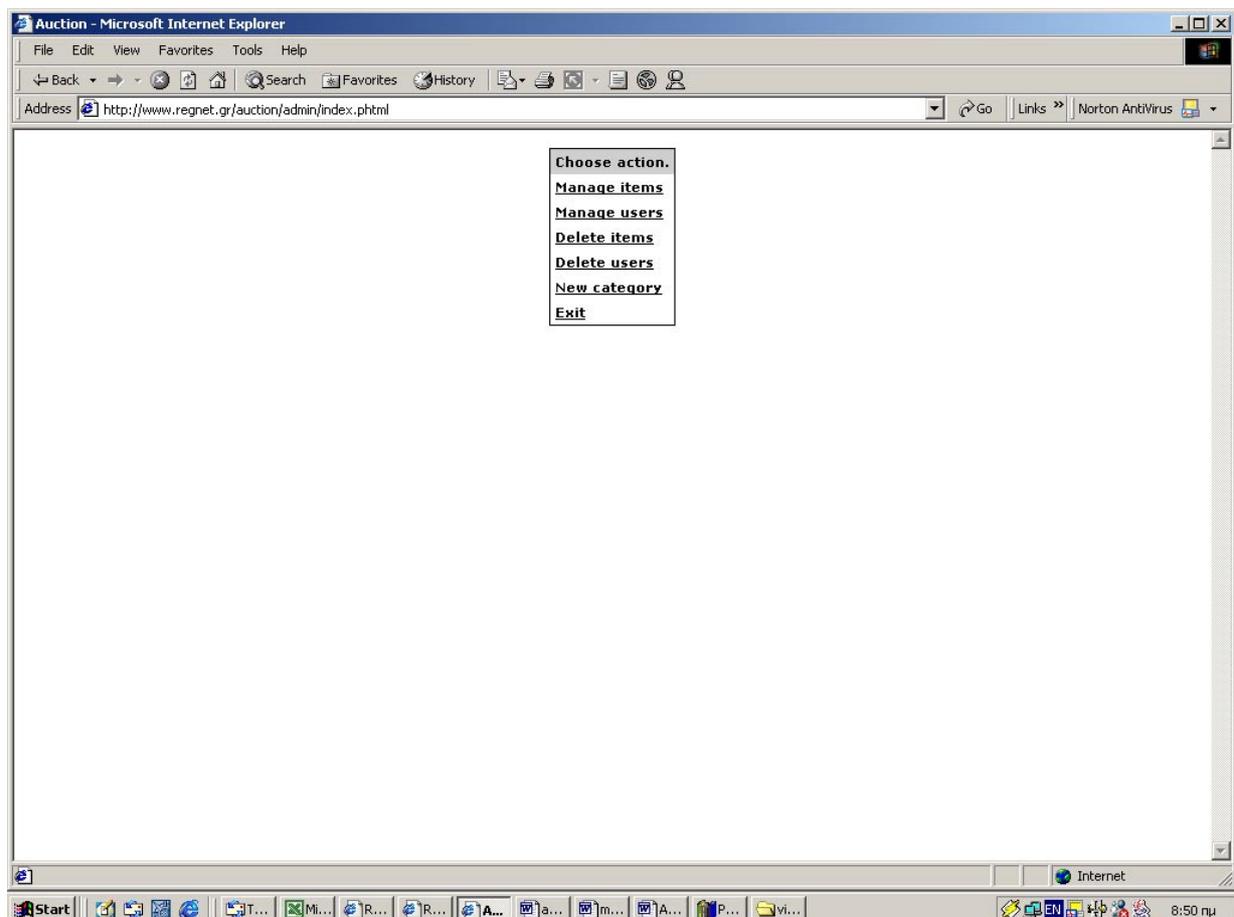


Figure 46: Administration system

12.1.9 MANAGE ITEMS

The system's Administrator has the ability to manage the items which are stored in the auction system. The Administrator can view the list of items presented on the auction, containing items description, dates of registration, start and end dates of the auction. (you can see the figure below).

Furthermore a result field exists, which can take one of the three following values:

Yes – if an item is sold

No – bidding with this lot is not finished yet.

Fin – bidding is finished, but the item was not sold.

Also there is an "Allow" field that enables the administrator to allow/forbid the participation of a lot in an auction (which is accessible, if the auction is not finished).



back

ID	Name	Description	Producer	Start price	Reserve price	Inserted	Started from	End time	Result	Allow?
8	Equestrian Portrait of the Emperor Maximilian	Equestrian Portrait of the Emperor Maximilian Hans Burgkmair (German, 1473 - 1531)	Hans Burgkmair	11.00	134.00	20020622201639	20020601010000	20200301010000	no	yes
10	OLYMPIC GAMES	2004 ATHENS	ATHENS	12.00	13.00	20020622202046	20020606190000	20020801010000	no	yes
9	A Tournament Scene	A Tournament Scene Lucas Cranach (German, 1472 - 1553)	Lucas Cranach	11.00	13.00	20020622202218	20020301010000	20200301010000	no	yes
11	Henri II, King of France, at the age of 28	Henri II, King of France, at the age of 28	Nicolo della Casa	124.00	155.00	20020622202511	20021005010000	20021010010000	no	yes
12	The Nativity with the Dream of Joseph	The Nativity with the Dream of Joseph Alternate title: The Nativity	Parmigianino	122.00	125.00	20020622202634	20021008080000	20200301010000	no	yes
7	Arm of Eve	During a stay in Venice during 1507-8, Dörner came to admire the use of blue paper by contemporary Venetian artists and began to use	Albrecht Dörner	112.00	113.00	20020622210455	20020601010000	20020622203000	no	yes

Figure 47: List of auction's items

12.1.10 DELETE ITEMS

The administrator has the ability to delete the items he/she wants. The "Delete?" field enables the administrator to delete the marked lots from the list, however this option is unavailable, if the bidding has not finished (you can see figure 15).

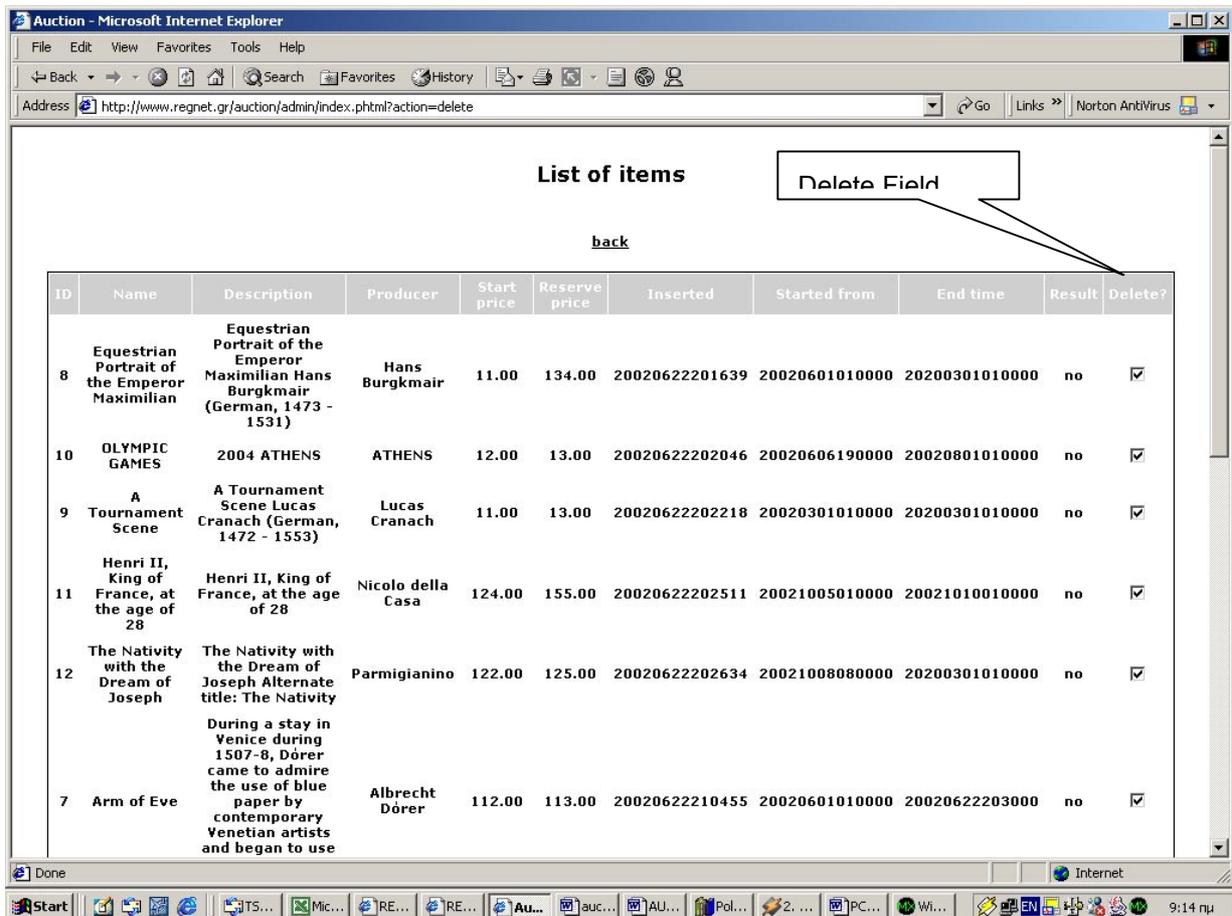


Figure 48: Delete items

12.1.11 MANAGE USERS

The Administrator of the system can view the list of registered users, which contains information about the profile of a user like the login, the password, the name and all the other specific information.

Furthermore, the field "Sale" exists, which can take the value yes or no in order to enable the administrator to allow or forbid a user to become a seller and hence sell his/her own items (You can see figure 16).

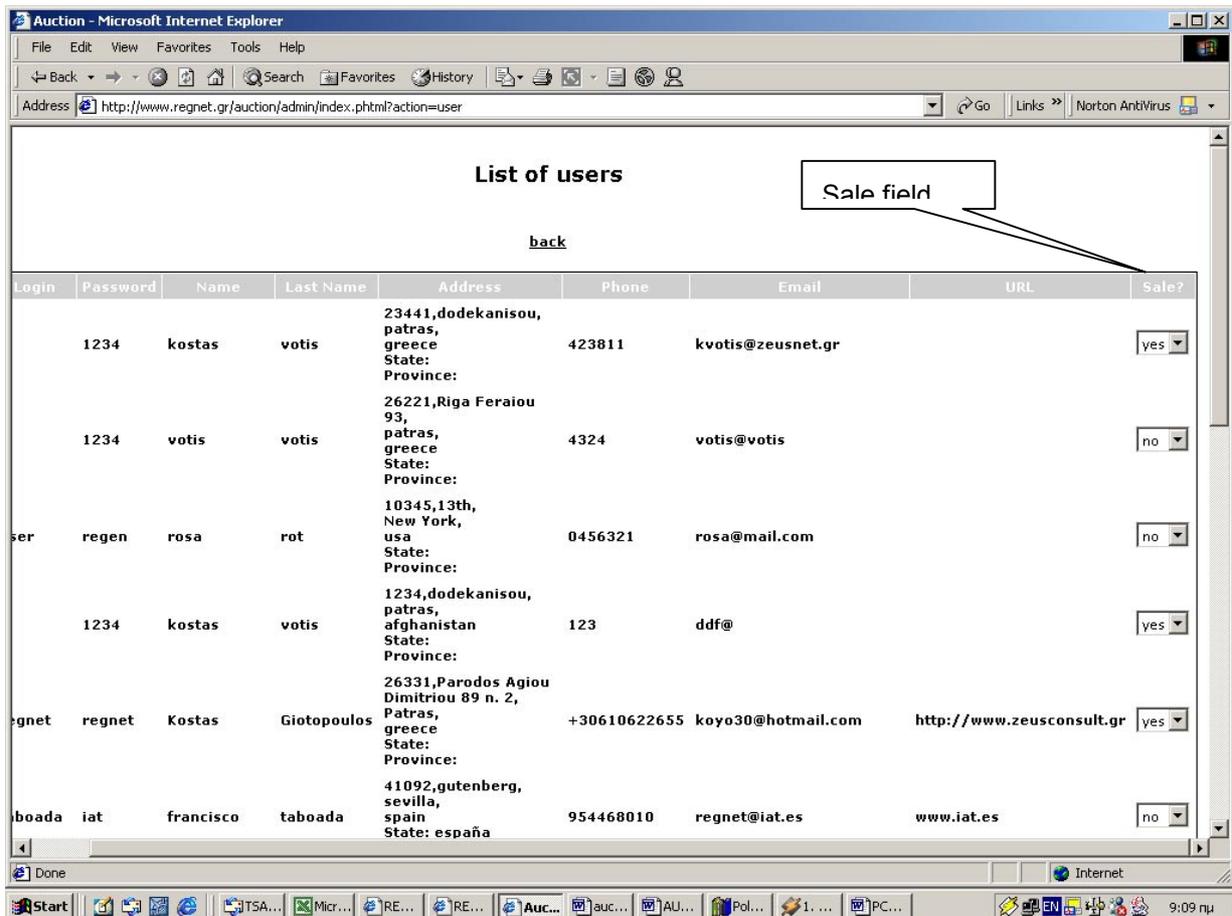


Figure 49: List of auction's users

12.1.12 DELETE USERS

Using the "Delete users" reference, the administrator can view the list of registered users, which contains all the information about users and also the "Delete" field, which enables the administrator to delete a user from the auction database.

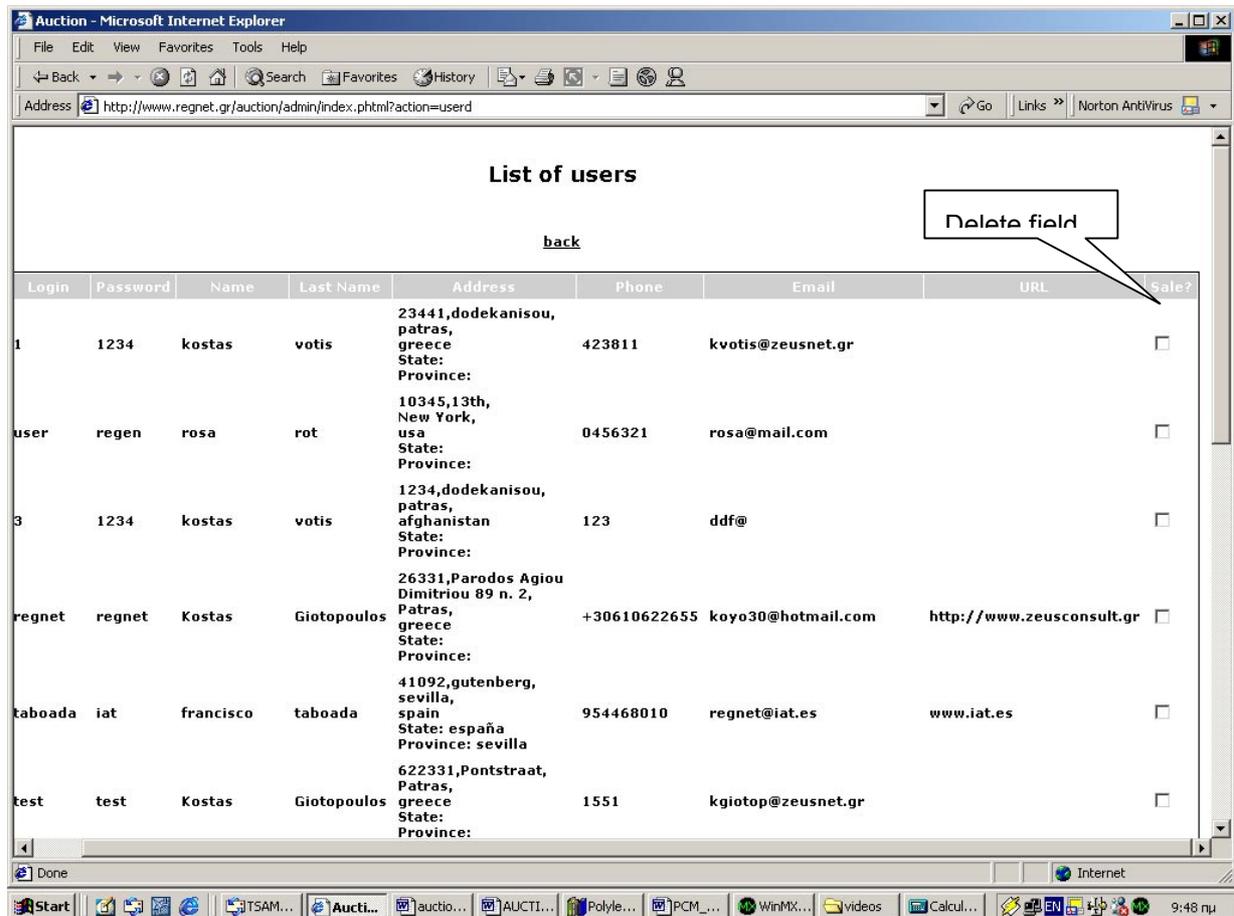


Figure 50: Delete users

12.1.13 ADD NEW CATEGORY

The Administrator has also the ability to add new categories of items in the auction system (you can see the figure below).

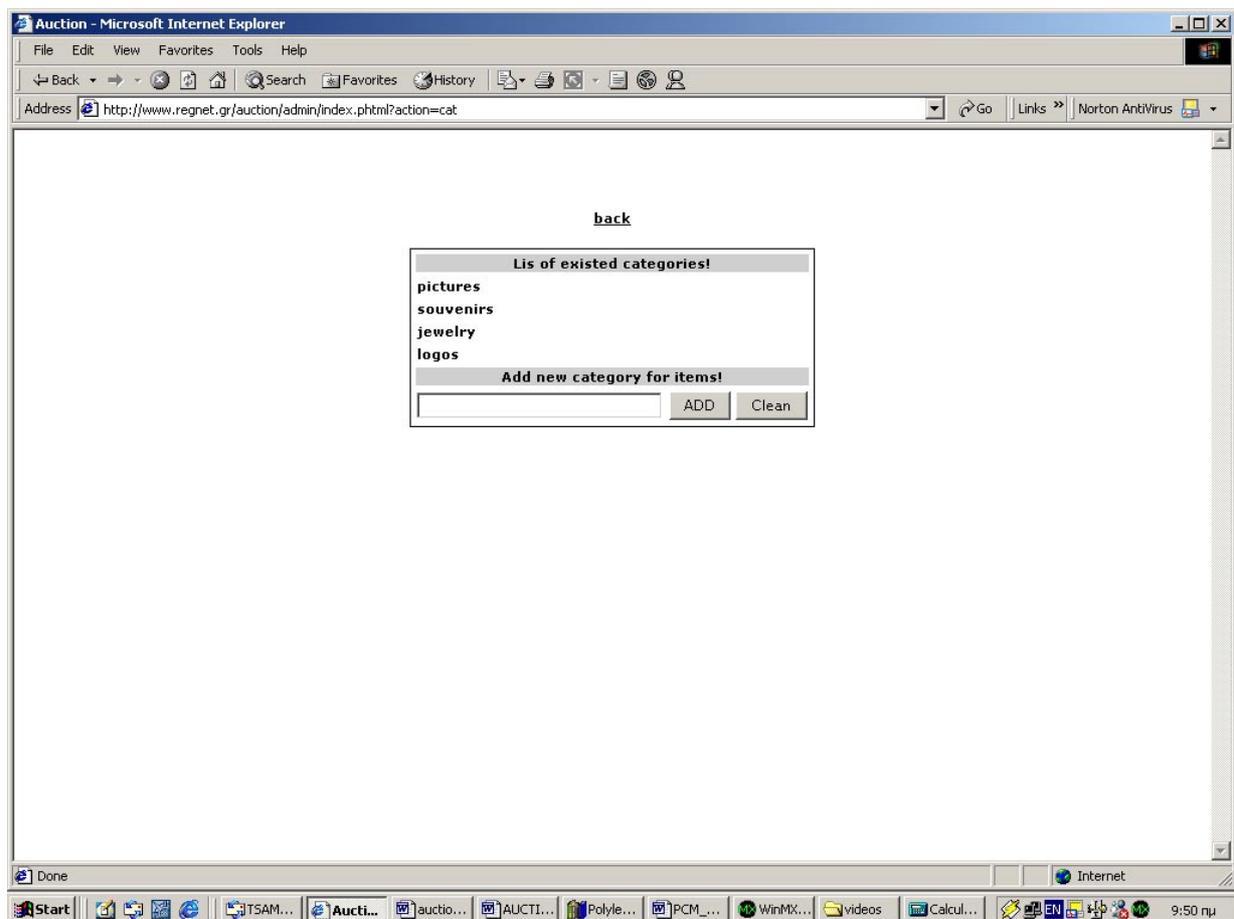


Figure 51: Add new category of items in the auction system

12.1.14 EXIT

Registered users have the option to undo their authorization by pressing the Exit link in the general menu of the site.

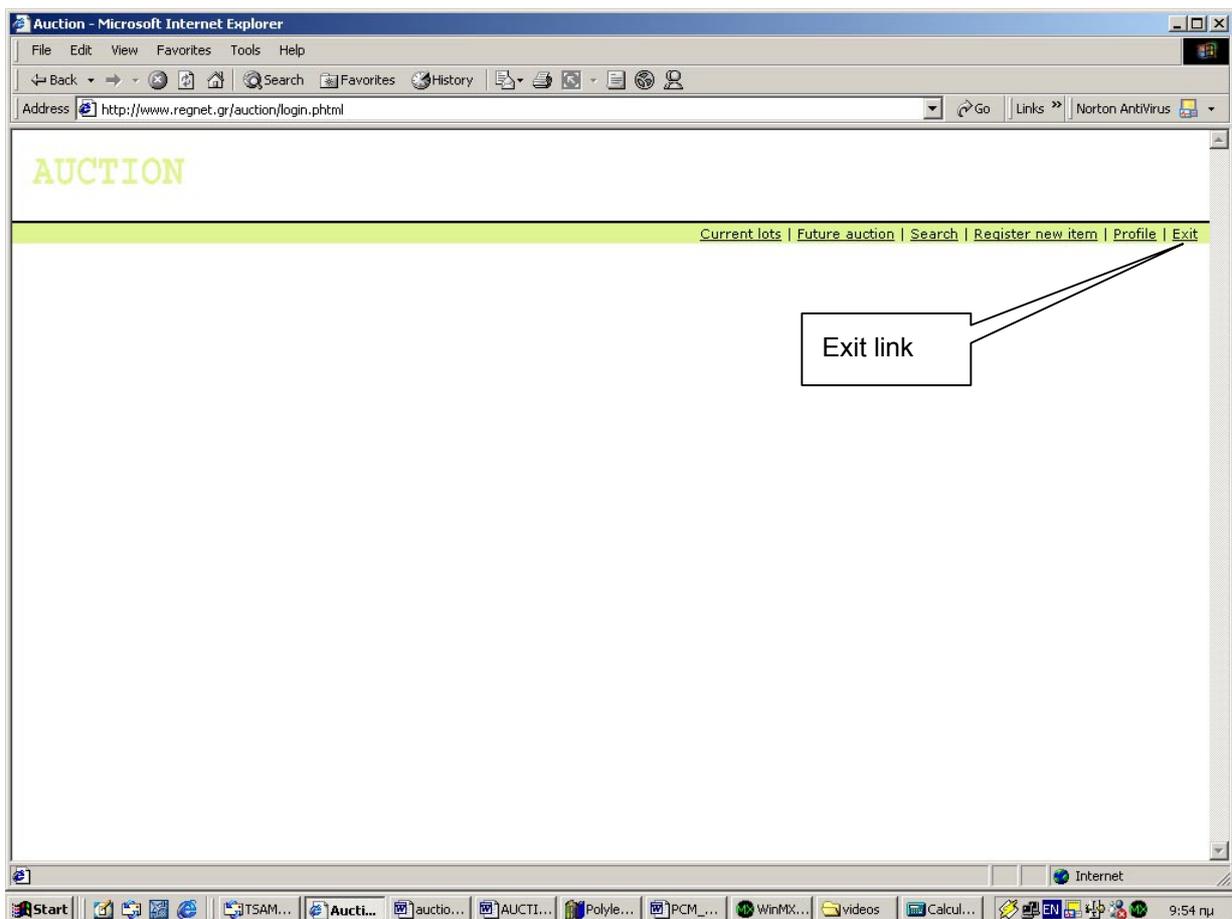


Figure 52: Exit

12.2 Delivery

The Delivery web service is invoked when in the procurement application a user click on the “calculate shipping cost” button in the shopping basket page.

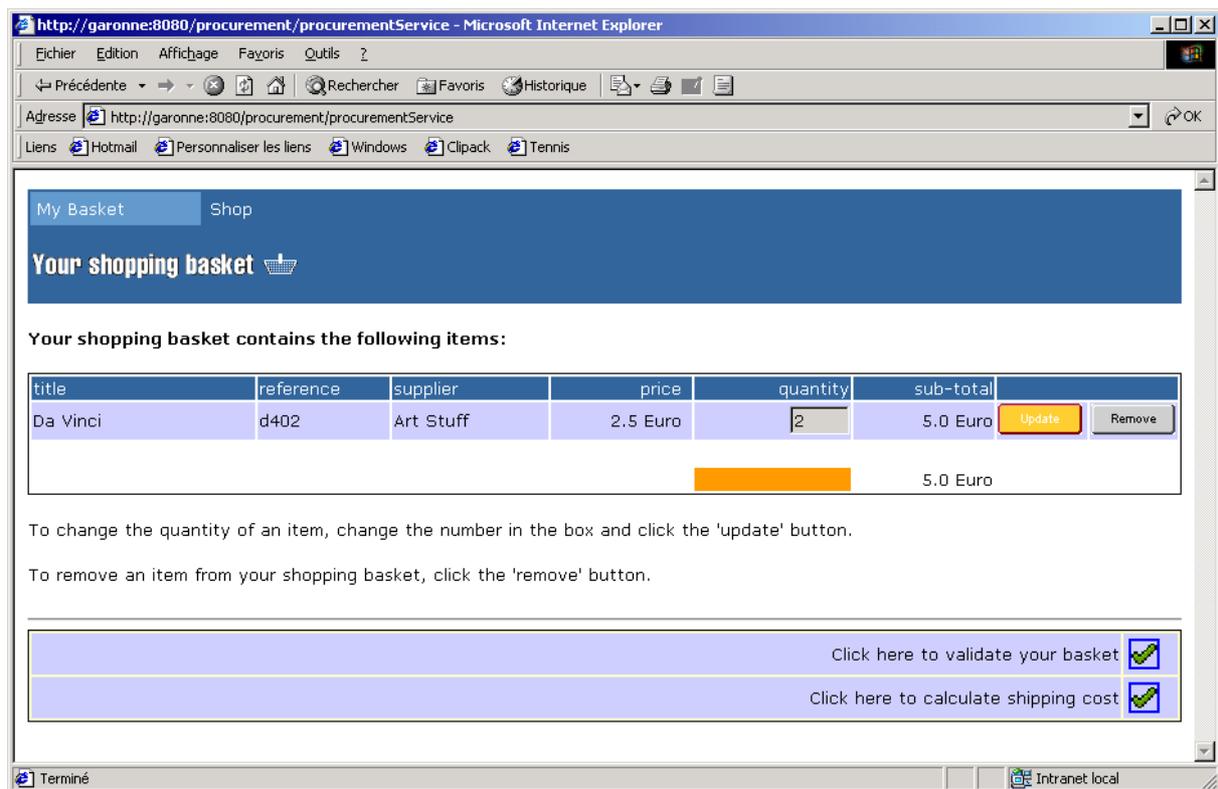


Figure 53: Screenshot delivery system

12.3 Index+ gateway

The index+ gateway webservice component has to be integrated into the Cultural Heritage Data Management node in order to managing retrieval of all digital surrogates stored into the Index+ DB.

Hereafter is presented the screen shot of the webservice installed at Space premises. The public address to link the web service is <http://80.207.84.129:8088>

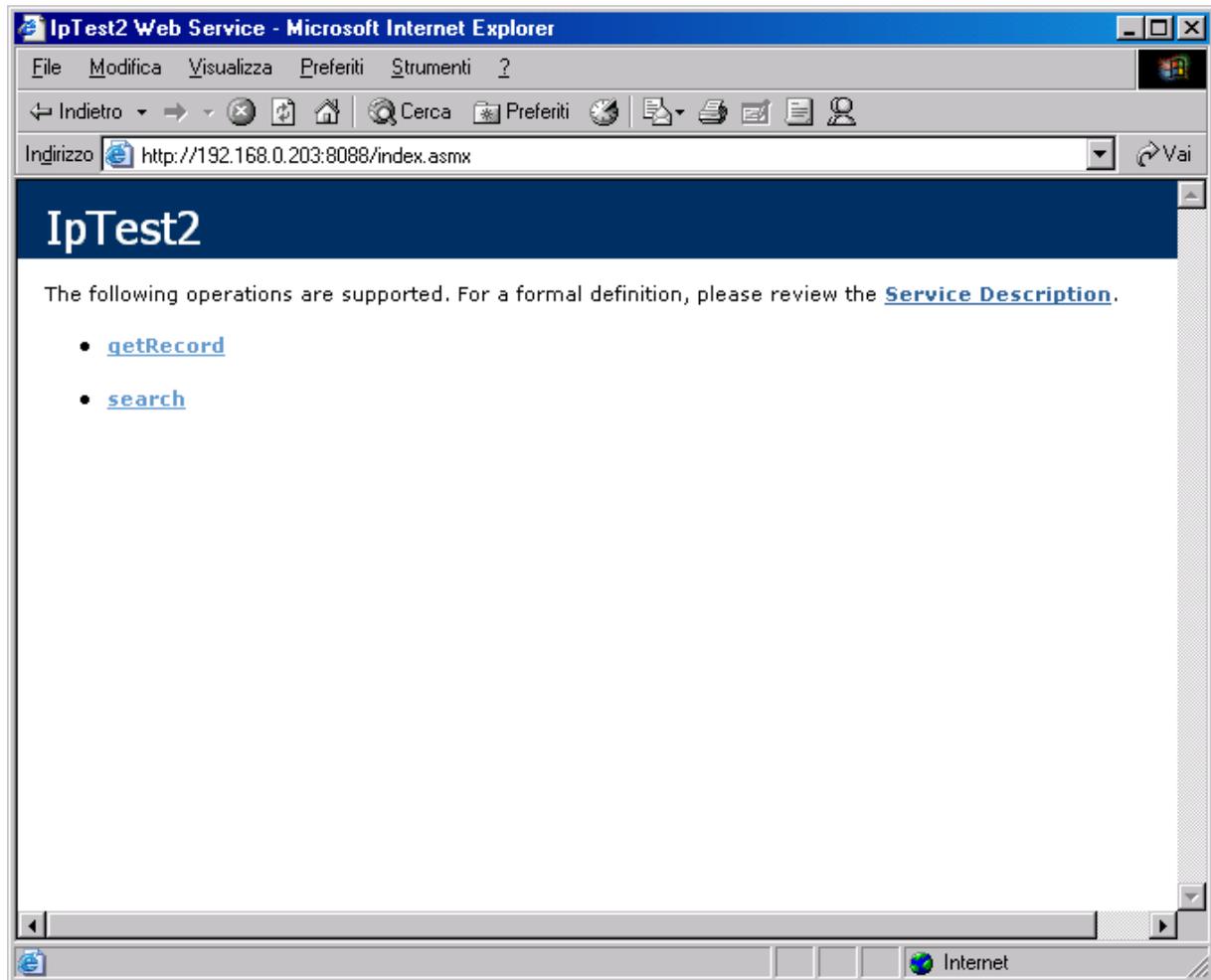


Figure 54: Screenshot web services Index+

As said in section 12.3 the Index+ gateway provides two simple services for searching and retrieving records in the Index+ DB:

- Search
The main method of the Index+ gateway. Returns query hits and the records found respectively. The result of search the key code and size of the array containing records found.
- GetRecord
Returns the records specified in the record array by position number (zero-based).

In this way the RegNet platform will be able to explore several catalogues adopted within the clustered OpenHeritage (OH) project and in general to browse and retrieve data from Index+ DB.

The Index+ DB used for testing the webservice contains images coming from :

- Musée de Bourgoin-Jallieu (France)
- Musée des Tissus de Lyon (France)
- Museum of Chini Manufactures, Borgo San Lorenzo (Italy)

On the following is presented an example for better understand the mechanism to browse and retrieve records from part of the Open Heritage collection.

For instance, in the next figure is presented a search request for “vase” term. The index+ gateway will discover all the records containing this term.

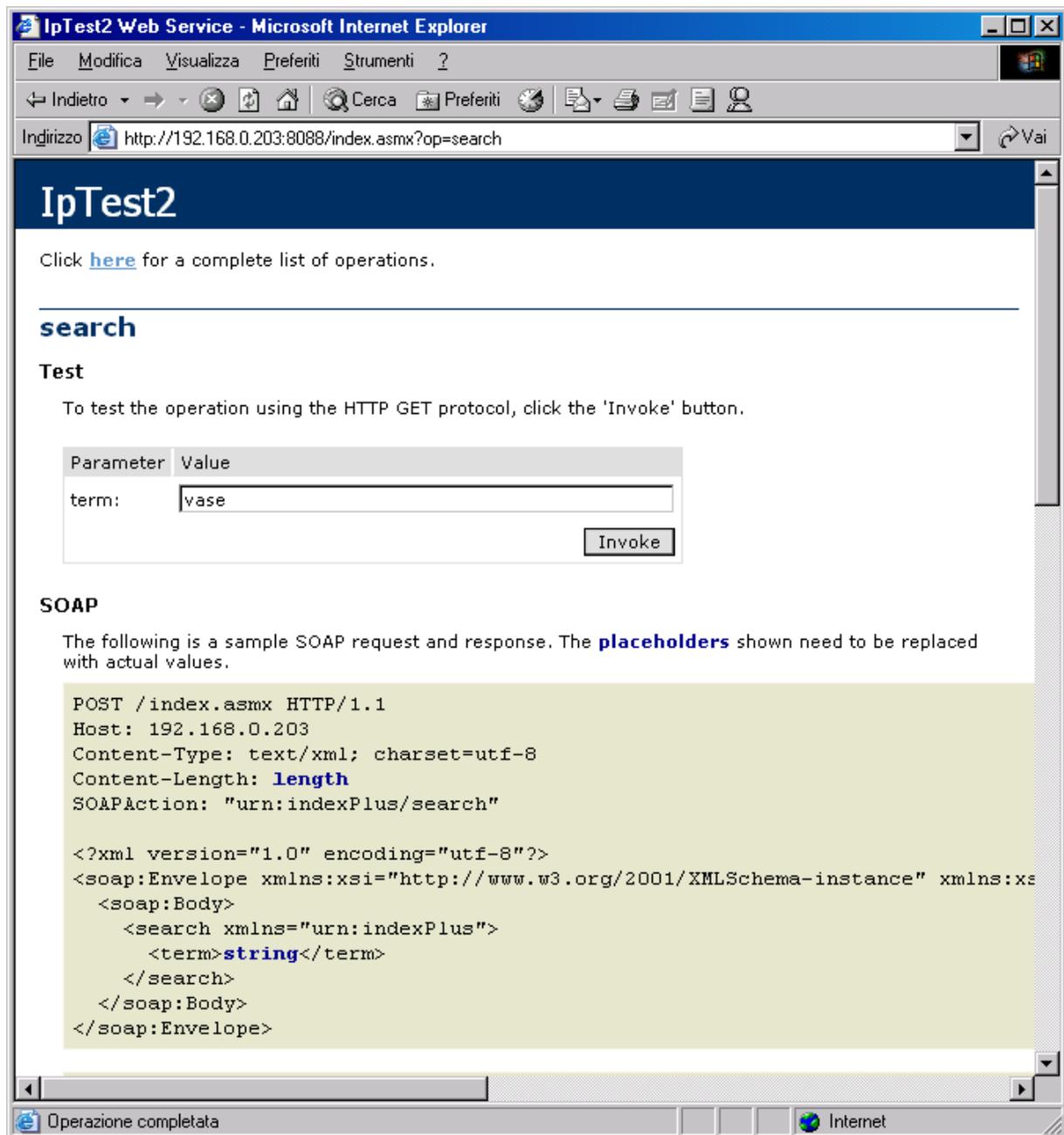


Figure 55: Serach request Index+

The response of this kind of query, will appear like in the following screenshot.

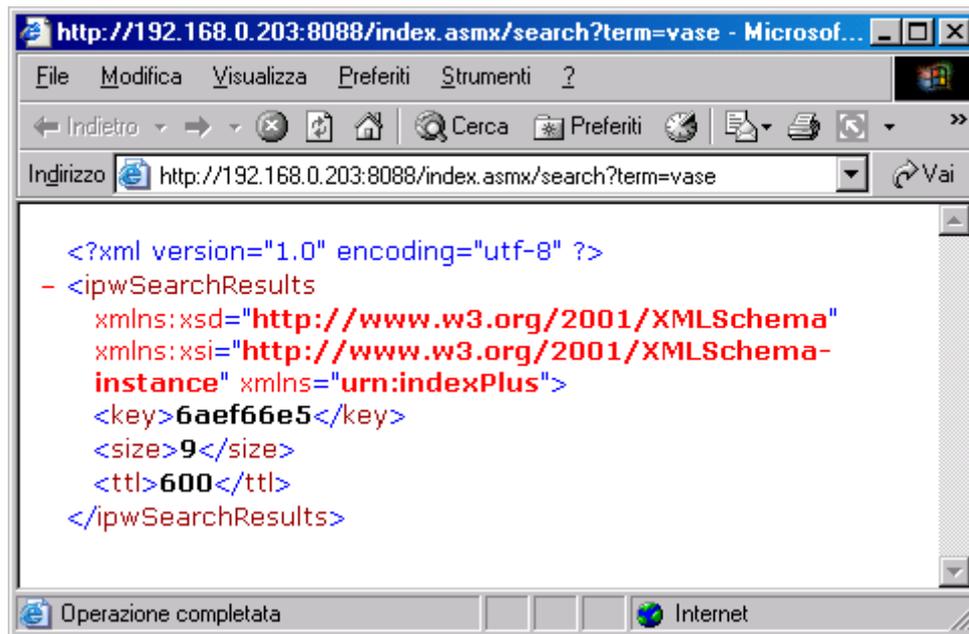


Figure 56: Search request Index+

The response message is an XML file containing 3 significant fields:

- **Key** is the generated code that locate the array of records found. This key will allow to recover the single record in the array when reported on the getRecord method.
- **Size** is the array dimension. This field indicates the number of resources present in the array of records found.
- **TTL** is the search Time Span.

With these data we can go to get the related record.

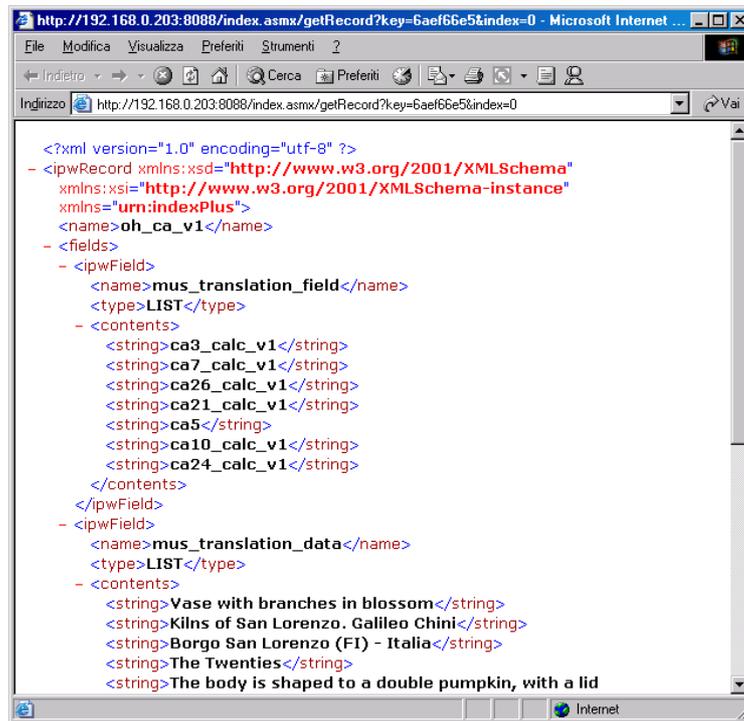


Figure 57: Screenshot Index+ record

```
<?xml version="1.0" encoding="utf-8" ?>
<ipwRecord xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="urn:indexPlus">
  <name>oh_ca_v1</name>
  <fields>
    <ipwField>
      <name>mus_translation_field</name>
      <type>LIST</type>
      <contents>
        <string>ca3_calc_v1</string>
        <string>ca7_calc_v1</string>
        <string>ca26_calc_v1</string>
        <string>ca21_calc_v1</string>
        <string>ca5</string>
        <string>ca10_calc_v1</string>
        <string>ca24_calc_v1</string>
      </contents>
    </ipwField>
    <ipwField>
      <name>mus_translation_data</name>
      <type>LIST</type>
      <contents>
        <string>Vase with branches in blossom</string>
        <string>Kilns of San Lorenzo. Galileo Chini</string>
        <string>Borgo San Lorenzo (FI) - Italia</string>
        <string>The Twenties</string>
        <string>The body is shaped to a double pumpkin, with a lid
          and a foot shaped as a frustum of a cone. The vase is decorated with
          branches in blossom and clouds.</string>
        <string>Polished turquoise majolica</string>
        <string>Museum of Chini Manufactures, Borgo San Lorenzo (FI) -
          Italia</string>
      </contents>
    </ipwField>
    <ipwField>
      <name>mus_translation_language</name>
      <type>LIST</type>
```



```
<contents>
  <string>English</string>
  <string>English</string>
  <string>English</string>
  <string>English</string>
  <string>English</string>
  <string>English</string>
  <string>English</string>
</contents>
</ipwField>
<ipwField>
  <name>rec_id</name>
  <type>TEXT</type>
  <contents>
    <string>192</string>
  </contents>
</ipwField>
<ipwField>
  <name>ca1</name>
  <type>TEXT</type>
  <contents>
    <string>CHINI_019</string>
  </contents>
</ipwField>
<ipwField>
  <name>ca3</name>
  <type>TEXT</type>
  <contents>
    <string>Vaso con rami fioriti</string>
  </contents>
</ipwField>
<ipwField>
  <name>ca7_1</name>
  <type>LIST</type>
  <contents>
    <string>Manifattura Fornaci di San Lorenzo. Galileo Chini</string>
  </contents>
</ipwField>
<ipwField>
  <name>ca7_2</name>
  <type>LIST</type>
  <contents>
    <string>Maker</string>
  </contents>
</ipwField>
<ipwField>
  <name>ca21_10</name>
  <type>LIST</type>
  <contents>
    <string>Anni Venti</string>
  </contents>
</ipwField>
<ipwField>
  <name>ca5</name>
  <type>TEXT</type>
  <contents>
    <string>Corpo a doppia zucca con coperchio e piede troncoconico. È
      decorato con rami fioriti e nuvole.</string>
  </contents>
</ipwField>
<ipwField>
  <name>ca10_4</name>
  <type>LIST</type>
  <contents>
    <string>Maiolica policroma a lustri</string>
  </contents>
</ipwField>
```



```
<ipwField>
  <name>ca26_1</name>
  <type>LIST</type>
  <contents>
    <string>Borgo San Lorenzo (FI) - Italia</string>
  </contents>
</ipwField>
<ipwField>
  <name>ca54</name>
  <type>TEXT</type>
  <contents>
    <string>CHINI-197</string>
  </contents>
</ipwField>
<ipwField>
  <name>ca24_1</name>
  <type>LIST</type>
  <contents>
    <string>Museo della Manifattura Chini, Borgo San Lorenzo (FI) -
      Italia</string>
  </contents>
</ipwField>
</fields>
</ipwRecord>
```

Above has been listed the XML file returned from a "vase" research (record 0).

The meaning of fields recovered is resumed on the following table. For each ipwField code there is a text file that describe the meaning of the code. To access this help just point the browser to <http://80.207.84.129:8088/help/caNNN.txt> where NNN is the code number



ID	Description	CIMI-1	Comments	Field type	Index+ code
Technical					
museum_record_id	Museum record Identifier		Record ID in the museum (in its own system)	Text	
collection_name	Collection name	collection	Orpheus package, representative of the collections (according to CHOL/museum agreement)	Text	
Copyright string and restrictions					
right_note	Rights note		Information on the rights	Text	ca36.2f
rep_right_owner	Reproduction right owner		museum name ?	Text	ca36.3f
photographer	Photographer	"unknown"	Name of the photographer (of the image, not the artist).	Text	
obj_auth_right_owner	Author rights owner	"unknown"		Text	ca36.3f
rep_restrictions	Reproduction restrictions	"not specified"		Text	
copyright	Copyright	copyrightRestriction	Mention of the copyright	Text	
High definition images					
hd_museum_available	HD Image Museum availability		Availability of the HD in the supplier. "YES" or "NO"	Boolean	
hd_museum_format	HD Image format			Text	
hd_museum_size	HD Image size		in Bytes	Integer	
hd_museum_file_name	HD Image name		if hd image not delivered	Text	
hd_path	HD Image path		if hd image not delivered	Text	
hd_scan_author	Scan author		Author of the numerical one	Text	
hd_scan_date	Scan date		Date of digitalization	Date	
hd_scan_type	Scanner type		Scanner type	Text	
hd_editing	Image editing permitted		Image editing permitted	Boolean	



obj_materials_info	Object Materials additional information		for the museums in conformity with OpenMuseum, the values of this multi-valued field will be fed by the converter from : Material components Material geographic origin Material name	Text	ca10.2 + ca10.3 + ca10.4
obj_color	Object color			Text	ca11
obj_date	Object date		Free text ;for the museums in conformity with OpenMuseum, the value of this field will be generated by the converter, by concatenation of the following infos (with separator to be defined) : Date entries Event date Precise date Interval date Start date End date Before of after JC Date precision Date certainty Date method Added date show	Text	ca21.1 + ca21.2 + ca21.3 + ca21.4 + ca21.4.1 + ca21.4.2 + ca21.5 + ca21.6 + ca21.7 + ca21.8 + ca21.9
obj_named_period	Named period		for the museums in conformity with OpenMuseum, the values of this multi-valued field will be fed by the converter from : Named period	Open List	21.10
obj_history	Object history			Text	ca22



obj_assoc	Object association		Places, people, associated events. Free text ;for the museums in conformity with OpenMuseum, the value of this field will be generated by the converter, by concatenation of the following infos (with separator to be defined) : Associated type Associated concept Associated cultural context Associated date Associated event type Associated event name Associated object Associated person type Associated person name Associated place Other associations	Text	de ca23.1 à ca23.12
obj_owner	Object owner	owner	The owner of the object is not always the museum where the object is.	Text	ca 24.1
obj_use	Object use			Text	ca25
obj_use_function	Object use function			Text	ca25.1
geo_prod_place	Production or discover place	placeOfOrigin	Place of production or discovery; for the museums in conformity with OpenMuseum, the values of this multi-valued field will be fed by the converter from: Production place	Text	ca26.1
geo_place_info	Geographic presentation		Free text ;for the museums in conformity with OpenMuseum, the value of this field will be generated by the converter, by concatenation of the following infos (with separator to be defined) : Geographic certainty Geographic remark Geographic indication type Geographic coordinates Geographic code	Text	ca26.2 + ca26.3 + ca26.4 + ca26.5 + ca26.6
prod_type	Production type			Text	ca 28



geologic_name	Geologic name	agePeriod		Text	ca 30
rec_type	Record type			Text	ca 39
museum_keywords	Museum keywords		Keywords delivered by the museums (according to their internal classification). For the museums in conformity with OpenMuseum, the values of this multi-valued field will be fed by the converter from : Position Concept Date Description Event type Event name Object Person Person name Described activity Described place	Text	ca46
obj_style	Object style	stylePeriod		Text	ca50
tech_type	Technic type	processTechnique		Text	ca71.1

The following screenshot is an example of an image retrieved by the record informations.

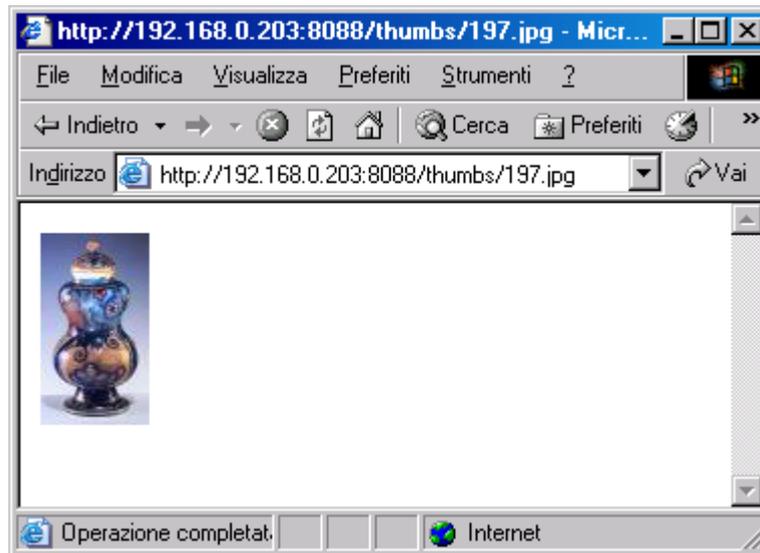


Figure 58: Retrieved image Image+

12.4 Cooperative thesauri

Thesauri are stored in the ontology database.

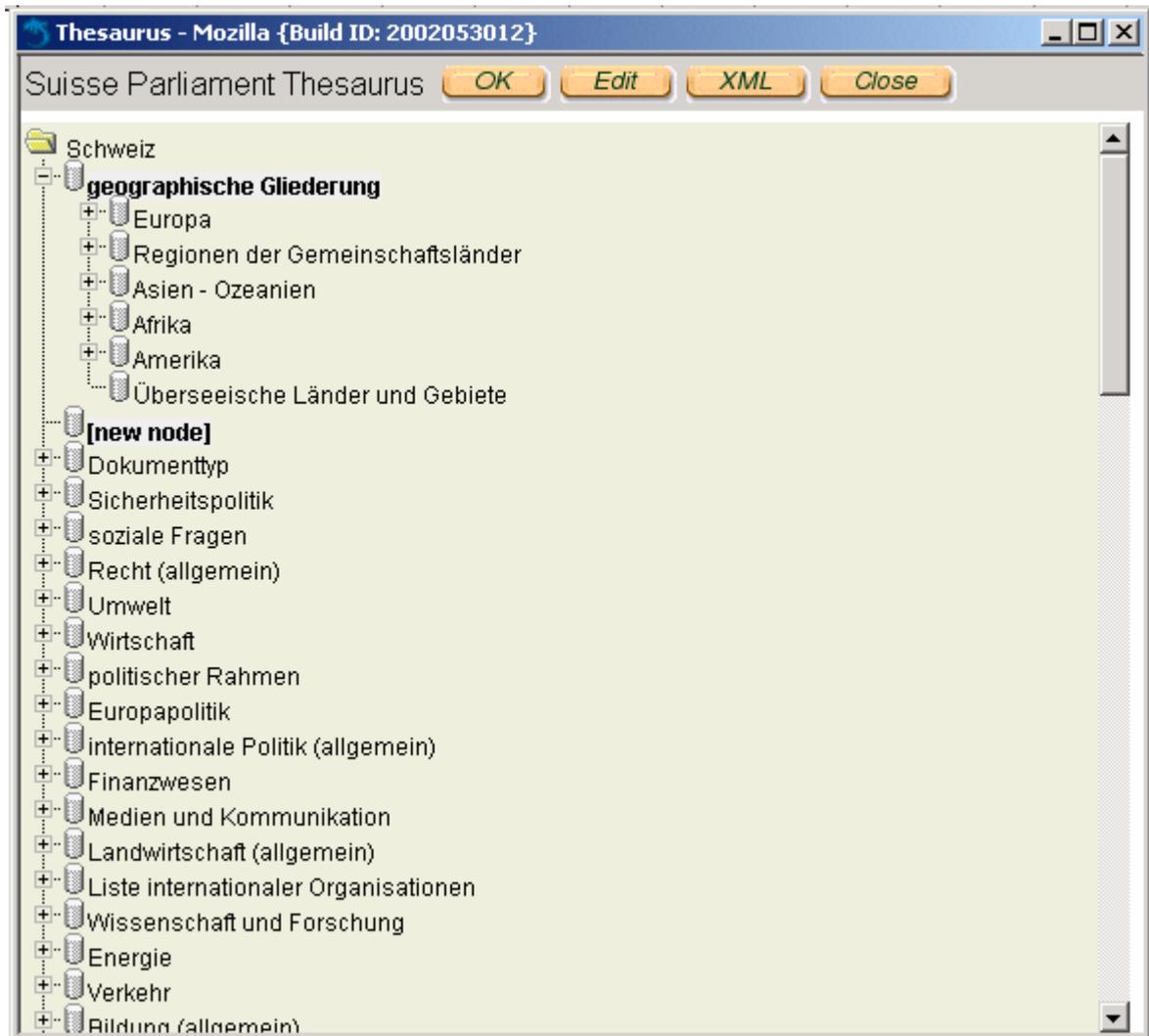


Figure 59: Screenshot thesaurus management

12.4.1 Navigation

Navigate as you do with the Windows Explorer.

The terms “parent node” and “child node” describe the hierarchy/dependence of terms. For example “geographische Gliederung” is the parent node to “Europa” and “Europa” is parent node to “Westeuropa”. Vice versa “Westeuropa” is child node to “Europa” and “Europa” is child node to “geographische Gliederung”.



Figure 60: Screenshot thesaurus management - navigation

12.4.2 Adding/updating/deleting entries

Click EDIT.



Figure 61: Screenshot thesaurus management - editing

Click on a term.

For this feature use the symbols at the right side

The blue arrows help to move the thesaurus term to another level or order (parent-child - just try yourself, this is a testversion).

The other symbols are for copying, deleting, adding a new term and saving.

Just one comment: add a new term at the first level of the thesaurus: add a term at a child node and move the new entry with the blue arrow to the first level.

To connect the thesaurus term with e.g. object data:

Add to existing metadata: edit the record click on the yellow/orange button with the + at the right side of the index card name OBJECT and DUBLINCORE, choose document type THESAURUS, click on the lookup list and select a term, add and save/apply.

Create a new record: choose document type OBJECT, enter data, add a new document type (is internally written into the same record!) THESAURUS, proceed as described above.

12.5 "Topic Maps Generator" tool

Topic Maps Generator 2.0 is better viewed with 1024x768 screen resolution and Internet Explorer 5.0 or above (Netscape Communicator does not correctly display the tree view of the XTM).

It can be found at: <http://160.40.50.22:8800/servlet/org.regnet.ontology.XTMLoader>

Figure below, shows the main screen of the tool.

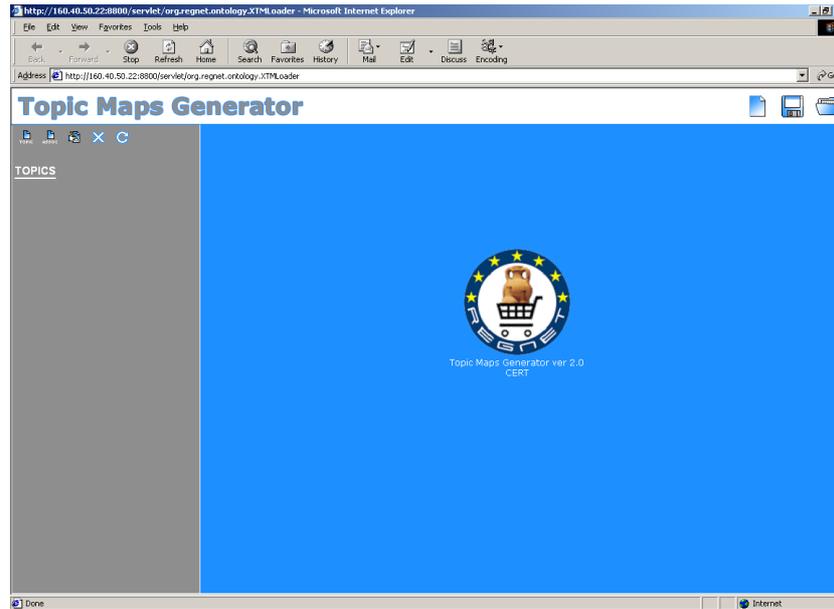


Figure 62: Main Screen Topic Map Generator

Figure 1.

Figure 2.

Every time a new Internet browser is loaded with the tool's URL, a new session is created. This means that multiple users can use the tool simultaneously since concurrent requests are fully supported by the tool. The latter, also means that if the page is just refreshed (F5), the session (and data) are not lost.

Functionalities

There are two major groups of buttons:



With the use of the first group, (upper-right), the user can manipulate XTM files, and communicate with the REGNET Knowledge Base



With the use of the second group, (left), the user can manipulate the items that compose an XTM (topics, associations, occurrences, etc).

Remember that every XTM contains multiple topics and/or associations.

XTM files Manipulation

Create New XTM



When the tool is loaded, automatically a new empty XTM file is created. By clicking this button and after the confirmation screen that warns for the loss of all items that compose the currently edited XTM, a new empty XTM file is created.

Save/Export/View XTM





This button has three different functionalities. When it is being clicked, a screen with a radio button choice appears, exhorting the user to select one of the following features:

Save XTM to the Knowledge Base: The user should provide a name for the XTM file in progress, which will be then stored to the Knowledge Base.

Export XTM to file: The user can download the XTM file in progress, by clicking on the link.

View XTM on screen: The user can view and also expand or contract every branch of the tree view of the XTM file in progress.

Open/Import XTM



This button has two different functionalities. When it is being clicked, a screen with a radio button choice appears, exhorting the user to select one of the following features:

Open XTM from the Knowledge Base: A list with the available XTM files from the Knowledge Base appears where the user can select a file. The topics and the associations that are contained into the selected file are then extracted and added to the currently edited XTM.

Import a local XTM file: The user can select a local file to be uploaded and stored to the Knowledge Base.

If the mouse pointer is left over a button for a second, an explanatory message appears, showing the use of the button.

Topics and Associations

The left gray side of the tool is being used for the manipulation of the topics and the associations that compose an XTM file.

The group of buttons on the top is used to:



Create a new Topic



Create a new Association



Edit Topic / Edit Association



Delete Topic / Delete Association



Refresh Topics List / Refresh Association List

The functionality of the Edit/Delete/Refresh change according to the selected view (topics or associations)

The left gray side also contains the lists of topics and/or associations as shown in figure below.



Figure 3. Associations and Topics

Create New Topic



When the user clicks on the button, the screen shown in figure, appears on screen:

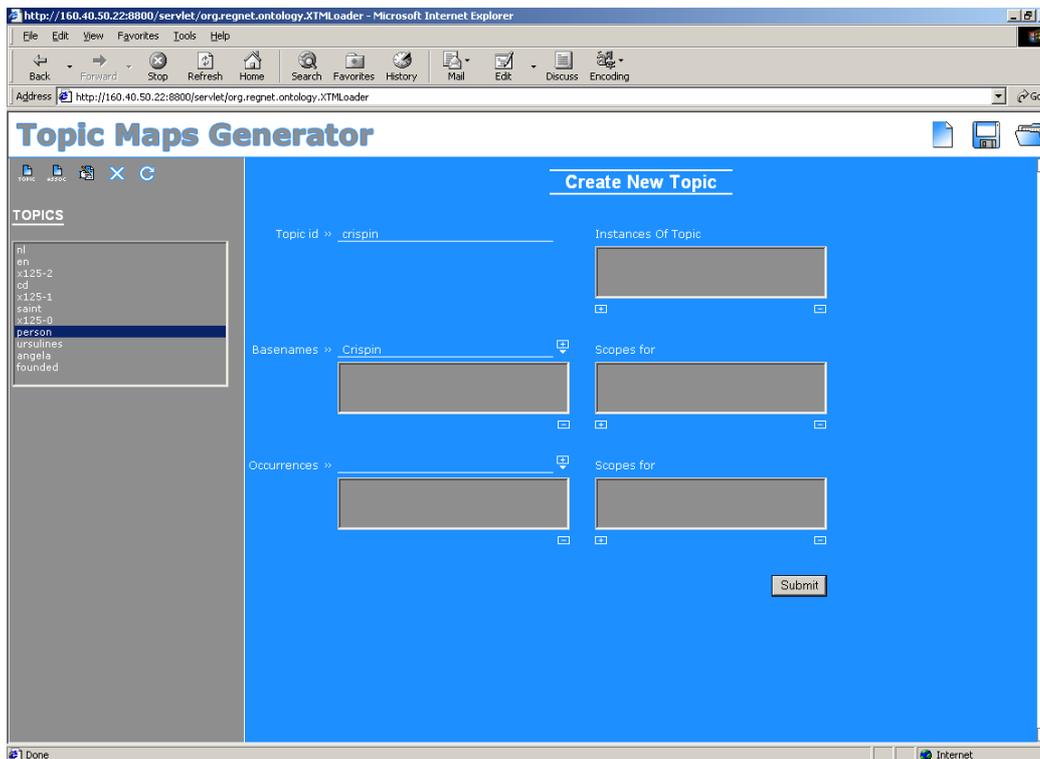


Figure 63: New Topic screen

Figure 4.

Figure 5.

In the working frame (blue), the user can type:

A topic id (if the id is not unique, the tool prompts the user)

Multiple basenames (the button  should be clicked to add the basename in the list)

Multiple occurrences (the button  should be clicked to add the occurrence in the list)

Multiple Instances of topic (in order to add a new instance, a topic should be selected from the topics list and the button  should be clicked)

Multiple Scopes for basenames / occurrences (in order to add a new scope, a topic should be selected from the topics list, a basename / occurrence should be selected from the corresponding list and the button  should be clicked)

In each case, if the button  is clicked, the selected item will be removed from the list.

By clicking on the Submit button, the new topic is created and automatically added to the topics list.

Create New Association



When the user clicks on the button, the screen shown in figure below, appears on screen:

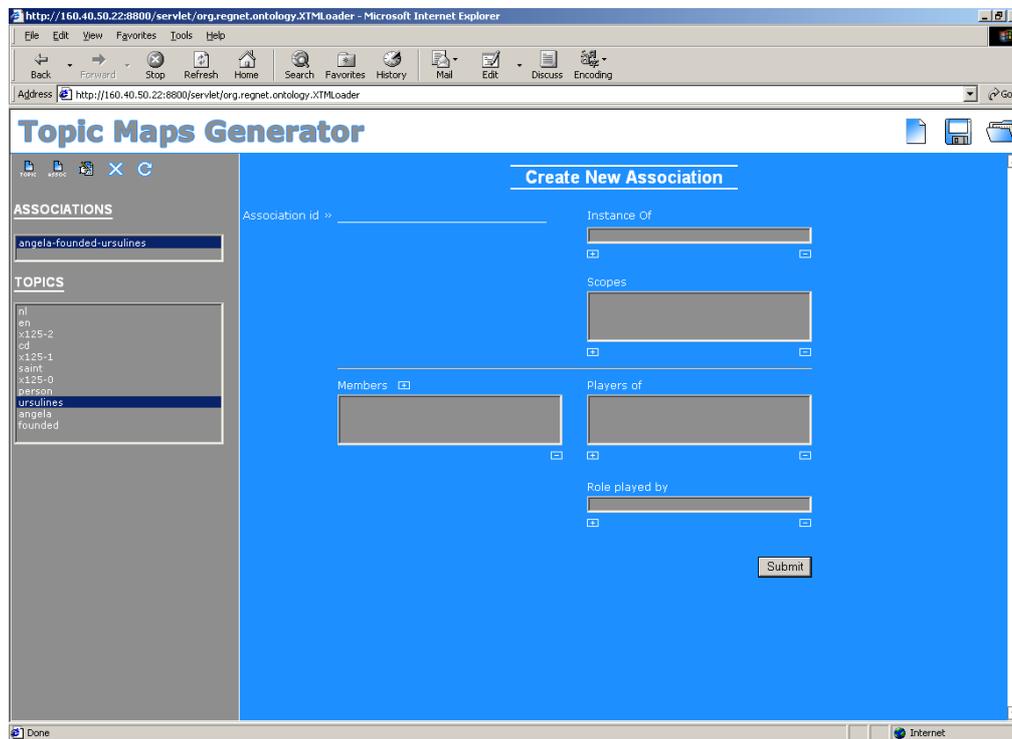


Figure 64: New association screen

Figure 6.

In the working frame (blue), the user can type:

An association id (if the id is not unique, the tool prompts the user)

An instance (in order to add a new instance, a topic should be selected from the topics list and the button  should be clicked)

Multiple scopes (in order to add a new scope, a topic should be selected from the topics list and the button  should be clicked)

Multiple members (in order to add a new member, the button  should be clicked)

Multiple Players of a member (in order to add a new player, a topic should be selected from the topics list, a member should be selected from the members list and the button  should be clicked)

A Role (in order to add a new role, a topic should be selected from the topics list, a member should be selected and the button  should be clicked)



In each case, if the button  is clicked, the selected item will be removed from the list.

By clicking on the Submit button, the new association is created and automatically added to the associations list.

Edit Topic / Association



When the user clicks on the button, he/she is able to edit the selected Topic or Association. After the changes, the tool checks for conflicts (e.g. if the id is changed and the new id already exists, etc) and prompts the user accordingly.

Delete Topic / Association



When the user clicks on the button, he/she is able to delete the selected Topic or Association. If for any reason (e.g. if the association has members) the deletion is not allowed, the tool prompts the user accordingly.

Refresh Topics / Associations List



Although, the Topics/Associations Lists are updated automatically, after the insertion or deletion of a topic or association, the user can refresh the lists by clicking on this button.



13 Annexe C: Internationalisation forms

13.1 eProcurement

13.1.1 Bulgarian

English	Bulgarian
Action	Дейност
All categories	Всички категории
All departments	Всички раздели
All showcases	Всички витрини
and	и
Browse Catalog	Разлистване на каталог
Category	Категория
Category is mandatory.	Категорията е задължителна
Choose a category	Избери категория
Choose a department	Избери раздел
Click here to validate your basket	Натисни тук за потвърждаване на кошницата
Create	Създай
Create product	Създай продукт
Create showcase	Създай витрина
Delete	Изтрий
Delete offer ?	Изтрий предложението
Delete product ?	Изтрий продукта
Department	Раздел
Department is mandatory.	Разделът е задължителен
Description	Описание
From	от
Manage product	Работа с продукта
Manage showcase	Работа с витрината
Mandatory field.	Задължително поле
My Basket	Моята кошница
Name	Име
Name is mandatory.	Името е задължително
No results	Без резултат
Online Shop	Online магазин
Order Total	Обща сума
over	над



Page	Страница
Picture	Картина
Price	Цена
Price is mandatory.	Цената е задължителна
Price is not a well formed number	Цената не е добре формулирана
Price Offer	Ценова листа
Product	Продукт
Product research	Търсене на продукт
Quantity	Количество
Quantity high is mandatory.	Горното количество е задължително
Quantity high is not a well formed number	Горното количество не е добре формулирано
Quantity low is mandatory.	Долното количество е задължително
Quantity low is not a well formed number	Долното количество не е добре формулирано
Reference	Справка
Reference is mandatory.	Справката е задължителна
Results	Резултати
Search	Търсене
Search product for updating	Търсене на продукт и обновяване
Shop	Магазин
Showcase	Витрина
Society	Общество
sub-total	подсума
Supplier	Доставчик
Thank you !	Благодаря
Then to enter price offers of product, use right panel.	За да въведете цената на продукта, използвайте десния панел
Title	Заглавие
To	До
To add a new product, click the 'Create product' button.	За въвеждане на нов продукт натиснете бутона "Създай продукт"
To add a new product, enter first product information and click the 'validate' button.	За въвеждане на нов продукт първо въведете информация за продукта и натиснете бутона "Създай продукт"
To change the quantity of an item, change the number in the box and click the 'update' button.	За изменяне на количеството на продукта променете числото в кутията и натиснете бутона "Обновяване"
To remove an item from your shopping basket, click the 'remove' button.	За изваждане на продукта от пазарската кошница натиснете бутона "Извади"
unit	брой
units	броя
Update	Обновяване



Update product	Обновяване на продукта
Update showcase	Обновяване на витрината
updated	Обновен
Validate	Потвърждение
Watch	Виж
You must create a showcase.	Трябва да създадете витрина
Your command has been successfully sent !	Командата Ви е успешно изпратена !
Your shopping basket contains the following items:	Вашата пазарска кошница съдържа:

13.1.2 Catalan

English	Catalan
Action	Acció
All categories	Totes les categories
All departments	Tots els departaments
All showcases	Tots els aparadors
and	i
Browse Catalog	Consultar el catàleg
Category	categoria
Category is mandatory.	La categoria és obligatòria
Choose a category	Triar una categoria
Choose a department	Triar un departament
Click here to validate your basket	Fer click aquí per validar el vostre cistell
Create	Crear
Create product	Crear un producte
Create showcase	Crear un aparador
Delete	Suprimir
Delete offer ?	Suprimir l'oferta
Delete product ?	Suprimir el producte
Department	Departament
Department is mandatory.	El departament és obligatori
Description	Descripció
From	De
Manage product	Gestionar el producte
Manage showcase	Gestionar l'aparador
Mandatory field.	El camp és obligatori
My Basket	El meu cistell
Name	Nom
Name is mandatory.	El nom és obligatori



No results	No hi ha resultats
Online Shop	Botiga online
Order Total	Total de la comanda
over	Sobre
Page	Pàgina
Picture	Imatge
Price	Preu
Price is mandatory.	El preu és obligatori
Price is not a well formed number	El preu no és correcte
Price Offer	Preu ofert
Product	Producte
Product research	Buscar productes
Quantity	Quantitat
Quantity high is mandatory.	La quantitat superior és obligatòria
Quantity high is not a well formed number	La quantitat superior no és correcta
Quantity low is mandatory.	La quantitat inferior és obligatòria
Quantity low is not a well formed number	La quantitat inferior no és correcta
Reference	Referència
Reference is mandatory.	La referència és obligatòria
Results	Resultats
Search	Buscar
Search product for updating	Buscar productes per actualitzar
Shop	Botiga
Showcase	Aparador
Company	Empresa
sub-total	Sub-total
Supplier	Distribuidor
Thank you !	Gràcies !
The shipping cost is :	Els costos del transport són:
Then to enter price offers of product, use right panel.	Per introduir els preus dels productes, utilitzi el panell de la dreta
Title	Títol
To	A
To add a new product, click the 'Create product' button.	Per afegir un nou producte, fer click al botó "Crear un nou producte"



To add a new product,
enter first product
information and click the
'validate' button. Per afegir un nou producte, introduir primer la informació del producte i fer click al
botó "validar".

To change the quantity of
an item, change the
number in the box and
click the 'update' button. Per canviar la quantitat d'un article, canviï el número de la casella i faci click al
botó "actualitzar"

To remove an item from
your shopping basket, click
the 'remove' button. Per retirar un article del cistell de la compra, faci click al botó "retirar"

unit Unitat

units Unitats

Update Actualitzar

Update product Actualitzar el producte

Update showcase Actualitzar l'aparador

updated Actualitzat

Validate Validar

Watch Veure

Weight Pes

You must create a
showcase. Vostè ha de crear un aparador

Your order has been
successfully sent ! La seva comanda ha estat enviada amb éxit !

Your shopping basket
contains the following
items: El vostre cistell conté els següents articles:

13.1.3 German

English	German
Action	Aktion
All categories	Alle Kategorien
All departments	Alle Abteilungen
All showcases	Alle Schaufenster
and	und
Browse Catalog	Im Katalog blättern
Category	Kategorie
Category is mandatory.	Kategorie ist verpflichtend
Choose a category	Wählen Sie eine Kategorie
Choose a department	Wählen Sie eine Abteilung
Click here to validate your basket	
Create	Anlegen



Create product	Produkt anlegen
Create showcase	Schaufenster anlegen
Delete	Löschen
Delete offer ?	Bestellung löschen?
Delete product ?	Produkt löschen?
Department	Abteilung
Department is mandatory.	Abteilung ist Pflicht
Description	Beschreibung
From	von
Manage product	Produkt verwalten
Manage showcase	Schaufenster verwalten
Mandatory field.	Feld ist verpflichtend, Pflichtfeld
My Basket	Mein Warenkorb
Name	Name
Name is mandatory.	Name ist verpflichtend
No results	Keine Ergebnisse
Online Shop	Online Shop
Order Total	Gesamtbestellung
over	über
Page	Seite
Picture	Abbildung
Price	Preis
Price is mandatory.	Preis ist verpflichtend
Price is not a well formed number	Schreibweise des Preises ändern
Price Offer	Angebot, Sonderangebot
Product	Produkt
Product research	Produktsuche
Quantity	Anzahl
Quantity high is mandatory.	Maximale Anzahl ist verpflichtend
Quantity high is not a well formed number	Schreibweise der maximalen Anzahl ändern
Quantity low is mandatory.	Minimale Anzahl ist verpflichtend
Quantity low is not a well formed number	Schreibweise der minimalen Anzahl ändern
Reference	Referenz
Reference is mandatory.	Referenz ist verpflichtend
Results	Ergebnisse
Search	Suche



Search product for	
updating	Produkt suchen für Aktualisierung
Shop	Shop
Showcase	Schaufenster
Society	Gesellschaft
sub-total	Gesamtsumme
Supplier	Lieferant
Thank you !	Vielen Dank!
Then to enter price offers of product, use right panel.	Benutze den rechten Bildschirm, um Preisangebote von Produkten einzugeben
Title	Titel
To	zu
To add a new product, click the 'Create product' button.	Klicken Sie auf "Produkt erstellen"um ein neues Produkt hinzuzufügen
To add a new product, enter first product information and click the 'validate' button.	Klicken Sie auf "Produkt Information" und dann "Bestätigen", um ein neues Produkt hinzuzufügen
To change the quantity of an item, change the number in the box and click the 'update' button.	Wechseln Sie die Nummer im Kasten und klicken dann auf "Aktualisieren", um die Anzahl der Artikel zu ändern
To remove an item from your shopping basket, click the 'remove' button.	Klicken Sie auf "Entfernen", um einen Artikel aus dem Einkaufswagen zu entfernen
unit	Einheit
units	Einheiten
Update	Aktualisieren
Update product	Produkt aktualisieren
Update showcase	Schaufenster aktualisieren
updated	aktualisiert
Validate	Bestätigen
Watch	Ansicht
You must create a showcase.	Sie müssen ein Schaufenster erstellen
Your command has been successfully sent !	Ihr Befehl wurde erfolgreich versandt
Your shopping basket contains the following items:	Ihr Warenkorb enthält die folgenden Artikel

13.1.4 Dutch

English	Dutch
Action	Actie



All categories	Alle categorieen
All departments	Alle afdelingen
All showcases	Alle vitrines
and	en
Browse Catalog	Catalogus doorzoeken
Category	Categorie
Category is mandatory.	Categorie is verplicht
Choose a category	Kies een categorie
Choose a department	Kies een afdeling
Click here to validate your basket	Klik hier om uw bestelling te bevestigen
Create	Maken
Create product	Product maken
Create showcase	Vitrine maken
Delete	Verwijderen
Delete offer ?	Aanbieding verwijderen?
Delete product ?	Product verwijderen?
Department	Afdeling
Department is mandatory.	Afdeling is verplicht
Description	Beschrijving
From	Van
Manage product	Product bewerken
Manage showcase	Vitrine bewerken
Mandatory field.	Verplicht veld
My Basket	Mijn winkelmand
Name	Naam
Name is mandatory.	Naam is verplicht
No results	Geen resultaten
Online Shop	Online Shop
Order Total	Bestellingsoverzicht
over	over
Page	Pagina
Picture	Afbeelding
Price	Prijs
Price is mandatory.	Prijs is verplicht
Price is not a well formed number	Onjuiste notatie van de prijs
Price Offer	Aanbieding
Product	Product
Product research	Producten zoeken



Quantity	Aantal
Quantity high is mandatory.	Maximaal aantal is verplicht
Quantity high is not a well formed number	Onjuiste notatie van het maximale aantal
Quantity low is mandatory.	Minimaal aantal is verplicht
Quantity low is not a well formed number	Onjuiste notatie van het minimale aantal
Reference	Referentie
Reference is mandatory.	Referentie is verplicht
Results	Resultaten
Search	Zoeken
Search product for updating	Producten zoeken voor bewerking
Shop	Winkel
Showcase	Vitrine
Society	Bedrijf
sub-total	Subtotaal
Supplier	Leverancier
Thank you !	Bedankt!
Then to enter price offers of product, use right panel.	Gebruik het rechter scherm om de prijs van het product toe te voegen
Title	Titel
To	Aan
To add a new product, click the 'Create product' button.	Klik op 'Nieuw product maken' om een nieuw product toe te voegen.
To add a new product, enter first information and click the 'validate' button.	Vul eerst de productinformatie in om een nieuw product toe te voegen en klik vervolgens op 'Bevestigen' knop
To change the quantity of an item, change the number in the box and click the 'update' button.	Verander het getal in het invoervak om het aantal producten te veranderen en klik vervolgens op bevestigen.
To remove an item from your shopping basket, click the 'remove' button.	Klik op 'Verwijderen' om een product uit uw winkelmandje te verwijderen.
unit	eenheid
units	eenheden
Update	Bijwerken
Update product	Product bijwerken
Update showcase	Vitrine bijwerken
updated	Bijgewerkt
Validate	Bevestigen
Watch	Bekijken
You must create a showcase.	U moet een vitrine maken
Your command has been successfully sent !	Uw opdracht is succesvol verstuurt!



Uw winkelmandje bevat de volgende
Your shopping basket contains the following items: artikelen:

13.1.5 English

13.1.6 Spanish

English	Spanish
Action	Acción
All categories	Todas las categorías
All departments	Todos los departamentos
All showcases	Todas las vitrinas
and	y
Browse Catalog(ue?)	Hojear el catálogo
Category	Categoría
Category is mandatory (obligatory?).	La categoría es obligatoria
Choose a category	Elija una categoría
Choose a department	Elija un departamento
Click here to validate your basket	Pulse aquí para verificar su cesta
Create	Crear
Create product	Crear producto
Create showcase	Crear vitrina
Delete	Eliminar
Delete offer ?	¿Eliminar oferta?
Delete product ?	¿Eliminar producto?
Department	Departamento
Department is mandatory.	El departamento es obligatorio
Description	Descripción
From	De
Manage product	Gestionar producto
Manage showcase	Gestionar vitrina
Mandatory field.	El campo es obligatorio
My Basket	Mi cesta
Name	Nombre
Name is mandatory.	El nombre es obligatorio
No results	No hay resultados
Online Shop	Tienda Online
Order Total	Total del pedido
over	sobre
Page	Página



Picture	Imagen
Price	Precio
Price is mandatory.	El precio es obligatorio
Price is not a well formed number	El precio no es una cantidad válida
Price Offer	Precio de oferta
Product	Producto
Product research	Buscar producto
Quantity	Cantidad
Quantity high is mandatory.	La cantidad máxima es obligatoria
Quantity high is not a well formed number	La cantidad máxima no es una cantidad válida
Quantity low is mandatory.	La cantidad mínima es obligatoria
Quantity low is not a well formed number	La cantidad mínima no es una cantidad válida
Reference	Referencia
Reference is mandatory.	La referencia es obligatoria
Results	Resultados
Search	Buscar
Search product for updating	Buscar productos para actualizar
Shop	Tienda
Showcase	Vitrina
Society	Sociedad
sub-total	Subtotal
Supplier	Proveedor
Thank you !	¡Gracias!
Then to enter price offers of product, use the right panel.	Después de introducir las ofertas del producto, use el panel derecho.
Title	Título
To	A
To add a new product, click the 'Create product' button.	Para añadir un nuevo producto, pulse el botón 'Crear producto'
To add a new product, enter first product information and click the 'validate' button.	Para añadir un nuevo producto, introduzca primero la información del producto y pulse el botón 'Verificar'
To change the quantity of an item, change the number in the box and click the 'update' button.	Para cambiar la cantidad de un artículo, cambie el número en el recuadro y pulse el botón 'Actualizar'
To remove an item from your shopping basket, click the 'remove' button.	Para eliminar un artículo de su cesta de la compra, pulse el botón 'Eliminar'
unit	unidad
units	unidades
Update	Actualizar
Update product	Actualizar productos



Update showcase	Actualizar vitrina
updated	actualizado
Validate	Verificar
Watch	Ver
You must create a showcase.	Debe crear una vitrina
Your command has been successfully sent !	¡Su pedido ha sido enviado con éxito!
Your shopping basket contains the following items:	Su cesta de la compra contiene los siguientes artículos:

13.1.7 French

English	French
Action	action
All categories	Toutes les catégories
All departments	Tous les rayons
All showcases	Toutes les vitrines
and	et
Browse Catalog	Parcourir le catalogue
Category	catégorie
Category is mandatory.	La catégorie est obligatoire
Choose a category	Choisir une catégorie
Choose a department	Choisir un rayon
Click here to validate your basket	Cliquez ici pour valider votre panier
Create	Créer
Create product	Créer un produit
Create showcase	Créer une vitrine
Delete	Supprimer
Delete offer ?	Supprimer l'offre ?
Delete product ?	Supprimer le produit ?
Department	rayon
Department is mandatory.	Le rayon est obligatoire
Description	Description
From	De
Manage product	Gérer produits
Manage showcase	Gérer la vitrine
Mandatory field.	Le champ est obligatoire.
My Basket	Mon panier
Name	Nom
Name is mandatory.	Le nom est obligatoire
No results	Pas de résultats



Online Shop	Boutique en ligne
Order Total	Total de la commande
over	sur
Page	Page
Picture	Image
Price	Prix
Price is mandatory.	Le prix est obligatoire
Price is not a well formed number	Le prix n'est pas dans le bon format
Price Offer	Offre de prix
Product	Produit
Product research	Rechercher produits
Quantity	Quantité
Quantity high is mandatory.	La quantité supérieure est obligatoire
Quantity high is not a well formed number	La quantité supérieure n'est pas dans le bon format
Quantity low is mandatory.	La quantité inférieure est obligatoire
Quantity low is not a well formed number	La quantité inférieure n'est pas dans le bon format
Reference	Référence
Reference is mandatory.	La référence est obligatoire
Results	Résultats
Search	Rechercher
Search product for updating	Rechercher produits pour mise à jour
Shop	Boutique
Showcase	Vitrine
Society	Société
sub-total	sous-total
Supplier	Fournisseur
Thank you !	Merci !
Then to enter price offers of product, use right panel.	Puis pour saisir les offres du produits, utilisez le panneau de droite
Title	Titre
To	A
To add a new product, click the 'Create product' button.	Pour ajouter un nouveau produit, cliquez sur le bouton 'Créer produit'
To add a new product, enter first product information and click the 'validate' button.	Pour ajouter un nouveau produit, entrez d'abord les caractéristiques du produit et cliquez sur le bouton 'valider'.
To change the quantity of an item, change the number in the box and click the 'update' button.	Pour changer le quantité d'un article, modifier la quantité affichée et cliquer sur le bouton 'Mettre à jour'



To remove an item from your shopping basket, click the 'remove' button. Pour retirer un article de votre panier, cliquez sur le bouton 'Retirer'

unit

unité

units

unités

Update

Mettre à jour

Update product

Mettre à jour produits

Update showcase

Mettre à jour la vitrine

updated

mis-à-jour

Validate

Valider

Watch

Voir

You must create a showcase.

Vous devez créer une vitrine.

Your command has been successfully sent !

Vous commande a été envoyée avec succès !

Your shopping basket contains the following items:

Votre panier contient les articles suivants :

13.1.8 Greek

English	Greek
Action	Ενέργεια
All categories	Όλες οι κατηγορίες
All departments	Όλα τα τμήματα
All showcases	Όλες οι εκθέσεις
and	Και
Browse Catalog	Εμφάνιση καταλόγου
Category	Κατηγορία
Category is mandatory.	Η κατηγορία είναι υποχρεωτική
Choose a category	Επιλέξτε μια κατηγορία
Choose a department	Επιλέξτε ένα τμήμα
Click here to validate your basket	Πιέστε εδώ για να επικυρώσετε το καλάθι σας
Create	Δημιουργία
Create product	Δημιουργία προϊόντος
Create showcase	Δημιουργία έκθεσης
Delete	Διαγραφή
Delete offer ?	Διαγραφή προσφοράς;
Delete product ?	Διαγραφή προϊόντος;
Department	Τμήμα
Department is mandatory.	Το τμήμα είναι υποχρεωτικό
Description	Περιγραφή
From	Από



Manage product	Διαχείριση προϊόντος
Manage showcase	Διαχείριση έκθεσης
Mandatory field.	Υποχρεωτικό πεδίο
My Basket	Το καλάθι μου
Name	Όνομα
Name is mandatory.	Το όνομα είναι υποχρεωτικό
No results	Κανένα αποτέλεσμα
Online Shop	online-κατάστημα
Order Total	Σύνολο παραγγελίας
over	Πάνω
Page	Σελίδα
Picture	Εικόνα
Price	Τιμή
Price is mandatory.	Η τιμή είναι υποχρεωτική
Price is not a well formed number	Η τιμή δεν είναι ένας έγκυρος αριθμός
Price Offer	Προσφορά τιμής
Product	Προϊόν
Product research	Αναζήτηση προϊόντος
Quantity	Ποσότητα
Quantity high is mandatory.	Η μέγιστη ποσότητα είναι υποχρεωτική
Quantity high is not a well formed number	Η μέγιστη ποσότητα δεν είναι ένας έγκυρος αριθμός
Quantity low is mandatory.	Η ελάχιστη ποσότητα είναι υποχρεωτική
Quantity low is not a well formed number	Η ελάχιστη ποσότητα δεν είναι ένας έγκυρος αριθμός
Reference	Αναφορά
Reference is mandatory.	Η αναφορά είναι υποχρεωτική
Results	Αποτελέσματα
Search	Αναζήτηση
Search product for updating	Αναζήτηση προϊόντος για ενημέρωση
Shop	Κατάστημα
Showcase	Έκθεση
Society	Εταιρεία (if the word is company, otherwise if the word is society, then use this word: Κοινωνία)
sub-total	Υποσύνολο
Supplier	Παροχέας
Thank you !	Ευχαριστώ
Then to enter price offers of product, use right panel.	Για την εισαγωγή προσφοράς τιμής του προϊόντος, χρησιμοποιήστε το δεξί πλαίσιο
Title	Τίτλος
To	Προς
To add a new product, click the 'Create product'	Για την προσθήκη νέου προϊόντος, πείτε το



button.	πλήκτρο δράσης "δημιουργία προϊόντος"
To add a new product, enter first product information and click the 'validate' button.	Για την προσθήκη νέου προϊόντος, εισάγετε πρώτα πληροφορίες για το προϊόν και μετά πιέστε το πλήκτρο δράσης "επιβεβαίωση"
To change the quantity of an item, change the number in the box and click the 'update' button.	Για να αλλάξετε την ποσότητα ενός αντικειμένου, αλλάξτε τον αριθμό στο κουτί και μετά πιέστε το πλήκτρο δράσης "ενημέρωση"
To remove an item from your shopping basket, click the 'remove' button.	Για να απομακρύνετε ένα αντικείμενο από το καλάθι σας, πιέστε το πλήκτρο δράσης "απομάκρυνση"
unit	Μονάδα
units	Μονάδες
Update	Ενημέρωση
Update product	Ενημέρωση προϊόντος
Update showcase	Ενημέρωση έκθεσης
updated	Ενημερωμένο
Validate	Επικύρωση
Watch	Παρακολούθηση
You must create a showcase.	Πρέπει να δημιουργήσετε μια έκθεση
Your command has been successfully sent !	Η εντολή σας έχει σταλεί επιτυχώς!
Your shopping basket contains the following items:	Το καλάθι σας περιέχει τα ακόλουθα αντικείμενα:

13.1.9 Italian

English	Italian
Action	Azione
All categories	Tutte le categorie
All departments	Tutti i dipartimenti
All showcases	Tutte le vetrine
and	E
Browse Catalog	Sfoggia il catalogo
Category	Categoria
Category is mandatory.	La categoria è obbligatoria
Choose a category	Scegliete una categoria
Choose a department	Scegli un dipartimento
Click here to validate your basket	Cliccate qui per confermare il carrello
Create	Create
Create product	Create un prodotto
Create showcase	Annulla
Delete	Annullare l'offerta?
Delete offer ?	Annullare l'offerta?



Delete product ?	Annullare l'offerta?
Department	Dipartimento
Department is mandatory.	Il dipartimento è obbligatorio
Description	Descrizione
From	Da
Manage product	Gestione prodotto
Manage showcase	Gestione vetrina
Mandatory field.	Questo campo è obbligatorio
My Basket	Il mio carrello
Name	Nome
Name is mandatory.	Il nome è obbligatorio
No results	La ricerca ha dato risultato nullo
Online Shop	Negozi virtuale
Order Total	Ordine complessivo
over	Sopra
Page	Pagina
Picture	Immagine
Price	Prezzo
Price is mandatory.	Il prezzo è obbligatorio
Price is not a well formed number	Il prezzo immesso non è ammesso
Price Offer	Prezzo offerta
Product	Prodotto
Product research	Ricerca prodotto
Quantity	Quantità
Quantity high is mandatory.	E' obbligatorio immettere una quantità superiore
Quantity high is not a well formed number	La quantità elevata non è accettabile
Quantity low is mandatory.	E' obbligatorio immettere una quantità bassa
Quantity low is not a well formed number	La quantità bassa non è accettabile
Reference	Riferimento
Reference is mandatory.	Il riferimento è obbligatorio
Results	Risultato
Search	Cerca
Search product for updating	Cerca prodotti aggiornati
Shop	Negozi
Showcase	Vetrina
Company	Società
sub-total	Sub-totale



Supplier	Fornitore
Thank you !	Grazie!
The shipping cost is :	Il costo per il trasporto è di:
Then to enter price offers of product, use right panel.	Poi per immettere il prezzo offerta, usate il pannello a destra
Title	Titolo
To	A
To add a new product, click the 'Create product' button.	Per aggiungere un nuovo prodotto, cliccate sul pulsante 'Crea prodotto'
To add a new product, enter first product information and click the 'validate' button.	Per aggiungere un nuovo prodotto, immettete prima l'informazione sul prodotto quindi cliccate il bottone 'Attiva'
To change the quantity of an item, change the number in the box and click the 'update' button.	Per cambiare la quantità di un articolo, cambiate il numero nella casella e quindi cliccate sul pulsante 'aggiorna'
To remove an item from your shopping basket, click the 'remove' button.	Per rimuovere un'articolo dal vostro carrello, cliccate sul pulsante 'Rimuovi'
unit	unità
units	unità
Update	Aggiorna
Update product	Aggiorna il prodotto
Update showcase	Aggiorna la vetrina
updated	Aggiornato
Validate	Convalidate
Watch	Guardate
Weight	Pesate
You must create a showcase.	dovete generare una vetrina
Your order has been successfully sent !	Il vostro ordine è stato completato con successo!
Your shopping basket contains the following items:	Il vostro carrello della spesa contiene i seguenti articoli:

13.1.10 Russian

English	Russian
Action	Действие
All categories	Все категории
All departments	Все области
All showcases	Все витрины
and	и
Browse Catalog	Пролистать каталог



Category	Категория
Category is mandatory.	Категория обязательна
Choose a category	Выбрать категорию
Choose a department	Выбрать область
Click here to validate your basket	Щелкните мышью здесь, чтобы подтвердить содержимое вашей корзины
Create	Создать
Create product	Создать продукт
Create showcase	Создать витрину
Delete	Удалить
Delete offer ?	Удалить предложение
Delete product ?	Удалить продукт?
Department	Область
Department is mandatory.	Область обязательна
Description	Описание
From	от
Manage product	Работа с продуктом
Manage showcase	Работа с витриной
Mandatory field.	Обязательное поле
My Basket	Моя корзина
Name	Имя
Name is mandatory.	Имя обязательно
No results	Без результатов
Online Shop	Online магазин
Order Total	Общая сумма заказа
over	над / свыше
Page	Страница
Picture	Картина
Price	Цена
Price is mandatory.	Цена обязательна
Price is not a well formed number	Цена сформирована неправильно
Price Offer	Ценовое предложение
Product	Продукт
Product research	Изучение продукта
Quantity	Количество
Quantity high is mandatory.	Верхнее значение количества обязательно
Quantity high is not a well formed number	Верхнее значение количества сформировано неправильно
Quantity low is mandatory.	Нижнее значение количества обязательно



Quantity low is not a well formed number	Нижнее значение количества сформировано неправильно
Reference	Справка
Reference is mandatory.	Справка обязательна
Results	Результаты
Search	Поиск
Search product for updating	Поиск продукта для обновления
Shop	Магазин
Showcase	Витрина
Society	Общество
sub-total	Промежуточная сумма
Supplier	Поставщик
Thank you !	Спасибо!
Then to enter price offers of product, use right panel.	Чтобы ввести ценовые предложения на продукты, используйте правую панель
Title	Заголовок
To	К
To add a new product, click the 'Create product' button.	Чтобы добавить новый продукт, нажмите кнопку "Создать продукт"
To add a new product, enter first product information and click the 'validate' button.	Чтобы добавить новый продукт, сначала введите информацию о нем и нажмите кнопку "Подтвердить"
To change the quantity of an item, change the number in the box and click the 'update' button.	Чтобы изменить количество, смените число в цифровом поле и нажмите кнопку "Обновить".
To remove an item from your shopping basket, click the 'remove' button.	Чтобы удалить товар из корзины, нажмите кнопку "Удалить"
unit	Единица измерения
units	Единицы измерения
Update	Обновить
Update product	Обновить продукт
Update showcase	Обновить витрину
updated	обновленный
Validate	Подтвердить
Watch	Проследить
You must create a showcase.	Вы должны создать витрину
Your command has been successfully sent !	Ваша команда успешно послана!
Your shopping basket contains the following items:	Ваша корзина содержит следующие товары:

13.1.11 Swedish

English	Swedish
Action	Åtgärd



All categories	Alla kategorier
All departments	Alla områden
All showcases	Alla digitala montrar
and	och
Browse Catalog	Bläddra i katalogen
Category	Kategori
Category is mandatory.	Obligatorisk kategori
Choose a category	Välj en kategori
Choose a department	Välj ett område
Click here to validate your basket	Klicka här för att bekräfta innehållet i din korg
Create	Skapa
Create product	Skapa en produkt
Create showcase	Skapa en digital monter
Delete	Radera
Delete offer ?	Radera erbjudande?
Delete product ?	Radera produkt?
Department	Område
Department is mandatory.	Du måste ange område
Description	Beskrivning
From	Från
Manage product	Hantera produkt
Manage showcase	Hantera digital monter
Mandatory field.	Obligatoriskt fält
My Basket	Min korg
Name	Namn
Name is mandatory.	Namnet är obligatoriskt
No results	Inget resultat
Online Shop	Online-butik
Order Total	Totalsumman för ordern
over	över
Page	Sida
Picture	Bild
Price	Pris
Price is mandatory.	Pris är obligatoriskt
Price is not a well formed number	Det angivna priset är inte i en tillåten form
Price Offer	Priserbjudande
Product	Produkt



Product research	Sök efter produkten
Quantity	Kvantitet
Quantity high is mandatory.	Högsta kvantitet är obligatoriskt
Quantity high is not a well formed number	Högsta kvantitet är inte angiven i en tillåten form
Quantity low is mandatory.	Lägsta kvantitet är obligatoriskt
Quantity low is not a well formed number	Lägsta kvantitet är inte angiven i en tillåten form
Reference	Referens
Reference is mandatory.	Referens är obligatoriskt
Results	Resultat
Search	Sök
Search product for updating	Sök efter produkten för uppdatering
Shop	Butik
Showcase	Digital monter
Society	Företag
sub-total	Totalsumma
Supplier	Leverantör
Thank you !	Tack!
Then to enter price offers of product, use right panel.	Använd den högra panelen för att lägga in priserbjudanden
Title	Titel
To	Till
To add a new product, click the 'Create product' button.	Klicka på "Skapa produkt"-knappen för att lägga till en ny produkt
To add a new product, enter first product information and click the 'validate' button.	För att lägga till en ny produkt, skriv först in produktinformationen och klicka sedan på "Bekräfta"-knappen.
To change the quantity of an item, change the number in the box and click the 'update' button.	För att ändra kvantiteten av en artikel, ändra antalet i boxen och klicka på "Uppdatera"-knappen
To remove an item from your shopping basket, click the 'remove' button.	För att ta bort en artikel från din shoppingkorg, klicka på "Ta bort"-knappen
unit	Enhet
units	Enheter
Update	Uppdatera
Update product	Uppdatera produkt
Update showcase	Uppdatera digital monter
updated	Uppdaterad



Validate	Bekräfta
Watch	Se
You must create a showcase.	Du måste skapa en digital monter
Your command has been successfully sent !	Ditt kommando har skickats!
Your shopping basket contains the following items:	Din korg innehåller följande artiklar:
weight	vikt
The shipping cost is	portokostnader



List of Figures

Figure 1: Turbine's extraction of Language information	8
Figure 2: REGNET customisation to intercept Language preferences	9
Figure 3: Topic Map Viewer Architecture.....	21
Figure 4: Topic Map Viewer Interface.....	23
Figure 5: The front end integration template	30
Figure 6: The interface of the portal using the specific tem.....	31
Figure 7: The default page of the e-shop interface using the specific template	32
Figure 8: The interface of the e-shop's basket	33
Figure 9: Architecture of the Topic Map Generator Tool (version 2).....	34
Figure 10: Screenshot of the "Topic Map Generator" tool.....	35
Figure 11: Multi-Portal architecture.....	37
Figure 12: Logical architecture for Portals and Ontology	38
Figure 13: Editing an existent Account: Class Diagram	39
Figure 14: Creation of a new Account: Class Diagram.....	39
Figure 15: Authentication algorithm	40
Figure 16: Relationship between "classic" Electronic Business Collaboration phases and ebXML stack	42
Figure 17: Architecture of an ebXML system	43
Figure 18: The ebXML system.....	44
Figure 19: Description of ebMS specifics	45
Figure 20: ebXML Business Service Interface	47
Figure 21: Architecture of the ebXML Business Process Specification Schema	48
Figure 22: Delivery channels	55
Figure 23: ebXML registry/repository.....	57
Figure 24: ebXML message example: architecture	60
Figure 25: Business transactions scheme	64
Figure 26: Use case diagram.....	65
Figure 27: Electronic Publishing Workflow	79
Figure 28: Structure of the database	82
Figure 29: User use cases	83
Figure 30: Algorithm of the auction.....	84
Figure 31: Architecture of the delivery system	85
Figure 32: Overview Index+ gateway	90
Figure 33: Index+ gateway component	91
Figure 34: Screenshot "Properties of a thesaurus field"	100
Figure 35: Auction system	104
Figure 36: Registration form	105



Figure 37: The view of the current lots	106
Figure 38: The future auctions	107
Figure 39: Searching form	108
Figure 40: Information about user's profile	109
Figure 41: User's bids	110
Figure 42: User's items	111
Figure 43: New item registration	112
Figure 44: Current Lots	113
Figure 45: Administration system	115
Figure 46: Administration system	116
Figure 47: List of auction's items	117
Figure 48: Delete items	118
Figure 49: List of auction's users	119
Figure 50: Delete users	120
Figure 51: Add new category of items in the auction system	121
Figure 52: Exit	122
Figure 53: Screenshot delivery system	123
Figure 54: Screenshot web services Index+	124
Figure 55: Serach request Index+	125
Figure 56: Search request Index+	126
Figure 57: Screenshot Index+ record	127
Figure 58: Retrieved image Image+	135
Figure 59: Screenshot thesaurus management	136
Figure 60: Screenshot thesaurus management - navigation	137
Figure 61: Screenshot thesaurus management - editing	137
Figure 62: Main Screen Topic Map Generator	138
Figure 63: New Topic screen	140
Figure 64: New association screen	141