D2.4.1 Semantic Requirements for Web Service Description

Rubén Lara (Universität Innsbruck)

with contributions from:

Walter Binder (EPFL), Marco Pistore, Paolo Traverso (UniTn), Ion Constantinescu (EPFL), Jeff Pan (VUM), Lyndon Nixon (Fu Berlin), Armin Haller and Tomas Vitvar (NUIG)

Abstract.
EU-IST Network of Excellence (NoE) IST-2004-507482 KWEB
Deliverable D2.4.1 (WP2.4)

In this document, several use case scenarios are analyzed (A2A, B2B, B2C, E-Government) in order to collect requirements regarding semantic enabled process integration. In particular, requirements on domain ontologies, web services and mediators are identified providing essential source for developing and improving conceptual and formal framework for semantic web services including web service discovery and composition.
Keyword list: Semantic Web Services; Mediator, Ontology, Web services requirements; A2A, B2B, B2C, E-Government Use Cases

Document Identifier	KWEB/2004/D2.4.1/v1.0
Project	KWEB EU-IST-2004-507482
Version	v1.0
Date	January 10, 2005
State	final
Distribution	public

Knowledge Web Consortium

This document is part of a research project funded by the IST Programme of the Commission of the European Communities as project number IST-2004-507482.

University of Innsbruck (UIBK) - Coordinator
Institute of Computer Science
Technikerstrasse 13
A-6020 Innsbruck
Austria
Contact person: Dieter Fensel
E-mail address: dieter.fensel@uibk.ac.at

École Polytechnique Fédérale de Lausanne (EPFL)
Computer Science Department
Swiss Federal Institute of Technology
IN (Ecublens), CH-1015 Lausanne
Switzerland
Contact person: Boi Faltings
E-mail address: boi.faltings@epfl.ch

France Telecom (FT)
4 Rue du Clos Courtel
35512 Cesson Sévigné
France. PO Box 91226
Contact person : Alain Leger
E-mail address: alain.leger@rd.francetelecom.com

Freie Universität Berlin (FU Berlin)
Takustrasse 9
14195 Berlin
Germany
Contact person: Robert Tolksdorf
E-mail address: tolk@inf.fu-berlin.de

Free University of Bozen-Bolzano (FUB)
Piazza Domenicani 3
39100 Bolzano
Italy
Contact person: Enrico Franconi
E-mail address: franconi@inf.unibz.it

Institut National de Recherche en Informatique et en Automatique (INRIA)
ZIRST - 655 avenue de l'Europe -
Montbonnot Saint Martin
38334 Saint-Ismier
France
Contact person: Jérôme Euzenat
E-mail address: Jerome.Euzenat@inrialpes.fr

Centre for Research and Technology Hellas / Informatics and Telematics Institute (ITI-CERTH)
1st km Thermi - Panorama road
57001 Thermi-Thessaloniki
Greece. Po Box 361
Contact person: Michael G. Strintzis
E-mail address: strintzi@iti.gr

Learning Lab Lower Saxony (L3S)
Expo Plaza 1
30539 Hannover
Germany
Contact person: Wolfgang Nejdl
E-mail address: nejdl@learninglab.de

National University of Ireland Galway (NUIG)
National University of Ireland
Science and Technology Building
University Road
Galway
Ireland
Contact person: Christoph Bussler
E-mail address: chris.bussler@deri.ie

The Open University (OU)
Knowledge Media Institute
The Open University
Milton Keynes, MK7 6AA
United Kingdom
Contact person: Enrico Motta
E-mail address: e.motta@open.ac.uk

Universidad Politécnica de Madrid (UPM)
Campus de Montegancedo sn
28660 Boadilla del Monte
Spain
Contact person: Asunción Gómez Pérez
E-mail address: asun@fi.upm.es

University of Karlsruhe (UKARL)
Institut für Angewandte Informatik und Formale
Beschreibungsverfahren - AIFB
Universität Karlsruhe
D-76128 Karlsruhe
Germany
Contact person: Rudi Studer
E-mail address: studer@aifb.uni-karlsruhe.de

University of Liverpool (UniLiv)
Chadwick Building, Peach Street
L697ZF Liverpool
United Kingdom
Contact person: Michael Wooldridge
E-mail address: M.J.Wooldridge@csc.liv.ac.uk

University of Manchester (UoM)
Room 2.32. Kilburn Building, Department of Computer
Science, University of Manchester, Oxford Road
Manchester, M13 9PL
United Kingdom
Contact person: Carole Goble
E-mail address: carole@cs.man.ac.uk

University of Sheffield (USFD)
Regent Court, 211 Portobello street
S14DP Sheffield
United Kingdom
Contact person: Hamish Cunningham
E-mail address: hamish@dcs.shef.ac.uk

University of Trento (UniTn)
Via Sommarive 14
38050 Trento
Italy
Contact person: Fausto Giunchiglia
E-mail address: fausto@dit.unitn.it

Vrije Universiteit Amsterdam (VUA)
De Boelelaan 1081a
1081HV. Amsterdam
The Netherlands
Contact person: Frank van Harmelen
E-mail address: Frank.van.Harmelen@cs.vu.nl

Vrije Universiteit Brussel (VUB)
Pleinlaan 2, Building G10
1050 Brussels
Belgium
Contact person: Robert Meersman
E-mail address: robert.meersman@vub.ac.be

Work package participants

The following partners have taken an active part in the work leading to the elaboration of this document, even if they might not have directly contributed to writing parts of this document:

École Polytechnique Fédérale de Lausanne
Freie Universität Berlin
National University of Ireland Galway
Universität of Innsbruck
University of Manchester
University of Trento

Chapter 1

Changes

Version	Date	Author	Changes
0.1	04.10.04	Rubén Lara	creation
0.2	09.11.04	Armin Haller	e-Government chapter added
0.3	21.12.04	Tomas Vitvar	executive summary, introduction and conclusion sections added
1.0	10.01.05	Tomas Vitvar	Final version

Chapter 2

Executive Summary

Almost everybody today acknowledges that the internet as a global network will lead us to an open economy where information like pricing strategies, delivery performances, or suppliers' and customers' satisfaction scores will be available at a mouse-click distance for anybody and anywhere. With respect to this open economy, several trends emerged, such as approaches to integration of business partners (B2B), growing number of e-business services such as e-Commerce or B2C, or on-line competing suppliers at e-Marketplaces.

Traditional approaches to business process integration based on EDI, BPEL, RosettaNet or web services are limited, as communication scenarios in heterogeneous systems are hard coded in systems' processes. In this traditional communication, role of humans is essential when applying new requirements for communication. Thus, ability to adapt to changes in systems behavior is limited as well as ability to discover new opportunities in exchanging of information is bounded to humans. To overcome obstacles of traditional business processes integration, semantic-enabled communication is required to be part of business processes integration scenarios. We understand this semantics to be an explicit description of services, its capabilities, and description of concepts (ontologies) as well as user goals towards these services.

The goal of this document is to analyze selected business scenarios to collect requirements concerning semantic description of services. To do that, we have addressed several use case scenarios to cover requirements for integration of a business within its boundaries as a legal entity (A2A), integration of a business with its customers (B2C), integration of a business with its business partners (B2B) and integration of government authority with its citizens (e-government). As a conclusion, requirements on domain ontologies, web services and mediators were identified so that described integration scenarios could be provided automatically and efficiently. Collected requirements from this work is the essential source for developing and improving Web Services Modelling Ontology (WSMO)[RLK04] and Web Services Modelling Language (WSML)[dBLK$^+$04] within conceptual and formal framework for semantic web services as well as for web service discovery and composition.

Contents

Chapter 3

Introduction

Almost everybody today acknowledges that the internet as a global network will lead us to an open economy where information like pricing strategies, delivery performances, or suppliers' and customers' satisfaction scores will be available at a mouse-click distance for anybody and anywhere. With respect to this open economy, several trends emerged, such as approaches to integration of business partners (B2B), growing number of e-business services such as e-Commerce or B2C, or on-line competing suppliers at e-Marketplaces.

Traditional approaches to business process integration based on EDI, BPEL, RosettaNet or web services are limited, as communication scenarios in heterogeneous systems are hard coded in systems' processes. In this traditional communication, role of humans is essential when applying new requirements for communication. Thus, ability to adapt to changes in systems behavior is limited as well as ability to discover new opportunities in exchanging of information is bounded to humans. To overcome obstacles of traditional business processes integration, semantic-enabled communication is required to be part of business processes integration scenarios. We understand this semantics to be an explicit description of services, its capabilities, and description of concepts (ontologies) as well as user goals towards these services.

3.1 Approach for Extracting Requirements

In order to gather as much as possible requirements regarding semantic enabled integration of processes, we have addressed several use case scenarios in this document. These use case scenarios were selected to cover requirements for integration of a business within its boundaries as a legal entity (A2A), integration of a business with its customers (B2C), integration of a business with its business partners (B2B) and integration of government authority with its citizens (e-government). Each use case scenario in this document describes actors involved in the use case, their roles as well as goals, and also descrip-

tion of business processes including new requirements for semantic enabled services and processes.

3.2 Goal and overview of the document

The goal of this document is to analyze selected business scenarios to collect requirements concerning semantic description of services. In section 4 we describe scenario of application to application integration within legal boundaies of a business on an example of shared prices, in section 5 we analyze business to customer ordering processes on an example of virtual travel agency, in section 6 we describe scenario of a supply chain when business integration with its business partners is required and finally in section 7 we analyze scenario involving e-government services on an example of waste disposal license approval by responsible authorities. In section 8 requirements on semantic description of services are aggregated and summarized.

Chapter 4

A2A Use Case

4.1 Retrieve Share Prices

In this section, we present an Application-to-Application (A2A) use case in the context of a financial institution, whereas A2A describes the integration of different types of software. It represents a buzzword widely used in the past years; but its interpretations are different among people. We are using the definition of Bussler, who calls Application-to-Application a subset of B2B integration. Hence it is the foundation for successful interactions between two companies in a B2B scenario [Bussler, 2003]. Furthermore he defines A2A as an integration which refers to the integration of any number of back-end application systems that are hosted and require integration. This includes the integration of back-end application systems over a network that resides in a remote division of the enterprise. Therefore A2A is every integration of back-end applications where the boundaries of the company as a legal entity are not crossed. In the following, we introduce the use case and give its details in order to extract the requirements posed on Semantic Web Service descriptions and their exploitation.

4.1.1 Introduction of the Use Case

For our use case we will exemplify the retrieval of a share price and the purchase of a share within a bank operating in several countries with different divisions dealing with investment funds. One of them, a division managing a global hedge fund, wants to retrieve information about a specific share. Typically purchases of shares on different markets within a bank are managed by so called Settlement divisions. In large-scale banks there might be not only one Settlement, but several ones in different countries with overlapping functionality. Similarly different funds are managed in different divisions (e.g. pension funds in different departments than hedge funds), either by outsourced sub companies or independently entities within the bank.

As the term A2A implies, two applications have to communicate with each other. Heterogeneous environments rule out the use of a uniform platform or language-specific object model or programming API. Instead, message exchange must occur in a platform and language-independent manner, allowing all players to participate without having to rewrite their existing business applications. In our use case this API is provided by Web Service. The remaining issue on the communication layer [DRRB03] to overcome protocol mismatches by translating and converting messages between heterogeneous protocols in our use case is dealt within an integrator broker. Furthermore this middleware component is compliant to a new paradigm in integration architecture design, the service-oriented architecture (SOA). A SOA is essentially a collection of services communicating with each other. The communication can involve either simple data passing or it could involve two or more services coordinating some activity. To be a service-oriented architecture it has to play the following three roles [Bur00]:

- Allow service providers to publish the availability of their services.
- Allow service brokers to register published services.
- Allow service requesters to use broker services to find a needed service.

The step beyond current approaches in our use case is the dynamic behaviour of the integration tool. Published services can be located dynamically with no need for prior agreements. Business Processes can be dynamically composed based on the request received and the available services published to the registry.

In a nutshell the integrator broker will provide the following functionality. As mentioned above, an application, acting as an agent for a division managing a global hedge fund, further called HFD (hedge fund division), requests the spot price of a specific share. The functionality for ordering is provided by an application acting as an agent for a settlement division within the bank. To discover the service it has to be published by the settlement division beforehand and registered to the integrator broker. Based on the Semantic description the broker is able to discover the Web Service based on the requested functionality by the HFD agent application. The benefit for the requesting entity is the dynamic behaviour of the integrator broker regarding the actual invocation of the Web Service. Hence the requesting agent application is not bound to invoke a certain Web Service for a specific operation, but the invocation is handled by the integrator broker based on the requesters goal expressed in a logical expression. The following sections describe in detail what requirements on this semantic descriptions are imposed to fully support the dynamic discovery and invocation.

4.1.2 Actors, Roles and Goals

In the general use case there are three actors. The following defines why they participate in this use case (goal) and the particular interactions they are involved in (roles).

- requesting application: proprietary portfolio management application installed in the HFD, which interacts with the integrator broker to consume a service published by a provider.
  - Goal: automated resolution of its request by the integrator broker in a transparent fashion
  - Role: interacts with integrator broker for service usage
- providing application: proprietary application connecting to different market places to retrieve real-time information and settle specific trades.
  - Goal: providing high quality settlement to act on behalf of other divisions within the bank
  - Role: provide their services via Web Services and publish it to the integrator broker with the necessary semantic descriptions of such Web Services
- integrator broker: the intermediary application, overcoming heterogeneity issues when connecting different applications. It allows service providers to register their Web Services and provides discovery functionality for service requestors
  - Goal: aggregates functionality available within the bank by overcoming heterogeneity problems on the communication layer and providing the means to publish services
  - Role: interacting with the requesting application (offers its functionality as web service), locating, composing (if necessary) and using the services offered by the providing application and, ultimately, solving the request

4.1.3 Usage Scenarios

1. Proprietary portfolio management application requests a specific share price.

Goal/Context: The proprietary application managing the portfolio of the HFD was configured in a way that it automatically has to check the spot price of different shares in regular intervals to be able to derive technical analysis. Since the application is configured to avoid tight coupling with specific applications, it sends the goal to get the price of Bridgestone Corporation (symbol BRDCY.PK) at the Tokio Stock Exchange (TSE) to the integrator broker.

The integrator broker has to locate Web Services offering relevant information to the requester within the bank.

Participating actors: requesting application, integrator broker, providing applications.

Scenario/Steps:

1. The portfolio management application constructs a request based on a template with its desired stock symbol and associated settlement constraints.
2. The applications sends the request to the integrator broker.
3. The integrator broker receives the request and interprets its semantics.
4. Based on the previous request, service providers in settlement divisions that can provide information related to the customer request are located.
5. Based on the settlement constraints and user preferences the integrator broker selects one service provider.
6. Based on the previously interpreted message, the integrator broker constructs a request for information from a service provider.
7. the integrator broker submits the request to the service provider's application.
8. the application receives the request and processes it.
9. the invoked application at the service provider side returns the relevant information to the integrator broker.
10. the integrator broker receives the requested information from the service provider.
11. The integrator broker filters out irrelevant information and aggregates the relevant information for the requesting application.
12. The aggregated information is sent back to the requesting application.
13. The portfolio management application receives the answer to its request and processes it.

Possible Extensions:

1. The integrator broker does not locate an atomic Web Service, but a Web Service with a more complex interaction; e.g. the Web Service of the integrator broker first requires login information to get real time market prices.
2. The requesting application and the service provider use different terms to express their requests and results; more precisely the ontologies differ.
3. The requesting application dynamically discovers one or more integrator brokers within the bank to process its request.

Requirements: In the following, we describe the requirements on different aspects of Web Service descriptions extracted from the usage scenario above:

- Domain Ontologies.

  1. *From Steps 1, 13:* The appropriate vocabulary for expressing the service request and the provided service must be available i.e. domain ontologies providing the terminology for the financial domain must be in place.

- Requests.

  1. *From Step 1:* The requesting application needs to have some goal template configured to express its request. It has to be some formalised way understandable by the integrator broker.
  2. *From Step 1:* It has to be possible to express constraints and preferences in requests.
  3. *From Step 7:* The integrator broker needs to submit the requests to the located service providing applications. Therefore, the integrator broker needs to know how to formalize these requests to be understood by such services, including the transformation from the requesting application's message into the appropriate request for the service providers.
  4. *From Step 5:* The requesting application has to be able to include non-functional properties in its request, and the integrator broker needs to do the same for the requests sent to the service providers.

- Web Services.

  1. *From Steps 2, 13:* The integrator broker has to describe its Web Service interface to ensure that the requesting application can submit its requests and know how the results will be returned.
  2. *From Steps 2, 13:* The requesting application has to describe its Web Service interface to ensure that the integrator broker knows how the results will be returned.
  3. *From Step 4:* The services offered by different settlement divisions accessible via Web Services have to be formally described. That means the Web Services have to include a formal description of their functionality.
  4. *From Step 4:* The Web Services offered by the settlement divisions have to be published to the integrator broker in order to be found in the matchmaking process.
  5. *From Step 4:* A mechanism to match requests from the HFD agent application to Web Services offered by different settlement divisions is required.

6. *From Step 8, 9:* Service providers have to describe how a request is to be submitted by the integrator broker and how the results will be returned i.e. the Web Service descriptions have to include invocation details.
7. *From Step 7:* The choreography of the service provider has to be formally described in order to enable interoperation with the integrator broker.
8. *From Step 2, 12:* The choreography of the service requester has to be formally described as well in order to enable interoperation with the integrator broker.
9. *From Step 5:* The Web Services of the service providers have to describe their non-functional characteristics.
10. *From Extension 3:* The choreography of the interface offered by the integrator broker has to be formally described so that the requester can interact with it.
11. *From Extension 3:* The Web Service of the integrator broker has to include a formal description of its functionality in order to be located.

- Mediation support.

1. *From Step 2, 7, 12:* Mediation is required at the choreography level in order to enable the interoperation between the service requesting applications and the service providers.
2. *From Extension 2:* Mediation at the terminology level i.e. ontology mediation is required to enable understanding between the heterogeneous actors.

- Others.

1. *From Step 4:* Tool support must be in place to publish formal descriptions of Web Services by service providers.

Chapter 5

B2C Use Cases

5.1 Virtual Travel Agency

In this section, we present a B2C use case in the context of e-Tourism services. The use case is based on the W3C Travel Agent Use case defined in [HHO04]. In the following, we introduce the use case and give its details in order to extract the requirements posed on Semantic Web Service descriptions and their exploitation.

5.1.1 Introduction of the Use Case

Customers often require different tourism services packaged to cover all their needs for a trip. Travel agencies offer their services to arrange a trip for their customers, which implies contacting different providers e.g. hotel providers, airlines, or car rental companies, getting information from them, booking the most appropriate option for the customer, and paying the different providers on behalf of the customer. The customer gets the final result of this process i.e. a trip in a transparent fashion; the travel agency takes care of the details.

Our use case is situated in this context, but presents some features that go beyond traditional travel agencies. We will call such an enhanced travel agency VTA, which stands for Virtual Travel Agency. This agency is an e-Tourism service provider which offers a wide range of services by using and interacting with other e-Tourism service providers. The functionality of the VTA is that of a traditional travel agency: getting a request from a customer, dealing with different e-Tourism providers to put together an appropriate offer covering the customer request, arranging all the booking and payment with the different providers, and transparently offering the final trip to the customer. The step beyond traditional travel agencies and existing electronic travel agencies is the dynamic behaviour of the VTA: the available e-Tourism providers can be located dynamically with no need for prior agreements, the business process of the VTA can be composed dynamically based

on the request received and the available providers, and it presents a number of additional dynamic features that will be discussed in more detail in subsequent sections. By applying Semantic Web Services, the VTA will invoke Web Services provided by several e-Tourism suppliers and aggregate them into new customer services in a (semi)automatic fashion.

Our VTA, in a nutshell, shall provide the following functionality: A customer uses the VTA service as the entry point for his requests. These end-user services are provided by the VTA by invoking and combining Web Services offered by several tourism service providers. Figure 5.1 gives an overview (modified and extended from W3C Travel Agent Use Case overview, as defined in [HHO04]) of the use case.

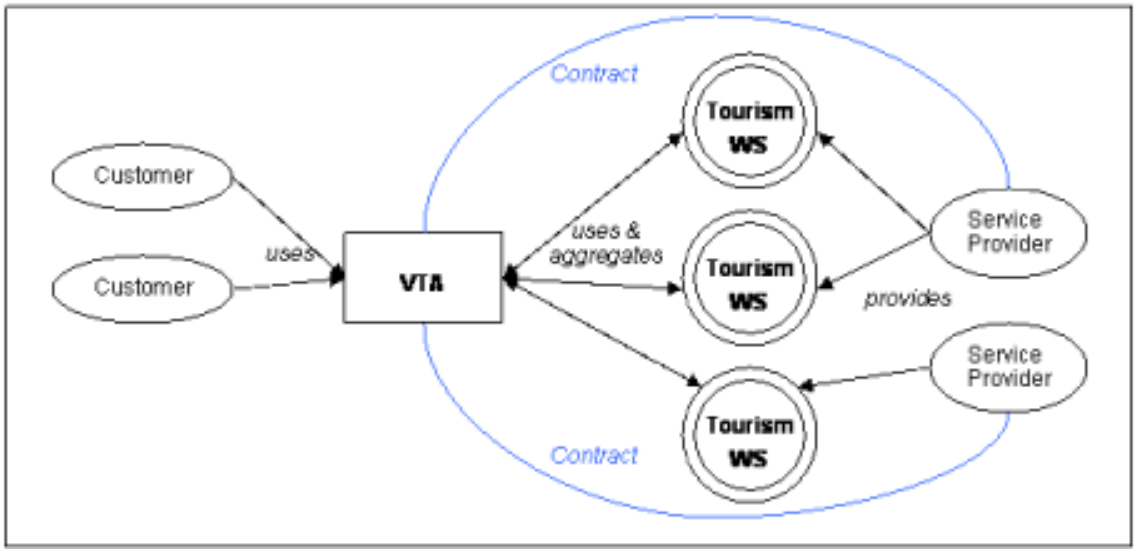

Figure 5.1: VTA overview

The figure outlines a general structure for VTAs that can be extended to more complex scenarios wherein the customer can be a Web Service itself, thus creating a network of composed services that offer complex tourism services. For example, one VTA can provide flight booking services for an airline union, another VTA aggregates booking service for a worldwide hotel chain, and a third VTA provides booking services for rental cars by combining the services of several car rental agencies. Then, another VTA uses these services for providing an end-user service for booking complete holiday trips worldwide.

5.1.2 Actors, Roles and Goals

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

- Customer: the end-user that requests a service provided by the VTA.
  - Goal: automated resolution of his request by the VTA in a transparent fashion.
  - Role: end-user, interacts with VTA for service usage, payment, and non-computational assets (e.g. receiving the actual ticket when booking a trip).
- Tourism service providers: commercial companies that provide specific tourism services.
  - Goal: sell their tourism services to customers.
  - Role: provide tourism services via Web Services, and they also provide the necessary semantic descriptions of such Web Services.
- VTA: the intermediary between the Customer and the tourism service providers. It provides high-quality tourism services to customers by aggregating the separate services of different tourism service providers.
  - Goal: provides high-quality end-user tourism services, uses existing tourism services and aggregates these into new services. It represents the union of available tourism service providers.
  - Role: interacting with the customer, locating, composing (if necessary) and using the services offered by the tourism providers and, ultimately, solving the customer request.

5.1.3 Usage Scenarios

1. Customer requests VTA for searching flight offers.

Goal/Context: The customer wants to fly from Innsbruck to Madrid on 15.01.2005, in business class, and arriving in Madrid no later than 18:00. A direct flight is preferred. He requests the VTA to find an offer for that flight, but not booking yet.

The customer will provide the VTA with the itinerary for the flight, the class, the date, and the time constraints. With this information, the VTA has to use flight information services to find out what options are available for the requested flight, select the ones that fit the customer request, and return them to the customer.

For this, the VTA has to locate web services from tourism service providers that can provide relevant information for the customer request i.e. relevant information about flights from Innsbruck to Madrid.

Participating actors: Customer, VTA, tourism service providers.

Scenario/Steps:

1. The user constructs a request with his desired itinerary and associated constraints.
2. The user submits the request to the VTA.
3. The VTA receives the request and ”understands” it.
4. Based on the previous understanding, the VTA constructs a request for information from the tourism service providers.
5. Based on the previous request, tourism service providers that can provide information related to the customer request are located.
6. For each tourism service provider found:
   (a) The VTA submits its request to the tourism service provider.
   (b) The tourism service provider receives the request and processes it.
   (c) The tourism service provider returns the relevant information to the VTA.
   (d) The VTA receives the requested flight information from the tourism service provider.
7. The VTA, after collecting all the flight information from the tourism service providers, filters out irrelevant information and aggregates the relevant information for the customer.
8. The VTA delivers the aggregated relevant information to the requester.
9. The customer receives the answer to his request.

Possible Extensions:

1. The customer of the VTA is not a human user but a Web Service.
2. The customer wants to include non-functional properties in his request e.g. country of the provider, kind of encryption used, privacy policy, etc.
3. The VTA does not provide an atomic Web Service but a Web Service with a more complex interaction e.g. the Web Service of the VTA first requires login information, if it is valid it accepts the request, if some information is missing it requests from the customer the missing information, etc.
4. The customer and the VTA have different choreographies. We understand choreography in the sense of the external visible behaviour of the interacting Web Service [RLK04].

5. The customer, VTA and tourism service providers use different communication protocols.

6. There is no tourism service provider fulfilling the VTA (customer) request, but the combination of several tourism service providers can fulfill it.

7. Some of the located tourism service providers do not provide an atomic Web Service but a Web Service with a more complex interaction.

8. The VTA and the tourism service providers have different choreographies.

9. The customer, the VTA, and the tourism service providers use different tourism terminologies to express their requests and results.

10. The customer, the VTA, and the tourism service providers use ontologies described in different languages.

11. The customer locates the VTA dynamically.

12. The VTA uses some pre-defined providers for some flight itineraries with which it has a prior agreement, while it dynamically selects the information provider for any other itinerary.

Requirements: In the following, we describe the requirements on different aspects of Web Service descriptions extracted from the usage scenario above:

- Domain Ontologies.

    1. *From Steps 1, 9:* The appropriate vocabulary for expressing the customer request and understanding the results found by the VTA must be available i.e. domain ontologies providing the terminology for the tourism domain must be in place.
    2. *From Steps 3, 4, 7:* The necessary tourism domain ontologies providing the terminology to understand the user request, construct the request for the tourism service providers, and understand their results, must be available to the VTA.
    3. *From Step 6(b):* Appropriate tourism domain ontologies must be available to the tourism service providers in order to understand the VTA requests.
    4. *From Extension 2:* Domain ontologies providing the terminology for non-functional properties such as security, privacy, etc. must be available to the customer, VTA and tourism service providers.

- Requests.

1. *From Step 1 and Extension 1:* The customer (himself or an agent acting on his behalf) needs to know how he has to express his request to be understood by the VTA i.e. how to formalize his request.
2. *From Steps 1:* It has to be possible to express constraints and preferences in requests.
3. *From Step 4:* The VTA needs to know how to express its request to enable the location of appropriate tourism service providers.
4. *From Step 6(a):* The VTA in turn needs to submit his requests to the located tourism service providers. Therefore, the VTA needs to know how to formalize these requests to be understood by such services, including the transformation from the customer request into the appropriate request for the tourism service providers.
5. *From Step 9:* A way to check that the returned results fulfill the user request is needed.
6. *From Extension 2:* The customer has to be able to include non-functional properties in his request, and the VTA needs to do the same for the requests sent to the tourism service providers.
7. *From Steps 1, 3, 4, 6(a), 6(b), and Extension 2:* Requests need to be expressed in terms of the afore-mentioned domain ontologies in order to enable understanding between the customer and the VTA, and between the VTA and the tourism service providers.
8. *From Extension 11:* The customer needs to know how to express its request to enable the location of an appropriate VTA for its request.

- Web Services.

1. *From Steps 2, 9:* The Web Service has to describe how the request is to be submitted by the customer and how the results will be returned i.e. the VTA Web Service description has to include invocation details.
2. *From Step 5:* The services of the tourism service providers accessible via Web Services have to be formally described. That means that the Web Services of the tourism service providers have to include a formal description of their functionality in order to be located.
3. *From Step 5:* A mechanism to match requests from the VTA to Web Services offered by tourism service providers is required. That implies that is has to be possible to find a relation between customer requests and the functionality description of the VTA.
4. *From Step 6(a), 6(c):* The Web Services of the tourism service providers have to describe how the request is to be submitted by the VTA and how the results will be returned i.e. the Web Service descriptions have to include invocation details.

5. *From Step 7:* The data flow from the VTA to the different tourism service providers and back has to be described. In addition, how to manipulate data resulting from the invocation i.e. how different offers will be aggregated needs to be described.
6. *From Extension 1:* The choreography of the customer has to be formally described in order to enable interoperation with the VTA.
7. *From Extension 2:* The Web Services of the VTA and tourism service providers have to describe their non-functional characteristics.
8. *From Extension 3:* The choreography of the Web Service offered by the VTA for customer consumption has to be formally described so that the customer can interact with it.
9. *From Extension 5:* All the Web Services used need to describe their communication protocols.
10. *From Extension 6:* The description of the service offered by different tourism service providers must be amenable to composition.
11. *From Extension 7:* The choreographies of tourism service providers have to be formally described in order to enable interoperation with the VTA.
12. *From Extension 11:* The Web Service of the VTA has to include a formal description of its functionality in order to be located.
13. *From Extension 12:* The orchestration of the VTA Web Services must allow the use of both statically defined i.e. hardwired Web Services and of dynamically located Web Services. That means that some activities of the business process described by the orchestration can be expressed as a request and resolved at run-time by locating appropriate tourism service providers, while others are resolved using pre-defined Web Services.

- Mediation support.

1. *From Extensions 4, 8:* Mediation is required at the choreography level in order to enable the interoperation between the customer, the VTA, and the tourism service providers.
2. *From Extension 5:* Mediation is required at the protocol level to resolve protocol heterogeneities between the customer, VTA and tourism service providers.
3. *From Extension 9:* Mediation at the terminology level i.e. ontology mediation is required to enable understanding between the heterogeneous actors.
4. *From Extension 10:* Ontology mediation has to be provided to translate between different languages.

- Others.

1. *From Steps 2, 3:* Tool support must be in place to define and submit customer requests e.g. Web-based forms.
2. *From Extension 2:* It must be possible to monitor that the constraints imposed by the customer e.g. privacy policy are enforced during invocation.
3. *From Extensions 6, 12:* The description of the orchestration of the VTA must have underlying formal semantics in order to enable e.g. formal verification.

2. Customer requests VTA for booking one of the located flights.

Goal/Context: The customer wants to fly from Innsbruck to Madrid on 15.01.2005, in business class, and arriving in Madrid no later than 18:00. A direct flight is preferred. He has requested (see the usage scenario above) the VTA to find an offer for that flight, and he has received one or more offers that can (totally or partially) fulfill his needs. Now, he wants to select one of the offers and actually book the flight given in such offer.

The customer will analyze the offers available in order to know which ones completely fulfill his needs. If none of them does it, he will either select the one most closely fitting his expectations or he will change his request and repeat the search process described in the previous usage scenario.

Eventually, he will select one of the offers or he will not go ahead with the booking process in case he did not get any acceptable offer. In the former case, the customer will request the VTA to book the flight corresponding to the selected offer.

The VTA will ask the (one or more) tourism service providers which generated the offer to book the flight corresponding to part of the offer (in case several providers have to book different parts of a composed itinerary) or to the complete offer (in case one single provider offered a direct flight).

Finally, when the tourism service providers complete the booking, the VTA will receive the corresponding confirmations, will integrate them, and return them to the customer.

Participating actors: Customer, VTA, tourism service providers.

Scenario/Steps:

1. The user selects the flight he wants to book and the detailed information of the payment methods he wants to use e.g. credit card type, number, holder and expiration date.
2. The user submits the request to the VTA.
3. The VTA receives the request and ”understands” it.

4. Based on the previous understanding, the VTA constructs a request for booking, with the payment details given, for the tourism service providers that gave the offer selected by the customer.

5. For each of the tourism service providers involved in the offer:

   (a) The VTA submits its request to the tourism service provider.

   (b) The tourism service provider receives the request and processes it.

   (c) The tourism service provider performs the booking, charges the customer, and returns a confirmation to the VTA.

   (d) The VTA receives the confirmation of the booking from the tourism service provider.

6. The VTA, after collecting all the confirmations from the tourism service providers, aggregates them.

7. The VTA delivers the aggregated confirmations to the requester.

8. The customer receives the confirmation for the booking of his flight.

Possible Extensions:

1. The customer wants to include new conditions in his request e.g. that he wants to book the selected offer only if cancellation is possible and a maximum of 10% cancellation fee is applied.

2. The customer wants to include new non-functional properties for his request, involving security, privacy, etc.

3. Some of the tourism service providers that provided the offer are now off-line.

4. For a composite booking e.g. involving different tourism service providers, transactionality is required.

5. Booking and payment are separate steps.

6. The payment information given by the requester is not accepted by some of the tourism service providers.

7. The confirmations of the different tourism service providers use different terminologies.

3. Customer requests VTA for cancelling the booked flight.

Goal/Context: The customer booked a flight from Innsbruck to Madrid on 15.01.2005. However, he wants to now cancel the booking.

The customer will send a request to the VTA for cancelling the flight he booked. The VTA will have to know, based on the booking confirmations of the flight booked by the customer, what tourism service provider actually booked the flight (or flights, if it was not a direct connection).

The VTA will contact the relevant tourism service providers to cancel the booked flights. The tourism service providers will return a confirmation of the cancelling including the cancellation fees applied. The VTA will receive this information, integrate it, and return it to the customer.

Participating actors: Customer, VTA, tourism service providers.

Scenario/Steps:

1. The user constructs a cancellation request for a flight previously booked, including the booking information.
2. The user submits the request to the VTA.
3. The VTA receives the request and ”understands” it.
4. Based on the previous understanding, the VTA constructs a request cancellation for the tourism service providers that booked the flight to be cancelled.
5. For each of the tourism service providers involved in the booking:
   (a) The VTA submits its cancellation request to the tourism service provider.
   (b) The tourism service provider receives the request and processes it.
   (c) The tourism service provider performs the cancelling, returns (a percentage of) the flight price to the customer, and returns a confirmation to the VTA.
   (d) The VTA receives the confirmation of the cancellation from the tourism service provider.
6. The VTA, after collecting all the confirmations from the tourism service providers, aggregates them.
7. The VTA delivers the aggregated confirmations to the requester.
8. The customer receives the confirmation for the cancellation of his flight.

Possible Extensions:

1. The customer wants to include new non-functional properties for his request, involving security, privacy, etc.
2. Some of the tourism service providers that provided the booking are now off-line.
3. The confirmations of the different tourism service providers use different terminologies.

4. Customer requests VTA for offers for a complete trip.

Goal/Context: The customer wants to make a trip from Innsbruck to Argentina in December 2005, visiting Buenos Aires, the Iguazu falls and the glaciers national park in 10 days. He wants to stay in 4 starts hotels, to have breakfast and dinner included, and the preferred way of travelling in Argentina is by plane.

The customer will send his request to the VTA, which has to take care of locating the necessary tourism service providers to build offers for a trip meeting the customer requirements.

The VTA will contact different tourism service providers that provide flights from Innsbruck to Argentina, flights in Argentina, and hotels in Buenos Aires, the Iguazu falls and the glaciers national park to know their offers. Finally, different offers for the complete trip will be returned to the customer, who will be able to select one of them and book it.

Participating actors: Customer, VTA, tourism service providers.

Scenario/Steps:

1. The user constructs a trip request, including all his requirements and preferences.
2. The user submits the request to the VTA.
3. The VTA receives the request and ”understands” it.
4. Based on the previous understanding, the VTA constructs a request for the complete trip or parts of it for the available tourism service providers.
5. For each of the tourism service providers providing a part of the complete trip:
   (a) The VTA submits its request (flight, hotel or both) to the tourism service provider.
   (b) The tourism service provider receives the request and processes it.

   (c) The tourism service provider selects the relevant information for the request and returns it to the VTA.

   (d) The VTA receives the information from the tourism service provider.

6. The VTA, after collecting all the information from the tourism service providers, aggregates it and constructs several trip offers.

7. The VTA delivers the aggregated offers to the requester.

8. The customer receives the offers for his trip.

Possible Extensions:

1. There is no single tourism service provider providing the complete trip, but several combinations of providers can provide it.

2. It is not possible to build an offer that completely fulfills the customer request e.g. there is no 4 starts hotel available at the Iguazu falls.

3. The tourism service providers use different terminologies.

4. A complete trip meeting the requirements cannot be found at the moment, but the customer wants to be notified if there is any suitable offer available during this month.

Chapter 6

B2B Use Cases

6.1 Supply Chain

In this section, a dynamic supply chain use case is presented. In the following, we will describe the general purpose of the use case, the actors involved, and a set of usage scenarios that emerge from the use case. These will lead to different requirements that will be summarized at the end of the section.

6.1.1 Introduction of the Use Case

Supply chains are used in a number of markets, combining multiple services with the final goal of providing different goods to a given business. Supply chains can involve, for example, the selling of computers, and the shipping of the computers to the buyer.

With current technologies, the supply chain is completely defined at design time i.e. the providers involved are manually selected, probably as a result of prior agreements, and their interactions with the buyer hard-wired. The providers used to fulfill the concrete supply needs that appear at run-time can only be selected among a set of pre-defined, pre-agreed providers. Therefore, the buyer will have a limited number of choices to find the best offer to, for example, ship some goods from New York to Munich. In addition, if a provider goes off-line and no alternate provider was defined when designing the supply chain, the supply will fail. Summarizing, the selection of available providers to complete the supply chain is limited to the pre-selected providers, and the supply chain cannot be dynamically reconfigured to react to changes e.g. new providers entering the market or pre-selected providers going off-line.

Dynamic, reconfigurable supply chains are a step beyond traditional supply chains. Instead of having a rigid configuration, they introduce dynamics by automatically locating the best providers for a given supply need. This considerably decreases the effort required for configuring the supply chain and optimizes the supply process by always selecting the

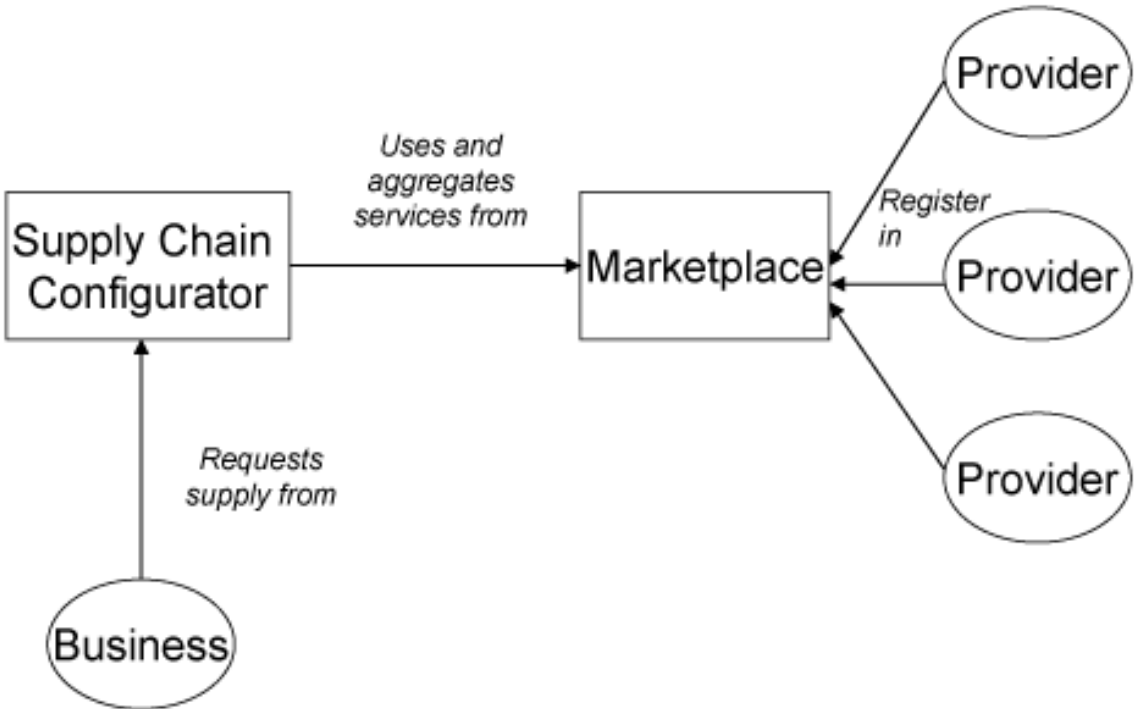

Figure 6.1: Supply chain overview

best providers based on some given requirements and preferences.

The use case presented in this section is a dynamic supply chain, where a supply need is dynamically resolved by selecting the most appropriate providers among the available ones. A business requiring the supply of some goods will get the appropriate supply chain dynamically configured to fulfill his needs. Other businesses offering the purchasing and shipping of goods will be registered in a marketplace.

In a nutshell, our use case will provide the following functionality: given a supply need, the most appropriate providers registered in the marketplace will be selected and arranged together to purchase and ship the required goods. Figure 6.1 gives an overview of the use case.

The figure outlines the general structure of the supply chain dynamic configuration. A given business has a supply request that the supply chain configuring process will resolve by locating, aggregating and using services from the providers registered in the marketplace.

6.1.2 Actors, Roles and Goals

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

- Requester business: business that has a concrete supply need.
  - Goal: automated resolution of his supply need by the supply chain configurator in a transparent fashion.
  - Role: interacts with the supply chain configurator for requesting a supply and for tasks related to the supply such as payment and confirmation of the request.
- Supply chain configurator: process that configures a supply chain given a supply request. It is not necessarily a third party but it can be a service integrated in the requester business, and it will provide its service for free.
  - Goal: resolve a supply request.
  - Role: provide dynamic supply chain configuration.
- Provider: provide purchasing and shipping of goods.
  - Goal: sell goods or services to other businesses.
  - Role: provide the services that will be used and aggregated to configure the supply chain.

6.1.3 Usage Scenarios

1. Requester business needs computers supply.

Goal/Context: The requester business wants to buy 100 computers with certain hardware requirements and one year guarantee. It wants to get the computers shipped to Innsbruck within 5 days (faster shipping is preferred), and the total price for the computers and the shipping cannot exceed 200.000 euros.

The requester will build its request and pass it to the supply chain configurator, which will detect computer providers in the marketplace that can offer computers with the required hardware characteristics, in sufficient number, with delivery to Innsbruck within 5 days or less, and for a price which meets the maximum stated by the requester business.

After locating the appropriate provider or providers, the supply chain configurator will build the supply chain with the necessary providers and will interact with them to get the computers shipped.

Finally, the requester business will get a confirmation of the purchase and shipping, and eventually the computers in Innsbruck.

Participating actors: requester business, supply chain configurator, providers.

Scenario/Steps:

1. The requester business constructs a request for buying and getting the computers in Innsbruck with the mentioned constraints.
2. The requester business passes its request to the supply chain configurator.
3. The supply chain configurator gets the request.
4. The supply chain configurator builds a request to locate suitable providers.
5. Based on the previous request, providers that are suitable for providing the requested computers are located.
6. The supply chain configurator builds a supply chain for the purchase and shipping of the computers.
7. The supply chain is used to purchase and ship the computers.
8. A confirmation of the supply is returned to the requester business.
9. The requester business gets the confirmation and details of the computers bought and the shipping.

Possible Extensions:

1. The cheapest price is required for the computers and the shipping.
2. The requester business requires to have a certified confirmation of the purchase and the shipping.
3. The requester business will only buy computers from manufacturers with a certain ISO certification.
4. There is no provider available for providing 100 computers, but two different providers can together provide the required number of computers.
5. The total number of computers have to be purchased and delivered or none of them i.e. no partial supplies are accepted.
6. There is no direct shipping service from the computer manufacturer location to Innsbruck.
7. The providers use different terminology than the requester business.
8. Different providers are combined and some of them use EDI for communicating while others use RossettaNet.

9. The requester business will not give payment information in his first request but only after the supply chain has been configured.

10. The requester wants to reuse this supply chain for future requests.

2. Requester business needs shipping.

Goal/Context: The requester business has a partially configured supply chain for purchasing and shipping computers. While the purchase is already defined to be done with a given set of computer manufacturers and the details of the interaction with them are fully defined, the shipping provider will be dynamically located depending on the location of the computer manufacturer from which the computers have been bought, on the number of computers to be shipped, and on the time constraints for the delivery.

In this context, the requester business will ask the supply chain configurator to satisfy a shipping request with providers that meet its constraints and whose interaction style, protocol, etc. can fit its partially pre-defined supply chain.

With this information, the supply chain configurator will locate suitable providers at run-time depending on the concrete shipping need, and will integrate the shipping with the requester business supply chain.

Participating actors: requester business, supply chain configurator, providers.

Scenario/Steps:

1. The requester business constructs a request for shipping a certain number of computers, from a given location to Innsbruck, and within a certain period of time.

2. The requester business passes its request to the supply chain configurator.

3. The supply chain configurator gets the request.

4. The supply chain configurator builds a request to locate suitable shippers.

5. Based on the previous request, shippers that are suitable for shipping the computers are located.

6. The supply chain configurator builds the interaction of the shippers with the requester business supply chain.

7. The supply chain is used to purchase and ship the computers.

Possible Extensions:

1. The requester business requires the shipping to have delivery guarantee.
2. The requester business has DHL as its preferred shipper.
3. The shippers selected have to use RossettaNet.
4. The use of multiple shippers to get the computers in Innsbruck is not acceptable.
5. The shippers have to provide encryption.
6. The requester business only accepts payment by VISA credit card.

Chapter 7

E-Government

In several application domains, service-oriented computing allows to integrate distributed business processes which deal with different organizations, companies and offices spread out over a country or the world. Some paradigmatic examples might derive from the development of web-services applications in the e-government domain.

Even if different units have internal and local needs, they have to interact and cooperate according to general and global requirements. Therefore at any time there is an interaction among partners, two opposite and often conflicting kinds of needs have to be taken into account:

- the global business rules: the rules of engagement between different organizations;
- the local business rules: the local rules which derive from internal business needs.

In the e-government domain, a huge amount of admistrative offices or departments have to take into account both local needs and the global rules deriving from European, national, regional requirements and norms. Then, global and local rules, having opposite goals and being represented with different conceptual schemas (systems of taxonomies, ontologies, etc.), might create some conflicts.

In the proposed business-use case the web-services applications and processes interleave the phases of specification of both global and local rules through negotiation between global and local needs. As a consequence, the choreography (i.e., the global view of how different partners interact), as well as the orchestration (i.e., the description of how one partner interacts with (some of) the other partners), as well as the internal business process of each partner, are incrementally built through negotiation steps.

7.1 Actors, Roles and Goals

As a typical business practices, we consider the specific licensing procedure for the establishment and management of a Wastes Disposal or Recycling Facility.

Starting from specific regulations and norms (D.C.I. 27 luglio 1984; L.R. 13 aprile 1995 n.59; D.Lgs. 5 febbraio 1997) and interacting with various actors involved in the analysis processes, any citizen proposing to establish or manage a facility for solid wastes disposal has to apply for a certificate of designation. The application must be accompanied by all documents specified in the norms (e.g. an engineering design and operations report), must be registered by the Protocol Office (PO), and then must be reviewed and validated by the Waste Management Office (WMO). If it is necessary, the WMO will interact with the citizen asking for additional information or clarifications until the documentation will be completed. This process must be completed within 30 days from the registration by the PO. Then, the WMO is in charge of appointing the Technical Commission (TC), which is composed by various consultants and directors of public agencies (e.g. Sanitary Agency, Water Quality Control Agency, Soil Water and Plant Testing Laboratories, Environment Engineers,..). Each member of the TC has to produce a technical report and has to send it to the WMO. The WMO will notify the working progress to all the participants (TC members, citizen, WMO's responsible for the application). The TC evaluates all the technical reports, determining whether the facility complies with the specific norms. Finally the WMO will be in charge of producing the recommended determination and sending it to the Province Board (PB) and to the citizen within 90 days from the TC evaluation. The PB will evaluate the recommendations, will draft the final determination and finally will notify the citizen. Each application for a solid waste disposal site or facility should complete within 150 days from the P.O. registration.

As explained the process is very complex, but it can be summarize in two steps as shown in Figure 7.1 and explained as follows:

- a citizen submits an application to obtain the license for its waste disposal or recycling facility (incinerator, private landfill);

- the local government, only if it complies with high standards dictated by norms and involving various agencies and experts, will evaluate the proposal and will authorize it.

Application Request

Preconditions: The citizen needs a certification

Postconditions: The description of the post is submitted at a specific office, that autonomously manages the whole process

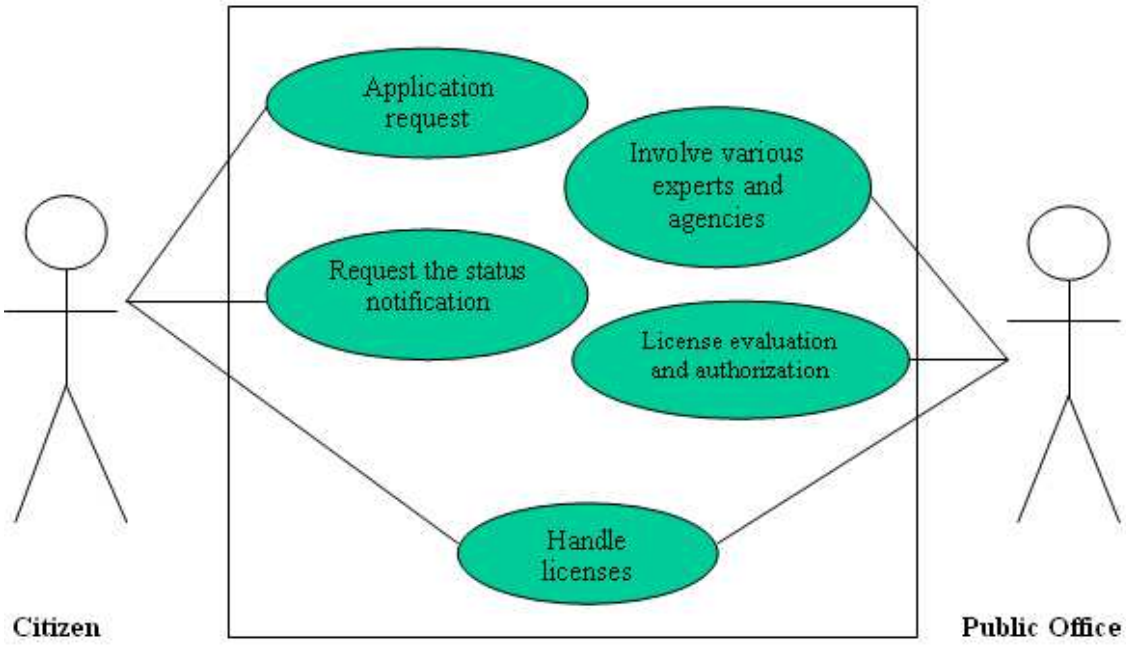

Figure 7.1: UML Use Case Diagram

Actor	Response
The actor requests an application/certificate	Request a certificate *The application must be accompanied by all documents specified by the norms*
The citizen obtains a protocol number and references of the WMO	Validate the request Send the request to the WMO, that has to interact with the citizen Make the requested procedure publicly accessible on the web

Involve various experts and agencies

Preconditions: The public office needs to know who else will participate to the process

Postconditions: The office validates the whole process

Actor	Response
The office interacts with different parties involved in the evaluation process at the strategic level relying on the intent/offer matching mechanism	Request a participation of expert and agencies *Contribution requests must be addressed to the right offices, and the documentation must be enclosed in the requests*
The office obtains and validates the contributions of experts and agencies	Obtain and validate the contributions, and make the result publicly accessible on the web

Request the status notification

Preconditions: The application is not finished yet; the citizen knows the protocol number of her application

Postconditions: The office gives information about the status of her application

Actor	Response
The citizen asks for additional information or clarifications until the documentation is complete	Request a status notification
The citizen obtains a status notification	The WMO office evaluate the whole process and publicly status information on the web

License evaluation and authorization

Preconditions: All experts and agencies have contributed to the application

Postconditions: The office publicly the licence

Actor	Response
The office obtains all the needed certifications from all the actors	All the actors send their contribution to the central office
The central office sends the licence to the citizen	The central office validates all the contribution and make public the licence on the web

Handle licenses

Preconditions: The process is finished for two possibly reasons:

- the process is validated and the license is given to the citizen;
- the procedure is finished for non conformity with the local and general requirements, or because of the process run out of time.

Postconditions: The citizen is informed of the applicants interest and has received the

licence documentation

Actor	Response
The office handles the license in local Databases	Package the information in a suitable form; send the license to the citizen and manage internal information *Send a communication to the applicant informing him/her of the finished process*

7.2 Usage Scenarios

The Environmental Protection Agency (EPA) is a local agency which deals with a wide range of environmental matters including protecting air, water and soil quality, managing waste, preventing or controlling pollution and promoting sustainable industry. As showed in Figure 1, the e-government domain is very complex, and EPA has to deal with complex administrative procedures to address these issues. Most of these procedures are distributed among various actors (its internal offices and other external entities) and are regulated by law. As it is shown in Figure 7.2 this processes are managed through Internet and partners share knowledge participating to same processes, interacting, exchanging documentation and providing evaluations and documentations. For instance the Ente Amministrazione Provinciale interact with ARP and SIR, to obtain documentation, information about processes and validation of some kind of applications. In any case each unit manages knowledge in an autonomous way, that better suits the local needs. In fact, global and local rules such as, European, National, Regional and Local Norms contribute to specialize the same procedure adding new constraints, new actors and goals for each unit. Norms influence the collection of goals and activities delegated to specific actors, and impose specific constraints and obligations concerning minimal and maximal durations of steps (or of the overall procedure), or limiting the period during which an action can be performed. In other words each unit manages personalized knowledge representation and processes, which better suit local needs, norms and issues.

7.3 Summary of Use Case Requirements

7.3.1 System Requirements Analysis

The distinction between global and local needs is very helpful also in the requirements analysis. In particular we distinguish and represent both the *global business rules* and the *local business rules*.

From the *choreography perspective* (global business rules), the requirements aim to

describe the interaction opportunities that are presented in the business domains. These opportunities are defined in terms of possible matches between the services required by certain actors and the services provided by other actors. The choreographical requirements also define the shared assumptions and constraints used to correctly interact inside the domain. In the *orchestration perspective* (local business rules), the point of view of a particular business party is taken into account, its strategic goals are described, the definition of the business processes needed to achieve these goals is made evident along with the decision of what services to implement internally and what external services to exploit.

The global requirement that the whole process must complete in a fixed elapsed time from the activation is represented with dependency among all the partners. The necessity of "transparency towards citizens" is represented with the dependency of other goals which are implicitly associated to all the actors participating to the business process. This can be represented by global rules of the case study.

Based on high-level description of choreography representation of the global requirements and rules, a business analyst can synthesize a choreography for the business process where each activity is linked to the corresponding global requirements. For instance Figure 3 depicts a view of the choreography resulting from the negotiation phase where only activities are represented for readability purposes. The intention of the Citizen are operationalized into activity (hexagons) activated within a process. The Citizen first submits the application SubmitApplication to the ProtocolOffice. This office is responsible to register the application (RegisterApplication) and then to provide application information (ProvideRegistrationInformation) to the WasteManagementOffice. The WasteManagementOffice initialise the application by validating the documents, analysing the content according to its needs and interpretation schemas, and finally asking the Citizen for further clarifications only once (ManageDocuments) it will be necessary. Whenever the validation fails (i.e. the WMO don't have enough information or cannot understand the documentation content) a rejection acknowledgement for the application is sent to the Citizen. Otherwise, the application is analyzed (AnalyzeApplication) by submitting it to a technical commission that formulates a technical report for the application. Then a conference is organized (OrganizeConference) among the Citizen, the WasteManagementOffice and the TechicalCommission to produce a recommended determination (ProvideRecommededDetermination)

Some other system requirements depend on the processed described above:

Application Request

System Requirements

- The application request can be executed online;
- The applications request must be submitted by the citizen through authentication processes such as digital signatures, identity cards;

- The application must be accompanied by all documents specified by norms;
- The request is validated for inconsistencies and stored with a protocol number in a database accessible by web.

Involve various experts and agencies

System Requirements

- All the actors involved in the process must be contacted and validated;
- Each request to the actors must be comprehensible;
- Everyone, dealing with personal and local needs, and knowledge representation have to coordinate with other to produce some common results;
- The intent/offer matching mechanism can be negotiated among actors.

Request the status notification

System Requirements

- The request status notification can be executed online.
- The status notification can be delivered by matching the document status of different actors.

License evaluation and authorization

System Requirements

- Each answer from the actors must be consistent and validated (thought authentication procedures);
- The licence evaluation can be executed online;
- The licence authorization can be executed online.

7.3.2 System Evaluation

Most of the e-government services are not developed on line. The most important web applications are used for publicly general information, and for communicate and notify procedures among offices spread out over the country. Processes and services, described above, are already available, or at least clearly identifiable when not yet implemented electronically.

In any case, the processes are often organized and coordinated by a superpartes unit which takes into account all possible emerging interaction among partners and develops a centralized and general architecture. It is strongly desiderable to have an application domain in which:

- Some controllers, at different levels, exist (i.e. Local, National, European);
- Only one supervisor requests service procedures, controls the correctness of:
  - processes;
  - the system of interconnections and dependencies of sub-processes;
  - the global lifecycle of the service, etc.
- Actors, which contribute to the global service, act in an autonomous way, and are not controlled by supervisors or controllers. The kind of contributions can be information exchange among actors such as:
  - Synchronous or asynchronous communications;
  - Sequences of parallel activities.
- The citizens apply a request only to one office, and even if some other actors participate and contribute, the citizen will obtain a complete answer from the same office. The process will generate and negotiate some dynamic requests to the actors who have to contribute by adding information or by validating some documentation. This process might be iterated all the time an actor needs support from others, until the answer is well formed.

Moreover, in this section we want to stress the fact that the designers usually focus on "how" different organizations must interact, and how a business process is implemented. However, most of the used languages are not able to describe the existing "why" and "what" behind the business service, and are not able to analyse these needs and expectations within the business-use case. A new language is needed, such as Tropos. Figure 7.3 is a Tropos diagram that describes the actors (circles), and all their strategic global shared goals (the ovals attached to the actors). For instance in the diagram we have the Citizen that aims to obtain a waste facility licence (goal GetWasteFacilityLicence); the WasteManagementOffice that aims to handle with the several applications for getting a licence (goal ManageApplication); the ProtocolOffice that aims to register applications by associating to each application a protocol number (RegisterIncomingApplications); the TechicalCommission that aims to evaluate applications by getting the opinion from experts (goal ExpertAnalysis); finally, there is the ProvinceBoard aiming to ratify determinations (DeterminizationRatification). Tropos allows also to describe the interactions among the different parties involved in the process at the strategic level relying on the intent/offer matching mechanism represented in the diagram by means of dependencies

(the ovals linked to two different actors). For instance the Citizen depends on the ProtocolOffice for the activation of the application to obtain a waste facility licence (ActivateApplicationManagement). Moreover, it depends on the ProvinceBoard for the final determination (dependency ProvdeFinalDetermination). The ProtocolOffice depends on the Citizen for the submission of the application (SubmitApplication). The WasteManagementOffice depends on: the ProtocolOffice for the registration of the application (dependency ProvideRegistrationInformation); the TechicalCommission for the technical reports from experts (ProvideReports); and the Citizen for application documents (ProvideDocuments). The ProvinceBoard depends on the WasteManagementOffice for the recommended determination.

With this diagram we can see how different units, participating to a business process, have to interact and cooperate according to general and global requirements. Moreover, each of them has its own internal business needs, which are specific to the business they must or want to carry out.

7.3.3 Evaluation of Semantic Web Technologies

The development process described before is supported by a precise definition of the global and local requirements. These requirements are underlying all the local knowledge representation systems (such as taxonomies, partonomies, and ontologies). With the aim of avoiding design or implementation problems, the use of *verification* tools becomes fundamental. Having analysed the specification, the verification tools provide a set of results that may be used further in the system refinement and analysis process. These results may represent possible correct scenarios of the system execution, invalid traces leading to the goal failure, incorrect scenarios for the cases where the specification error is detected.

Choreography/Orchestration

In order to catch misunderstandings and inconsistencies in this model one can verify it against set of properties that every execution of the system should satisfy (assertion properties) or some execution may satisfy (possibility properties). Querying the model allows one to check the correctness of the model with respect to the property or to check whether the model is not over-specified and some desirable behaviours are captured by the system.

Negotiation

To achieve better agreement between business parties, to allow better understanding of the whole picture, and to resolve conflicts among actors, the negotiation is used in different phases of the process:

- *Global and local requirement models negotiation*. At this phase the global, externalized goal model should be brought into line with the local models of each parties. The requirements imposed by partners and the process to a business party

may often conflict with its internal requirements. Here the local models are verified against the global model and erroneous scenarios are aimed to present what are the requirements that should be weakened/strengthened in order to resolve the conflict.

- *Global choreography process and local goal model negotiation*. This phase aims to bring the global process model extracted from the global goal model into conformance with the parties local needs and expectations. Here the process is verified against local needs and revealed inconsistencies assist the developer to correct the process.

- *Local process and global goal model negotiation*. Here the orchestration of the business party is validated against the global requirements and rules elaborated so far. Here one can verify whether the process does not conflict with the goals and rules of the partners and the whole domain.

- *Local process and global orchestration model negotiation*. This step is necessary to verify that the internal implementation of the particular party preserves the properties assigned to the global process. In this verification the part of the global process corresponding to the analysed business party is replaced with its local process. The model obtained in this way is verified against the behavioural properties like deadlocks, livelocks, time constraints etc.

Migration Needs

As it's shown in Figure 7.4 it is necessary to migrate from norms to business process. The refinement of the goal model, into the global process model, should also be supported by the verification techniques. First, it is necessary to check that the process itself is correct. Correctness of the process assumes the absence of anomalies such as "deadlocks" (when a execution is "blocked" and no longer proceeds through the process) and "livelocks" (when an execution becomes "stuck" in a never-ending loop), satisfaction of certain properties, time constraints etc.

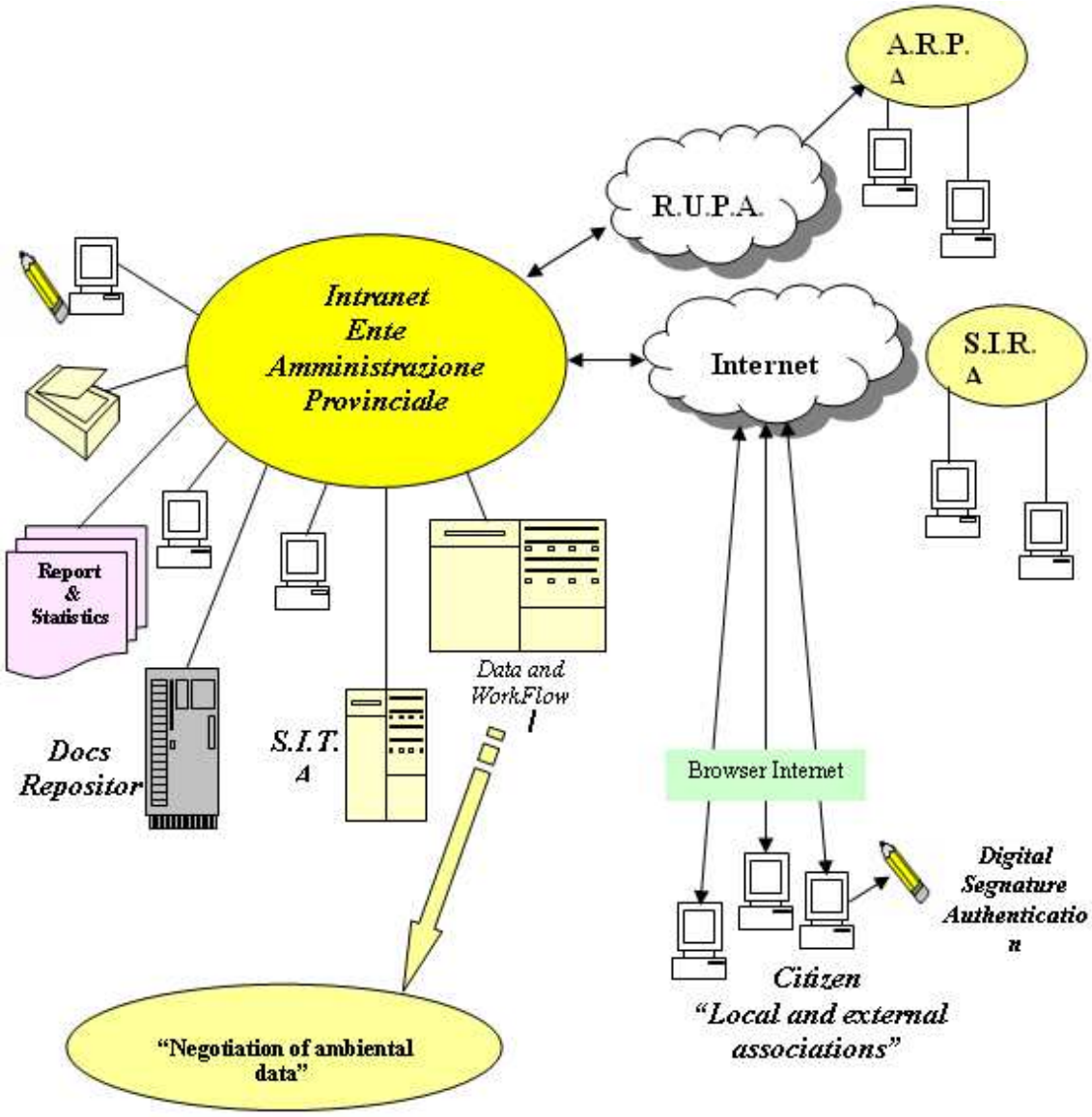

Figure 7.2: E-government business case

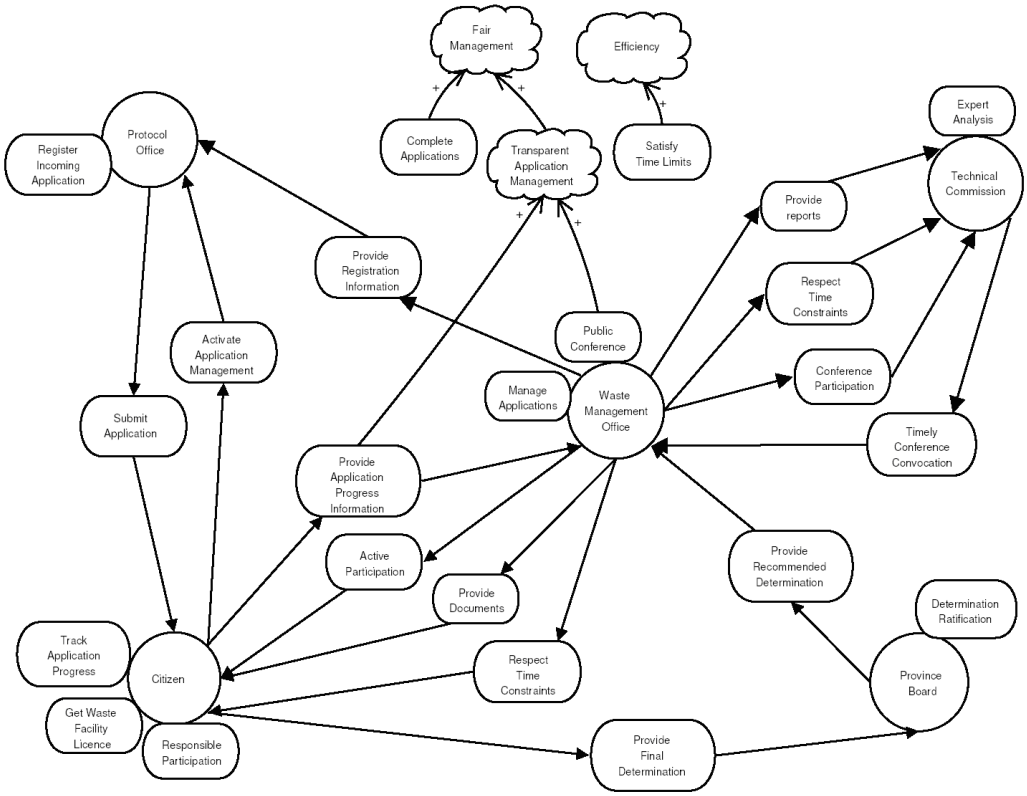

Figure 7.3: The Global Rational Requirements Diagram

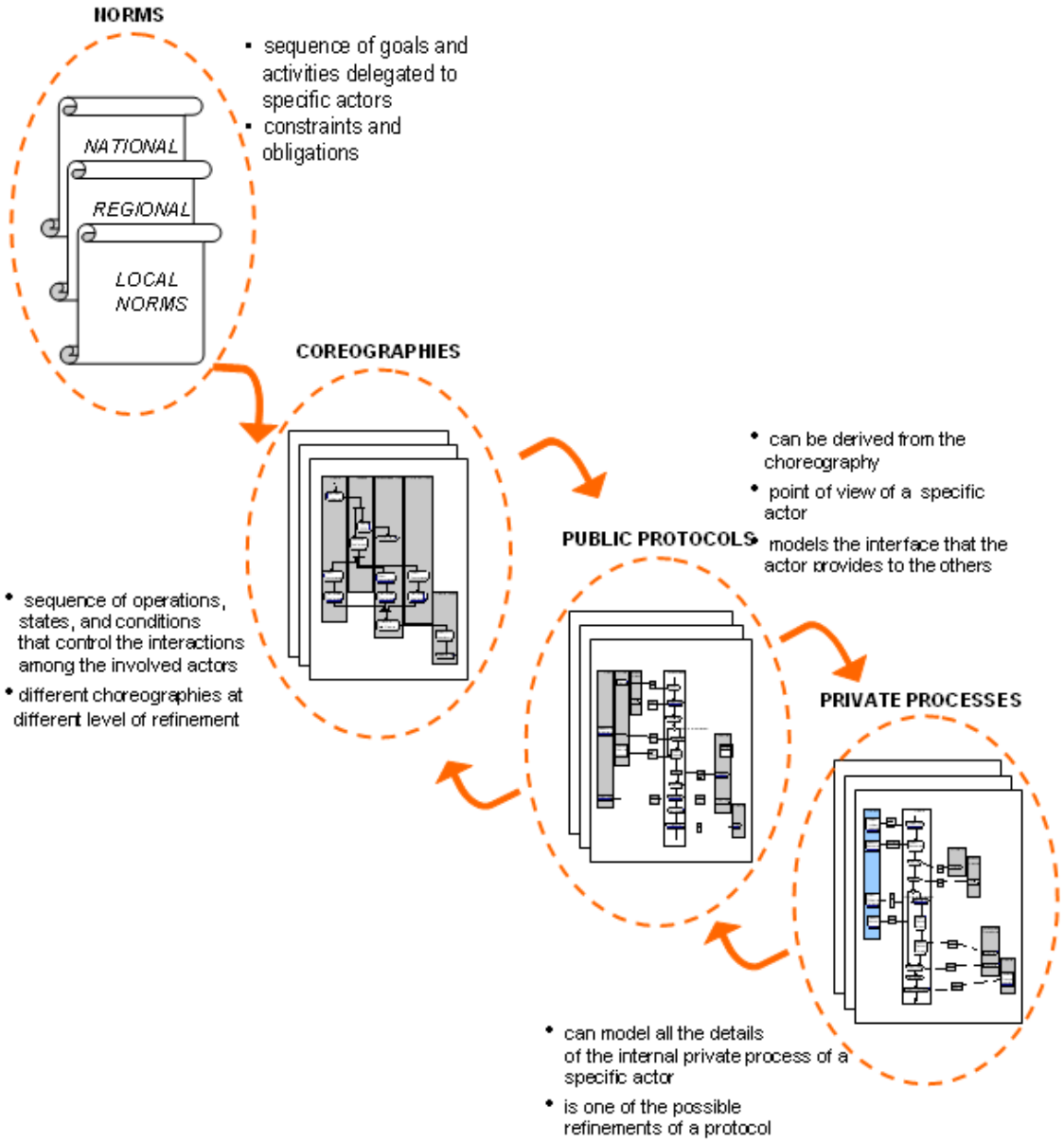

Figure 7.4: The migration system procedure

Chapter 8

Summary of Requirements

In this chapter, we briefly summarise the requirements that we have specified in various use cases presented in the above chapters.

Requirements on Domain Ontologies

Domain ontologies should provide the following in order to give semantics for requests and results:

1. precise and complete vocabulary, and
2. object and datatype constraints.

Requirements on Web Services

Web services should be able to

1. have their request, result, functionalities and non-functional characteristics formally described,
2. publish their interface, functionalities and non-functional characteristics.

Requirements on Mediators

Mediations are required to enable understanding between

1. heterogeneous actors at the ontology level and
2. service requesting applications and service providers at the choreography level.

A third kind of mediator should be able to provide matchmaking of Web services.

Chapter 9

Conclusions and Future Work

In this document we described several use case scenarios in order to analyze requirements regarding semantic enabled process integration, i.e. requirements for semantic description of web services. To do that, we selected several scenarios, such as Application to Application integration (A2A), Business to Business (B2B), Business to Customer (B2C) as well as scenario involving e-government processes. As a conclusion, requirements on domain ontologies, web services and mediators were identified so that described integration scenarios could be provided automatically and efficiently.

Collected requirements from this work are essential source for both short term as well as long term activities of Semantic Web Services of WP2.4. In particular, such requirements will be essential for developing and improving Web Services Modelling Ontology (WSMO)[RLK04] and Web Services Modelling Language (WSML)[dBLK$^+$04] within conceptual and formal framework for semantic web services as well as for web service discovery and composition.

Bibliography

[Bur00] S. Burbeck. The tao of e-business services. http://www-106.ibm.com/developerworks/webservices/library/ws-tao/?dwzone=webservices, October 2000.

[dBLK+04] Jos de Bruijn, Holger Lausen, Reto Krummenacher, Axel Polleres, and Dieter Fensel. *The WSML Family of Representation Languages*. 2004. WSML Working Draft 24 December 2004. Available from http://www.wsmo.org/2004/d16/d16.1/v0.2/.

[DRRB03] F. T. Dabous, F. A. Rabhi, P. K. Ray, and B. Benatallah. Middleware technologies for b2b integration. *Annual Review of Communications*, 56, November 2003.

[HHO04] H. He, H. Haas, and D. Orchard. Web services architecture usage scenarios. Working draft, W3C, February 2004. Available at http://www.w3.org/TR/ws-arch-scenarios/.

[RLK04] Dumitru Roman, Holger Lausen, and Uwe Keller, editors. *Web Service Modeling Ontology (WSMO)*. 2004. WSMO Working Draft D2v1.0, September 2004. Available from http://www.wsmo.org/2004/d2/v1.0/.