

# ATHENA Interoperability Framework (AIF)

## Table of contents

1	Introduction.....	8
1.1	ATHENA Interoperability Framework (AIF).....	8
1.1.1	About.....	8
1.1.2	Contributions.....	8
1.1.3	Acknowledgements.....	8
1.2	Background and motivation.....	9
1.2.1	What is interoperability?.....	9
1.2.2	ATHENA's approach to interoperability.....	11
1.3	Overview of the AIF.....	16
1.3.1	What is an interoperability framework?.....	16
1.3.2	Structure of the AIF.....	16
1.3.3	Interoperability reference architecture.....	17
1.3.4	Interoperability methodology framework.....	18
1.3.5	Interoperability profiles.....	19
1.4	Specification of the AIF.....	20
1.4.1	AIF conceptual model.....	20
1.4.2	AIF knowledge model.....	21
1.4.3	ATHENA knowledge base.....	22
1.5	References.....	23
1.5.1	Bibliography.....	23
2	Reference architecture.....	32
2.1	Reference architecture.....	32
2.1.1	Introduction.....	32
2.2	Interoperability reference architecture.....	32
2.2.1	Overview.....	32

2.3	Characterisation of interoperability issues.....	35
2.3.1	Interoperability classification.....	35
3	Interoperability methodology.....	42
3.1	Interoperability methodology.....	42
3.1.1	Introduction.....	42
3.1.2	ATHENA Interoperability Methodology (AIM).....	42
3.2	Baseline interoperability methodology.....	48
3.2.1	Overview.....	48
3.3	Business interoperability framework (BIF).....	50
3.3.1	Goal.....	50
3.3.2	Description.....	50
3.3.3	Resources.....	51
3.4	Enterprise interoperability maturity model (EIMM).....	51
3.4.1	Introduction.....	51
3.4.2	Solution - EIMM assessment.....	52
3.4.3	Resources.....	61
3.5	Interoperability analysis.....	61
3.5.1	Interoperability analysis.....	62
3.6	Requirements - solution mapping.....	64
3.6.1	Introduction.....	64
3.6.2	Annotation of business needs and solution.....	65
3.6.3	Mapping application by using the Protégé toolset.....	69
3.6.4	Analysis.....	70
3.7	Testing methodology.....	73
3.7.1	Goals.....	73
3.7.2	Description.....	73
3.7.3	Activities.....	74
3.8	Implementation.....	74
3.8.1	Overview.....	74
3.9	Interoperability impact analysis model (IIAM).....	74
3.9.1	Goal.....	74
3.9.2	Description.....	75

3.9.3 Resources.....	76
4 Guidelines and best practices.....	76
4.1 Guidelines and best practices.....	76
4.1.1 Introduction.....	76
4.2 Collaborative enterprise modelling.....	77
4.2.1 Introduction.....	77
4.2.2 Solution.....	77
4.2.3 Application.....	79
4.3 Cross-organisational business processes - Application guidelines and design rules... 80	
4.3.1 Introduction.....	80
4.3.2 Design rules.....	80
4.3.3 Application procedure.....	81
4.3.4 Guidelines and relevant issues.....	83
4.4 Flexible execution and composition of services.....	95
4.4.1 ATHENA Service-Oriented Interoperability (SOI) Framework.....	95
4.5 Information interoperability.....	95
4.5.1 Introduction.....	96
4.6 Semantics and ontologies - Application guidelines.....	96
4.6.1 Introduction.....	96
4.6.2 Preparation phase.....	97
4.6.3 Design phase.....	98
4.6.4 Run-time phase and testing.....	101
4.6.5 Integration phase.....	101
4.7 Dynamic architectures.....	102
4.7.1 ATHENA Model-Driven Interoperability (MDI) Framework.....	102
5 Technical architecture.....	103
5.1 Technical architecture.....	103
5.1.1 Service-oriented architecture.....	103
5.1.2 Technical architecture of the ATHENA platform.....	104
5.2 Collaborative enterprise modelling.....	105
5.2.1 Introduction.....	105
5.2.2 Solution.....	106

5.2.3 Application case.....	107
5.2.4 EM market.....	108
5.3 Cross-organisational business processes.....	108
5.3.1 Introduction.....	109
5.3.2 Tools.....	110
5.3.3 Model transformations.....	110
5.3.4 Need for task management.....	112
5.4 Flexible execution and composition of services.....	113
5.4.1 SOA framework.....	113
5.4.2 Modelling.....	114
5.4.3 Services.....	115
5.4.4 Agent.....	116
5.4.5 Integrated execution platform.....	118
5.5 Information interoperability.....	118
5.5.1 Introduction.....	118
5.5.2 Design-time architecture for business documents.....	119
5.5.3 Business document and mapping tools.....	120
5.6 Semantics and ontologies.....	121
5.6.1 Introduction.....	121
5.6.2 Semantic reconciliation suite.....	121
5.6.3 Semantic enhancement for SOA and UMT2OWLS.....	123
5.7 Model-driven interoperability.....	123
5.7.1 Introduction.....	123
5.7.2 Model-driven and adaptive interoperability development environment.....	124
5.7.3 Model-driven and adaptive interoperability runtime environment.....	125
6 Interoperability profiles.....	125
6.1 Interoperability profiles.....	125
6.1.1 Introduction.....	125
6.1.2 Interoperability profile definition, usage and lessons learnt.....	126
6.2 Guidelines for interoperability profiles.....	127
6.2.1 Procedure.....	127
6.2.2 Templates and supporting tools.....	132

6.3 Collaborative product development (CPD) profile.....	136
6.3.1 Collaborative product development (CPD) pilot.....	136
6.4 Networked collaborative product development (NCPD) profile.....	139
6.4.1 Networked collaborative product development (NCPD) pilot.....	139
6.5 Electronic procurement (e-Procurement) profile.....	140
6.5.1 Furniture pilot.....	140
6.6 Product portfolio management (PPM) profile.....	143
6.6.1 Product portfolio management (PPM) pilot.....	143
7 ATHENA solutions.....	145
7.1 ATHENA solutions.....	145
7.1.1 Composite solutions.....	145
7.1.2 Individual solutions.....	146
7.1.3 Demonstrators.....	148
7.2 Composite solutions.....	148
7.2.1 ATHENA Interoperability Framework (AIF).....	148
7.2.2 ATHENA Interoperability Project Support Suite.....	150
7.2.3 ATHENA Model-Driven Interoperability (MDI) Framework.....	151
7.2.4 ATHENA Modelling Tool Suite for Eclipse/RSM.....	153
7.2.5 Semantic reconciliation framework.....	155
7.2.6 ATHENA Service-Oriented Interoperability (SOI) Framework.....	157
7.3 Individual solutions.....	159
7.3.1 A*.....	159
7.3.2 Active Knowledge Model (AKM) Execution Platform.....	161
7.3.3 Active object flow (AOF) methodology.....	164
7.3.4 Agent-based Computing Architecture and Design/Runtime Platform.....	165
7.3.5 Autonomous Computing Engineering Reference Guide.....	167
7.3.6 ARES.....	168
7.3.7 ARIS (Architecture of Integrated Information Systems).....	170
7.3.8 ARGOS.....	171
7.3.9 ASSERT.....	173
7.3.10 ATHENA Event and Document Correlation (AEDoC).....	174
7.3.11 ATHOS.....	175

7.3.12 BPEL Metamodel Feature for Eclipse.....	177
7.3.13 Business Issues Planner (BIP).....	178
7.3.14 BRMF (Business Resource Management Framework).....	179
7.3.15 Capability Tables Loader (CTL).....	182
7.3.16 Conformance testing suite.....	183
7.3.17 EKA Metamodel Feature for Eclipse.....	185
7.3.18 EKA to PIM4SOA Transformation Feature for Eclipse.....	186
7.3.19 Enterprise Interoperability Degree Measurement (EIDM).....	188
7.3.20 Enterprise Interoperability Maturity Model (EIMM).....	189
7.3.21 Enterprise Interoperability Maturity Model for SMEs (EIMM-SME).....	191
7.3.22 EXP2PIM4SOA (Express to PIM4SOA model transformation).....	192
7.3.23 EXP2SCH (Express to Schematron model transformation).....	193
7.3.24 EXP2UML (Express to UML model transformation).....	194
7.3.25 EXP2XMI (Express to XMI model transformation).....	195
7.3.26 EXP2XSD (Express to XSD model transformation).....	197
7.3.27 Gabriel.....	198
7.3.28 Gap Table Analyser (GTA).....	200
7.3.29 GRAI Methodology.....	201
7.3.30 GraiTools.....	202
7.3.31 Solution template.....	203
7.3.32 Johnson.....	204
7.3.33 Maestro.....	206
7.3.34 Mediated Support Tool (MST).....	208
7.3.35 Metis.....	209
7.3.36 Model-driven integration of JACK and Web Services.....	210
7.3.37 MOF repository.....	212
7.3.38 MO <sup>2</sup> GO (Method for Object Oriented Business Process Optimization).....	213
7.3.39 Monitoring Support Tool (MST).....	215
7.3.40 MPCE (Modelling Platform for Collaborative Enterprises).....	216
7.3.41 Nehemiah.....	217
7.3.42 OPAL (Object, Process, Actor modelling Language).....	219
7.3.43 PIM4SOA (Platform-independent model for service-oriented architecture).....	220

7.3.44 PIM4SOA Execution and Simulation Platform.....	222
7.3.45 PIM4SOA Metamodel Feature for Eclipse.....	224
7.3.46 PIM4SOA to BPEL Transformation Feature for Eclipse.....	225
7.3.47 PIM4SOA to JACK model transformation.....	227
7.3.48 PIM4SOA to OWL-S model transformation.....	228
7.3.49 PIM4SOA to Web services model transformations.....	230
7.3.50 PIM4SOA to WSDL Transformation Feature for Eclipse.....	231
7.3.51 PIM4SOA to XSD Transformation Feature for Eclipse.....	233
7.3.52 POP* language interchange format and methodology.....	234
7.3.53 RSM (Rational Software Modeler).....	235
7.3.54 THEMIS.....	236
7.3.55 Semaphore - UML semantic mapping tool.....	238
7.3.56 SOAP feedback analyser.....	240
7.3.57 THEMIS.....	240
7.3.58 WSDL Analyzer.....	242
7.4 Demonstrators.....	244
7.4.1 Overview.....	244

## 1. Introduction

---

### 1.1. ATHENA Interoperability Framework (AIF)

---

#### 1.1.1. About

---

The operational ability to collaborate is a key success factor for networked enterprises, and interoperability is the target result of the enterprises involved in long established as well as ad-hoc or occasional forms of collaborations. The **ATHENA Interoperability Framework (AIF)** provides a compound framework and associated reference architecture for capturing the research elements and solutions to interoperability issues that address the problem in a holistic way by inter-relating relevant information from different perspectives of the enterprise.

#### 1.1.2. Contributions

---

The following people (listed in alphabetical order by surname) have contributed to the development of the ATHENA Interoperability Framework (AIF):

- Maria Anastasiou, INTRACOM, Greece
- Arne-Jørgen Berre, SINTEF ICT, Norway
- Brian Elvesæter, SINTEF ICT, Norway
- Nicolas Figay, EADS, France
- Oscar Garcia, AIDIMA, Spain
- Ulrike Greiner, SAP AG, Germany
- Claudia Guglielmina, TXT, Italy
- Svein G. Johnsen, SINTEF ICT, Norway
- Håvard D. Jørgensen, AKM AS, Norway
- Dag Karlsen, AKM AS, Norway
- Thomas Knothe, IPK, Germany
- Frank Lillehagen, AKM AS, Norway
- Sonia Lippe, SAP, Germany
- Jörg Müller, SIEMENS, Germany
- Lorenzo Pondrelli, FORMULA, Italy
- Igor Santos, ESI, Spain
- Giorgio Sobrito, CRF, Italy

#### 1.1.3. Acknowledgements

---

The work is partly funded by the European Commission through the ATHENA IP (Advanced Technologies for interoperability of Heterogeneous Enterprise Networks and their Applications Integrated Project) (IST-507849) (<http://www.athena-ip.org/>). The work does



not represent the view of the European Commission or the ATHENA consortium, and the authors are solely responsible for the content.

## 1.2. Background and motivation

---

### 1.2.1. What is interoperability?

---

System interoperability is a growing interest area, because of the continuously growing need of integration of new, legacy and evolving systems, in particular in the context of networked businesses and eGovernment. Enterprises today face many challenges related to lack of interoperability. Enterprises need to adapt more quickly to changes in the business and economic market and is required to become more responsive to customer needs. Although enterprises are heavily dependent on information communication technology (ICT) solutions in their day-to-day business operations, the solutions are often inflexible and difficult to adapt to meet the requirements of those changing enterprises [Truex, et al. 1999].

Enterprise applications and software systems need to be interoperable in order to achieve seamless business across organisational boundaries and thus realise virtual networked organisations. The current ICT solution space suffers badly from lack of interoperability. ICT systems are not able to sufficiently exchange information, and services offered by one system are not compatible with other systems. The effect of non-interoperability results in large budgets being spent on time-consuming system integration tasks.

The ATHENA project adopts the IEEE definition of interoperability as

*“the ability of two or more systems or components to exchange information and to use the information that has been exchanged”* [IEEE 1990].

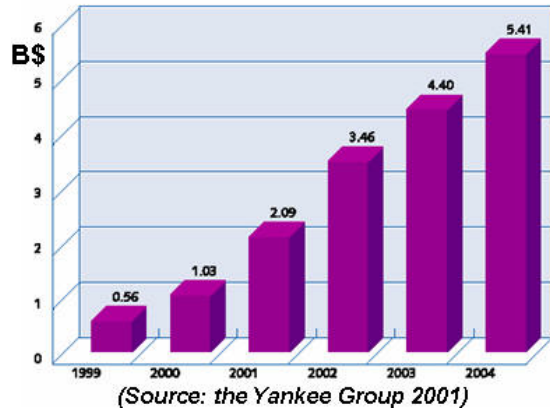
Please note that the term interoperability can be understood in a technical way or in a broad way, taking into account social, political and organisational factors. In the context of ATHENA the notion of interoperability is not limited to ICT systems, but also concerns the business processes and the business context of an enterprise. Therefore, ATHENA considers interoperation only meaningful, when all relevant levels of an enterprise are addressed. The diversity, heterogeneity, and autonomy of software components, application solutions, business processes, and the business context of an enterprise must be considered.

#### 1.2.1.1. Industrial need for interoperability

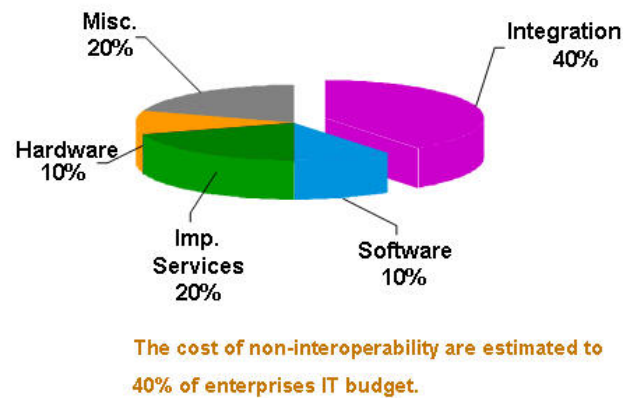
---

Lack of interoperability is today costing industry huge sums of money. An investigation performed by the Yankee Group in the US shows that some 40% of ICT project costs in most major manufacturing industries can be attributed to solve interoperability problems. The enterprise applications integration (EAI) market is projected to grow to some 7 bill US dollars in 2006 making it the biggest ICT market ahead of the enterprise architecture market.

### Application integration license revenue



### System implementation budget



#### 1.2.1.2. Economic reasoning for interoperability

Probably the most important issue related to interoperability is economical in its nature. It refers to the obtaining performing systemic behaviours within specific business environments even if the basic components and/or sub-systems have been developed independently and in different technological and business environments. Interoperability would make it possible to take advantage of scale and/or scope economics, in the development of the components, and to avoid, all the same, unbearable costs of development and/or integration each time the business or technological environments change.

Interoperability solutions have to address two big economics-related issues:

- looking to the future: to achieve the capability of obtaining fully integrated systemic functionalities, which often requires to design and develop the solutions from scratch together with the independent development and use of the components;
- looking to the past: to avoid jeopardising the huge investment made in existing systems, while accepting imperfect interoperability.

There are two accepted ways to achieve this:

- Defining standard applications and interface development environments, so that natively interoperable software systems can be specified, designed and implemented to respect the standards.
- Defining standard and open bridges (middleware) to consider organisational-semantic-technical aspects, so that natively non-interoperable software systems could achieve some predefined level of interoperability (i.e. SLA contracts).

In any case the goal is to operate across a boundary as if the boundary does not exist: interoperability happens at boundaries over which the information is exchanged; to agree where these boundaries should be is a basic step toward achieving it.

### 1.2.1.3. Islands of interoperability

---

Specific choices of where to put the boundaries may limit interoperability to specific classes of components and/or subsystems, and to certain specific business and/or normative contexts.

This does not mean that these components do not interoperate, it rather means that their interoperability is bounded within islands of interoperability, that are more or less large depending on the number of components and or environments, that comply with the specific design of their interfaces.

The dimensions of the interoperability islands usually depend on economic balances that are conditioned by normative and/or business contexts.

In conclusion, we can say that interoperability is an enabler/facilitator for seamless business enterprise networks, but it is neither necessary, nor sufficient. A mix of Technical and Economic reasoning could indeed suggest not achieving full enterprise interoperability in certain business contexts, where interoperability costs would exceed interoperability benefits (necessity), while "soft issues" like legal, social, contractual, political, psychological, cultural ones could definitively hinder ebusiness also in presence of a full, potential enterprises interoperability.

### 1.2.2. ATHENA's approach to interoperability

---

ATHENA – Advanced Technologies for interoperability of Heterogeneous Enterprise Networks and their Applications - is an Integrated Project sponsored by the European Commission in support of the Strategic Objective “Networked businesses and government” set out in the IST 2003-2004 Workprogramme of FP6. Building upon an ambitious Vision Statement “By 2010, enterprises will be able to seamlessly interoperate with others”, ATHENA aims to make a major contribution to interoperability by identifying and meeting a set of inter-related business, scientific & technical, and strategic objectives.

The ATHENA programme of work is defined for producing results that span the full spectrum of interoperability from technology components to applications and services, from research & development to demonstration & testing, and from training to evaluation of technologies for societal impact. In ATHENA, Research and Development is executed in close synergy and collaboration with Community Building, for ensuring that solutions to multi-disciplinary research challenges are of optimal industrial relevance leading to broad uptake by the end user.

The ATHENA consortium currently comprises 19 leading organisations in research, academia, industry and other stakeholder communities including SMEs, working collaboratively in pursuit of a common set of objectives in interoperability.

ATHENA is committed to creating a long term impact for advancing interoperability which is mainstream, inclusive and has critical mass. To this end, ATHENA is initiating an open, neutral and independent Enterprise Interoperability Centre (EIC) to which all stakeholders, in

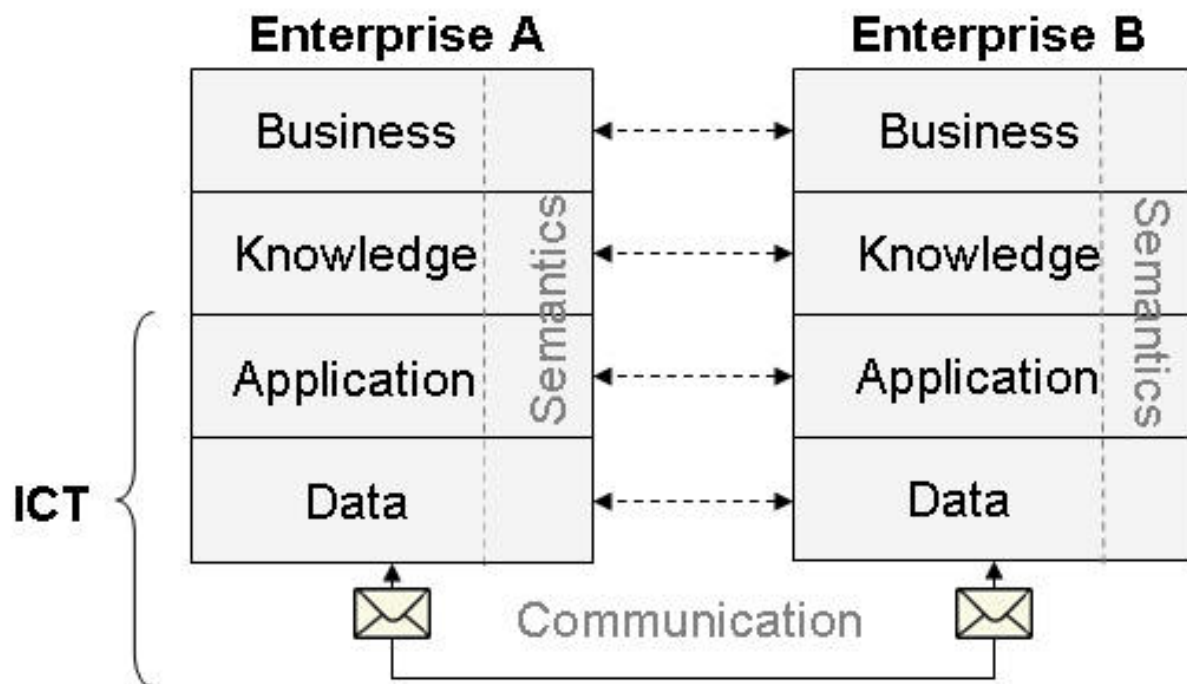
both private and public sectors, are invited to participate.

#### 1.2.2.1. Holistic approach to interoperability

---

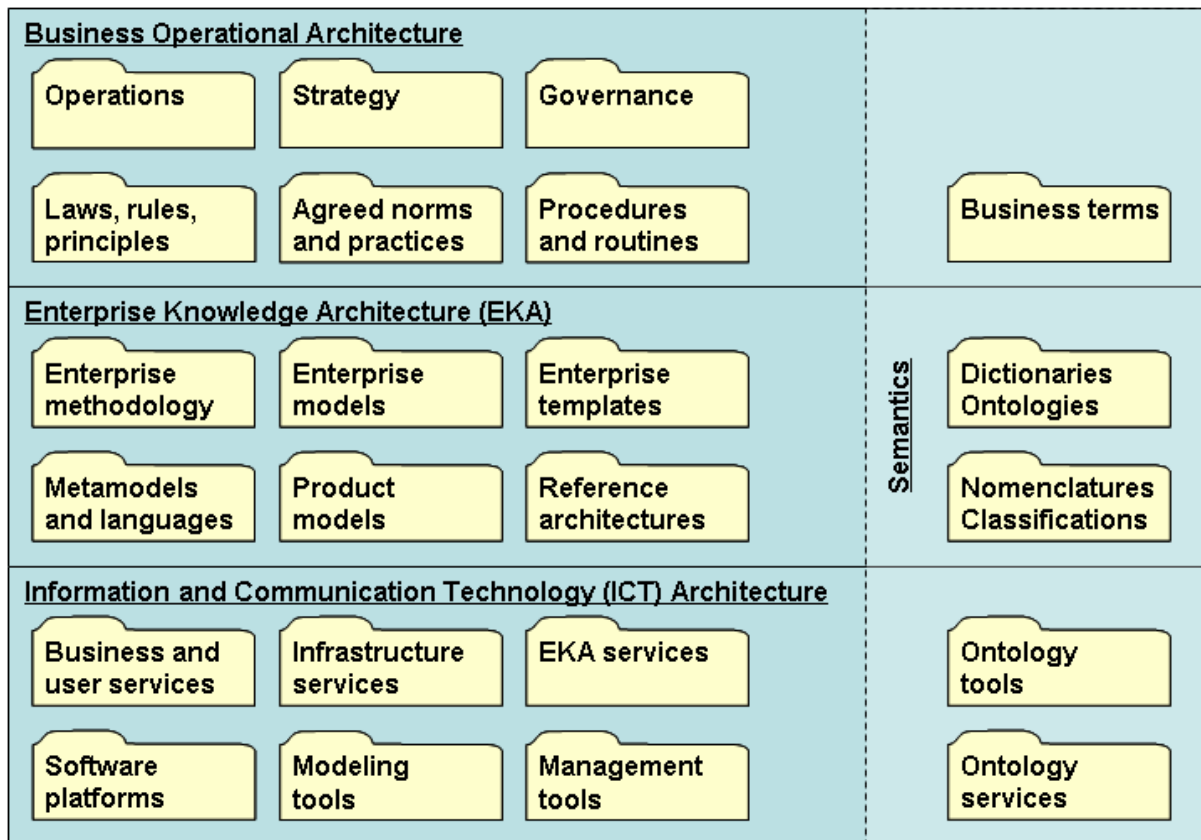
ATHENA adopts a holistic perspective on interoperability in order to achieve real, meaningful interoperation between enterprises. ATHENA builds upon the FP5 thematic network **IDEAS** (Interoperability Development for Enterprise Applications and Software, IST-2001-37368). The IDEAS network identified the need for a structured approach to collect, identify and represent the current state of the art, vision statements, and research challenges. It defined a framework for capturing and inter-relating this information from many perspectives called the **IDEAS Interoperability Framework**.

- The **business layer** is located at the top of the framework. In this layer, all issues related to the organisation and the operations of an enterprise are addressed. Amongst others, they include the way an enterprise is organised, how it operates to produce value, how it takes decisions, how it manages its relationships (both internally with its personnel and externally with partners, customers, and suppliers).
- The **knowledge layer** deals with acquiring a deep and wide knowledge of the enterprise. This includes knowledge of internal aspects such as products, the way the administration operates and controls, how the personnel is managed, and so on, but also of external aspects such as partners and suppliers, laws and regulations, legal obligations, and relationships with public institutions.
- The **ICT systems layer** focuses on the ICT solutions that allow an enterprise to operate, make decisions, exchange information within and outside its boundaries, and so on.
- The **semantic** dimension cuts across the business, knowledge and ICT layers. It is concerned with capturing and representing the actual meaning of concepts and thus promoting understanding.



To achieve meaningful interoperability between enterprises, interoperability must be achieved on all layers:

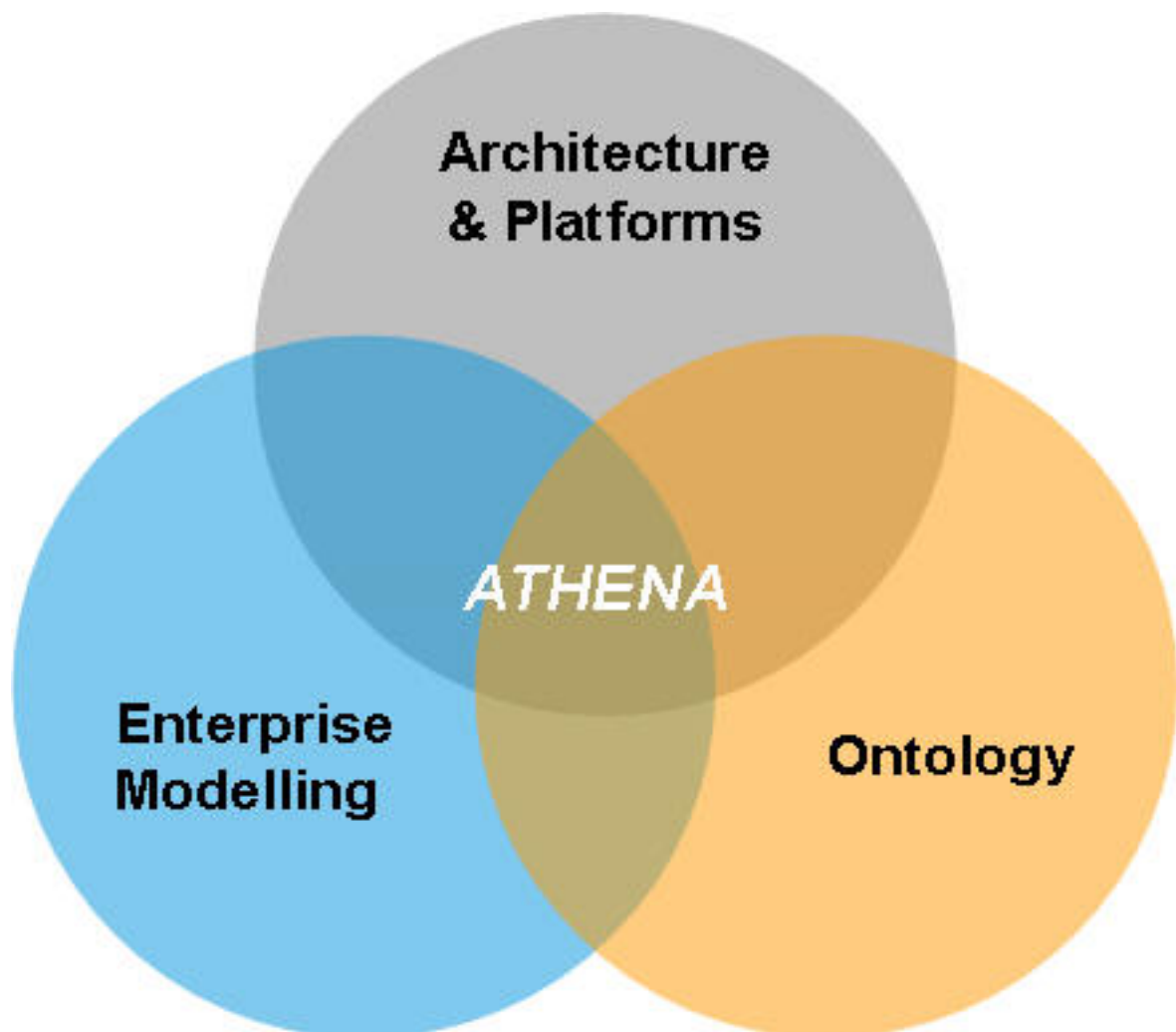
- Interoperability at **business level** should be seen as the organisational and operational ability of an enterprise to factually cooperate with other, external organisations, whether these organisations are enterprises or public institutions.
- Interoperability at **knowledge level** should be seen as the compatibility of the skills, competencies, and knowledge assets of an enterprise with those of other, external organisations.
- Interoperability at **ICT systems level** should be seen as the ability of an enterprise's ICT systems to cooperate with those of other, external organisations.
- To overcome the **semantic barrier**, which emerges from different interpretations of syntactic descriptions, precise, computer processable meaning must be associated with each concept. It has to be ensured that semantics are exchangeable and based on a common understanding to be indeed a means to enhance interoperability.



### 1.2.2.2. Multidisciplinary approach to interoperability

The originality of the ATHENA project is to take a multidisciplinary approach by merging three research areas supporting the development of interoperability of enterprise applications and software.

- **Architecture & Platforms:** to provide implementation frameworks,
- **Enterprise Modelling:** to define interoperability requirements and to support solution implementation,
- **Ontology:** to identify interoperability semantics in the enterprise.



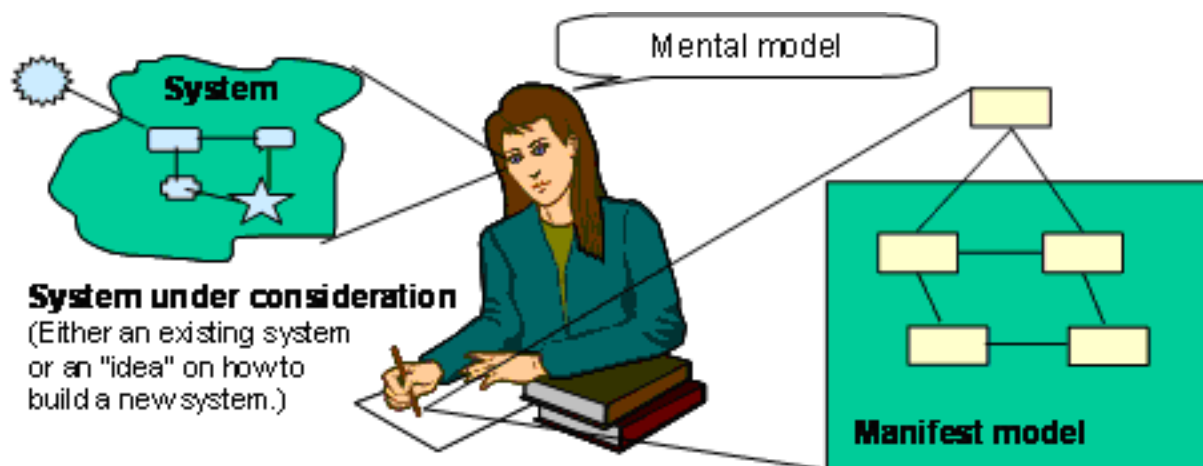
The multidisciplinary approach also represents some challenges. While ATHENA advocates a holistic and multidisciplinary approach, each of the disciplines may have different approaches that may be overlapping or even competing. We need to describe the various alternative approaches and provide guidelines for which to select depending on the context.

### **1.2.2.3. Model-driven approach to interoperability**

When creating a model, one must have a clear understanding of what the model is meant to and not meant to illustrate. Many aspects are hidden in a model; after all, one of the main purposes of models is to abstract from irrelevant details. One can, however, abstract along different dimensions, that is, one can choose to leave out different types of information of a system depending on the purpose of the model. For instance, one can choose to depict processes and activities of the system under consideration or one can choose to depict the information this system contains and how the different information elements are related.



We use *model* to refer to a specification of an entity in the real world. To be a model, this specification needs to have a commonly agreed semantics and a well-defined syntax. In other words, models are specifications that have a certain format (typically in a modelling language) and where all symbols in the language have a predefined and commonly understood interpretation. The figure below shows the process of system modelling; one studies a real world phenomenon which is the system under consideration (either an existing system or an “idea” on how to build a new system), and creates a model using a chosen modelling language.



A common characteristic of the ATHENA solutions is the fact that they are model-driven. The universe of discourse is the collaborative enterprise and the ICT systems used by the enterprises participating in the collaboration. The ATHENA solutions focus on modelling the interactions and information exchanges that occur during such collaborations, both on a business requirements level and a technical solution level.

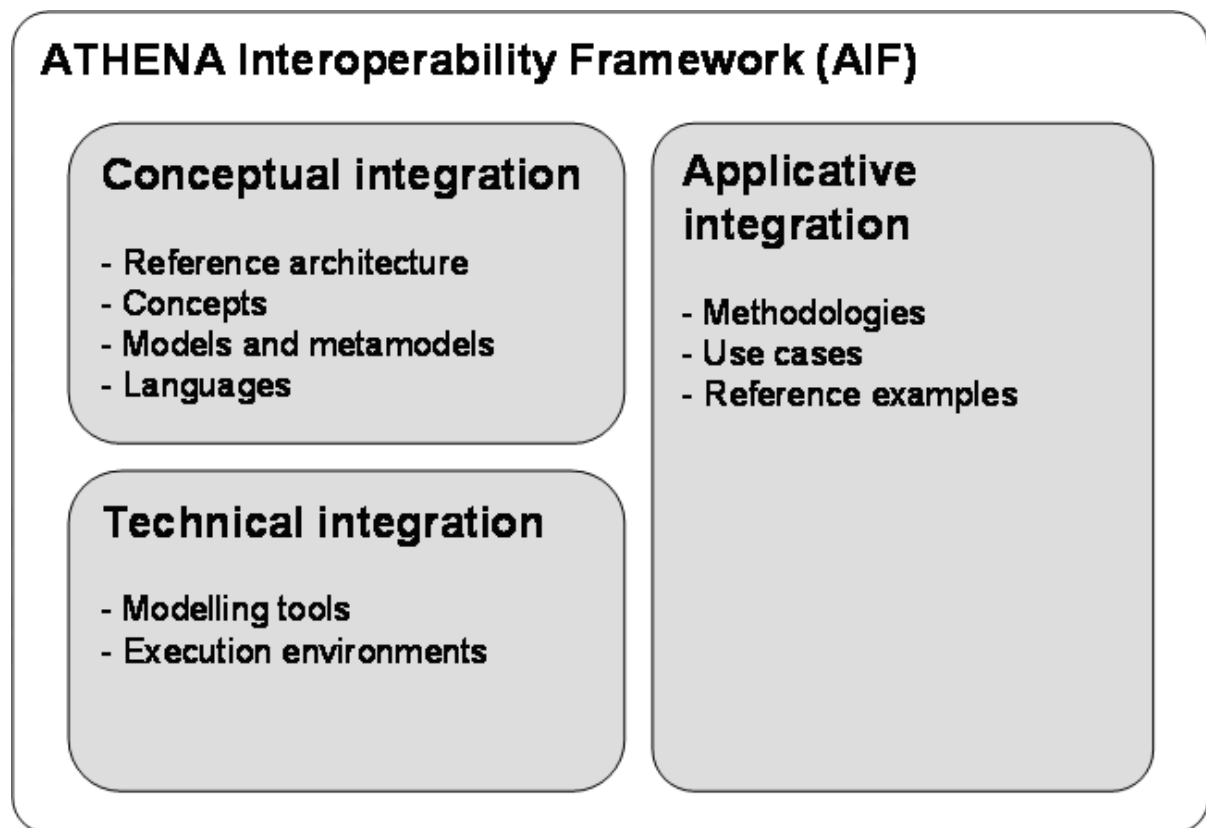
### 1.3. Overview of the AIF

#### 1.3.1. What is an interoperability framework?

A *framework* is a structure for supporting or enclosing something else, especially a skeletal support used as the basis for something being constructed. An *interoperability framework* provides a set of assumptions, concepts, values and practices that constitutes a way of viewing and addressing interoperability issues. The **ATHENA Interoperability Framework (AIF)** provides a compound framework and associated reference architecture for capturing the research elements and solutions to interoperability issues that address the problem in a holistic way by inter-relating relevant information from different perspectives of the enterprise.

#### 1.3.2. Structure of the AIF





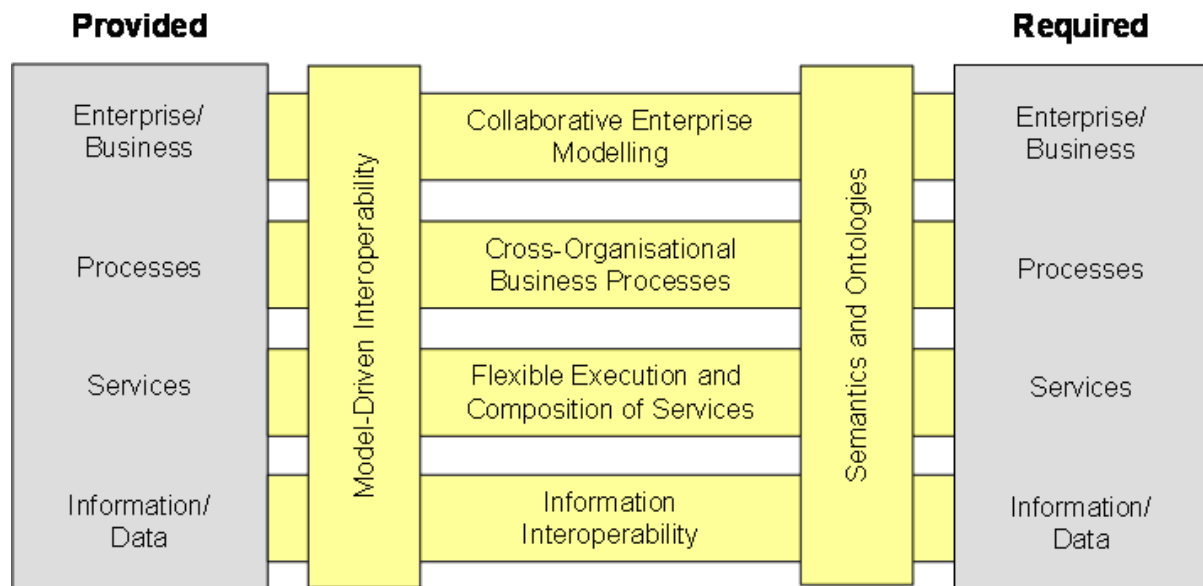
The **ATHENA Interoperability Framework (AIF)** is structured into three parts:

1. **Conceptual integration** which focuses on concepts, metamodels, languages and model relationships. The framework defines an *interoperability reference architecture* that provides us with a foundation for systemising various aspects of interoperability.
2. **Applicative integration** which focuses on methodologies, standards and domain models. The framework defines a *methodology framework* that provides us with guidelines, principles and patterns that can be used to solve interoperability issues.
3. **Technical integration** which focuses on the software development and execution environments. The framework defines a *technical architecture* that provides development tools and execution platforms for integrating processes, services and information.

### 1.3.3. Interoperability reference architecture

The *interoperability reference architecture* relates the solution approaches coming from the three different research areas of ATHENA, namely enterprise modelling, architectures and platforms, and ontology. The figure below illustrates the reference architecture that focuses on the *provided* and *required* artefacts of two collaborating enterprises. Interoperations can take place at the various levels (*enterprise/business, process, service and information/data*). For each of these levels we prescribe a *model-driven interoperability* approach where models

are used to formalise and exchange the relevant provided and required artefacts that must be aligned and made compatible through negotiations and agreements.



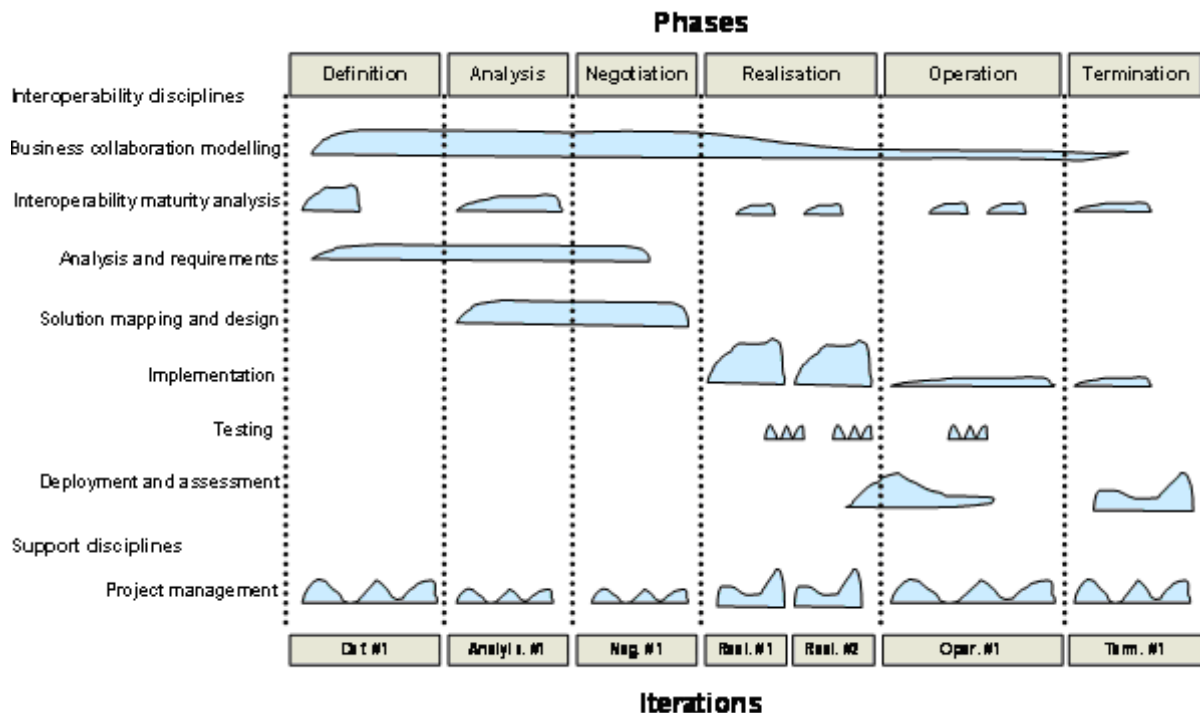
*Collaborative enterprise modelling* concerns the exchange and alignment of knowledge models for describing the processes, organisations, products and systems in the collaboration context. Modelling of *cross-organisational business processes* focuses on defining process views that describes the interactions between two or more business entities. *Flexible execution and composition of services* is concerned with identifying, composing and executing various applications. *Information interoperability* is related to management, exchange and processing of different documents, messages and other information structures.

To overcome the semantic barriers which emerge from different interpretations of syntactic descriptions, precise, computer processable meaning must be associated with the models expressed on the different levels. It has to be ensured that *semantics* are exchangeable and based on common understanding in order to enhance interoperability. This can be achieved using *ontologies* and an annotation formalism for defining meaning in the exchanged models.

### 1.3.4. Interoperability methodology framework

The AIF also provides an associated methodological framework, the **ATHENA Interoperability Methodology (AIM)**, which describes the approach towards interoperability from the decision to evaluate collaboration until solution maintenance, and the reference guidelines for the adoption of the reference architecture. In the figure below the AIM is rendered from an overall perspective, showing the essential structure of *phases* and *disciplines*. The phases of an interoperability project life-cycle are represented by the columns. The rows in the figure outline the set of principles that characterise the mature approach for the definition, creation, operation and termination of an interoperability project.

An *interoperability discipline* is a group of activities within a specific interoperability field which are logically grouped together. Within each discipline, the AIM recommends sets of activities to be performed in the different phases of the interoperability project.



### 1.3.5. Interoperability profiles

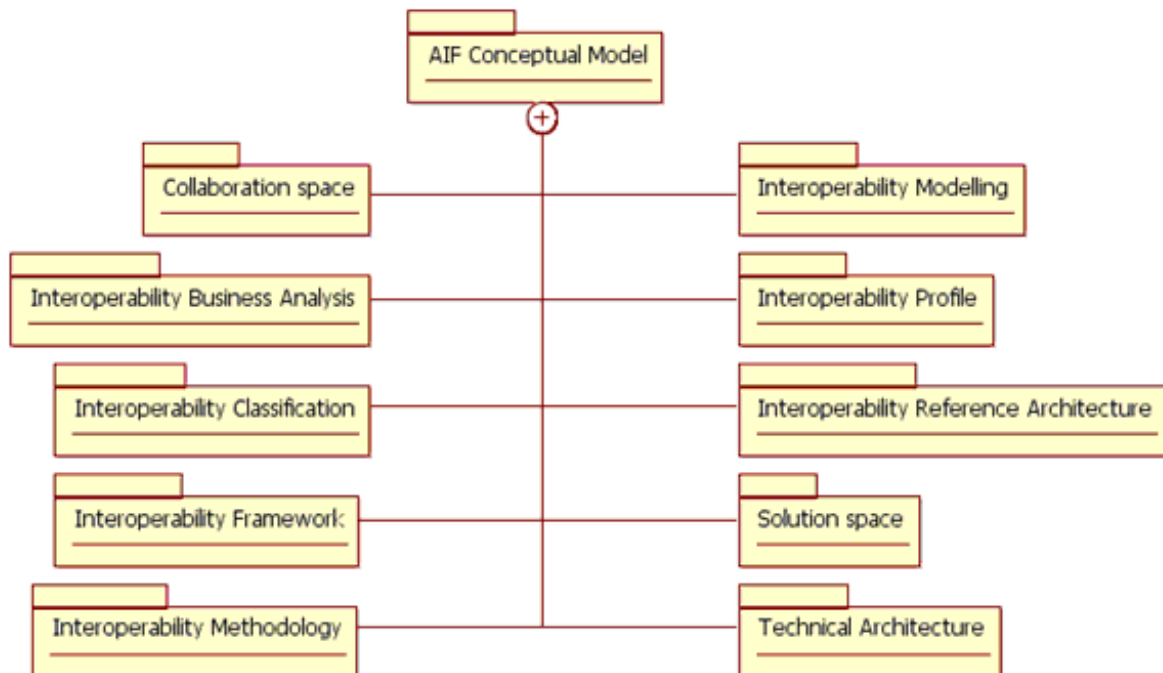
Profiles are a way to structure the complex relationships between the individual research results, scenario and the ATHENA Interoperability Framework. Based on the domain and industry sector specific standards, the business needs and the selected interoperability scenarios of an enterprise, an *interoperability profile* can be defined to ease the interoperability efforts for the enterprise by being valuable and applicable in the described scenarios. An *interoperability profile* defines a set of results or specifications that work together. It consists of interoperability guidelines, specifications, and integrated and configured solutions from the conceptual, applicative and technical parts of the AIF.

The concept of an *interoperability profile* was initially defined when preparing the description of work of the ATHENA project [ATHENA 2003], on basis of a categorisation per application domains (initially *Supply Chain Management*, *Product Portfolio Management*, *Collaborative Product Development*, and *e-Procurement*) and industry sectors (initially *Automotive*, *Aerospace*, *Furniture* and *Telecommunication*). The initial aim was to create four **ATHENA Interoperability Profiles (AIPs)** for the selected scenarios covering the application domains.

## 1.4. Specification of the AIF

### 1.4.1. AIF conceptual model

The ATHENA Interoperability Framework (AIF) provides a compound framework and associated reference architecture for capturing the research elements and solutions to interoperability issues that address the problem in a holistic way by inter-relating relevant information from different perspectives of the enterprise. The specification of the AIF has been formalised using a model-driven approach. This has resulted in the development of an **AIF conceptual model**. The conceptual model is structured into separate model packages covering specific concept domains that we see relevant for addressing interoperability. Each package contains descriptions of the concepts and their relationships (both within and across the concept domains).



A short description of the different concept domains are given below:

- Collaboration space: This domain defines the concepts for a collaboration space which is an environment that provides an infrastructure to support collaboration between members of a virtual enterprise network.
- Interoperability business analysis: This domain defines the concepts for interoperability business analysis which focuses on capturing business needs and interoperability issues and finding appropriate technical solutions.
- Interoperability classification: This domain defines the concepts for classification of

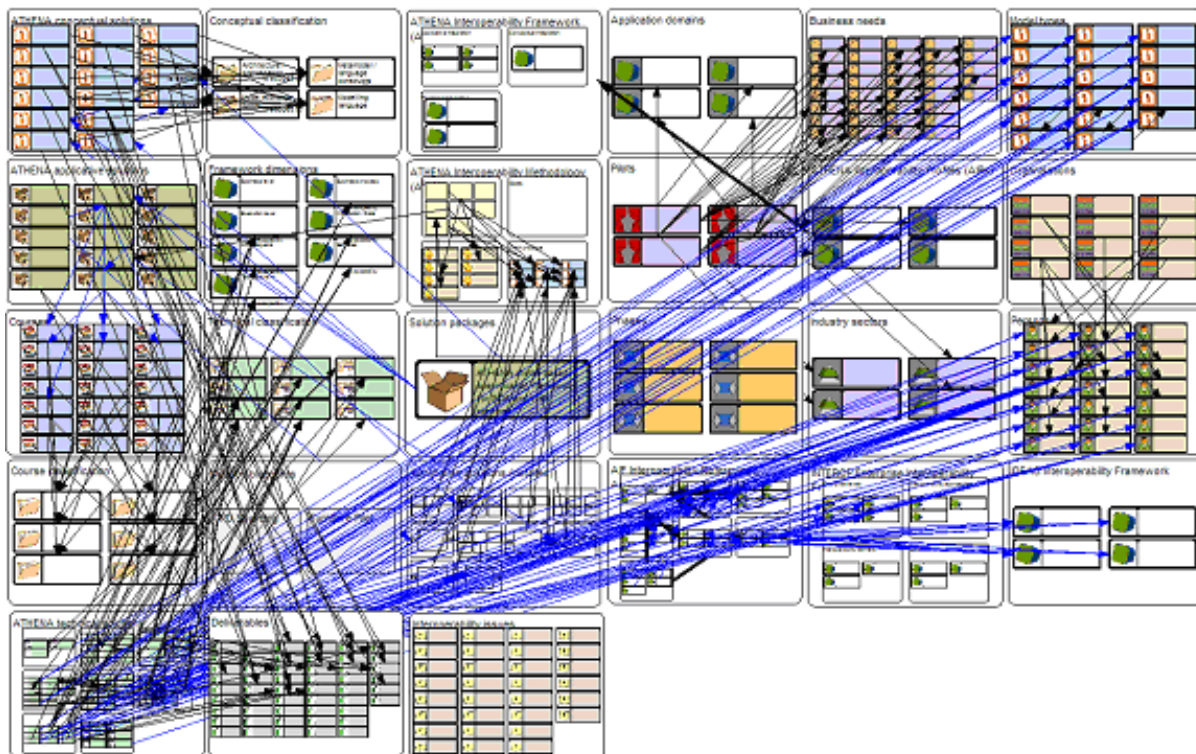
interoperability problems and solutions.

- Interoperability framework: This domain defines the concepts for an interoperability framework.
- Interoperability methodology: This domain defines the concepts for an interoperability methodology.
- Interoperability modelling: This domain defines the concepts for interoperability modelling that focuses on capturing essential information relevant to enterprise collaboration as formalised models.
- Interoperability profile: This domain defines the concepts for an interoperability profile. An interoperability profile defines a set of results or specifications that work together. It consists of interoperability guidelines, specifications, and integrated and configured solutions from the conceptual, applicative and technical parts of the AIF.
- Interoperability reference architecture: This domain defines the concepts for an interoperability reference architecture which relates the different approaches to interoperability coming from the different research areas of ATHENA.
- Solution space: This domain defines the concepts for a solution space which defines a structuring of solutions for interoperability problems.
- Technical architecture: This domain defines the concepts for a technical architecture which provides a blueprint for implementing your technical interoperability ICT infrastructure.

#### **1.4.2. AIF knowledge model**

---

Alongside the conceptual model we have developed a partial instance model [ATHENA 2007b] that describes essential artefacts of the ATHENA universe (solutions, methods, deliverables, etc.) and how these are related. The AIF knowledge model has been developed using the enterprise architecture modelling tool Metis [Trouw Technologies] and is shown in Figure 81 below. The model serves two purposes. Firstly it has helped us to reason about the concepts, structure and relationships in the specification of the AIF, and secondly it provides a “proof of concept” in the validation of the AIF.



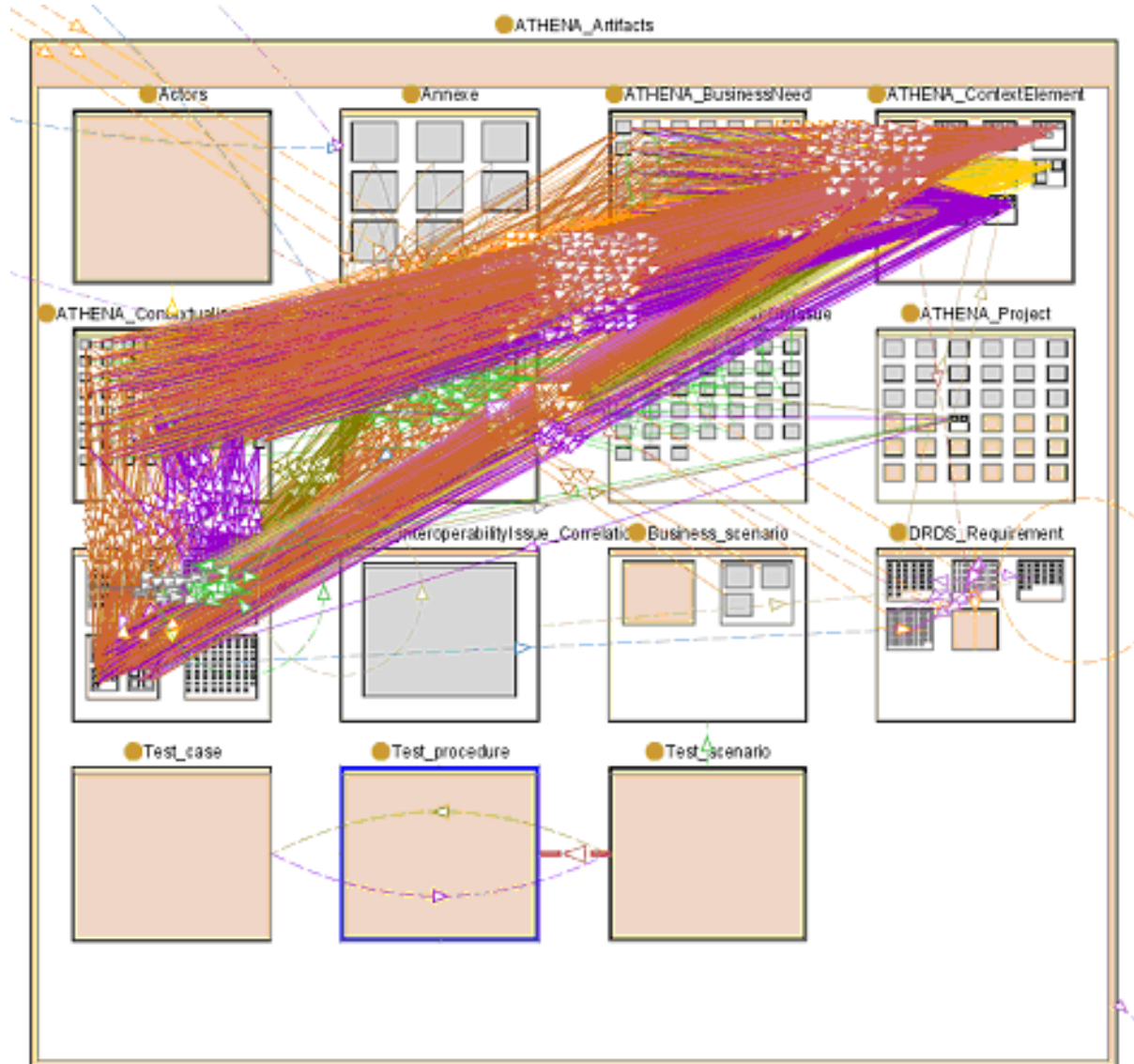
### 1.4.3. ATHENA knowledge base

Whereas the AIF knowledge model was developed primarily from a top-down perspective, a parallel modelling effort in the requirements and piloting activities of ATHENA developed the ATHENA knowledge base from a bottom-up perspective. The knowledge base establishes relationships between requirements, interoperability issues, generic solutions and solutions using the different models produced by ATHENA projects (e.g. AIF and BIF). It was decided that the different model views of ATHENA needed to be implemented as a federation of models that would be available as a Web resource to support the pilot users. The Ontology Web Language (OWL) is a technology that was designed to provide a common way to process the content of Web information and was chosen as the vehicle for implementing the knowledge base. The selected OWL editor was Protégé [Stanford Medical Informatics].

Once the knowledge model was created it was also important to provide querying and filtering mechanisms for the users in order to exploit the knowledge base. Protégé comes with a built-in visualisation mechanism to graphically visualize and navigate the knowledge model. Figure 82 below shows a visual representation of the knowledge model. As can be seen it resembles the AIF knowledge model described above. The difference is mainly that the AIF knowledge model is wider in scope but only contains content examples to validate the conceptual model, whereas the ATHENA knowledge base is more focused (in particular



relating requirements, interoperability issues to solutions) and is fully populated for this purpose.



## 1.5. References

### 1.5.1. Bibliography

[Apache.org] Apache.org, "The Apache Forrest Project". <http://forrest.apache.org/>

[ATHENA 2003] ATHENA, "Annex I - Description of Work", ATHENA IP, 12. December

2003.

[ATHENA 2006a] ATHENA, "ATHENA Model-Driven Interoperability (MDI) Framework", ATHENA IP, 2006a. <http://modelbased.net/mdi/>

[ATHENA 2006b] ATHENA, "ATHENA Service-Oriented Interoperability (SOI) Framework", ATHENA IP, 2006b. <http://modelbased.net/soi/>

[ATHENA 2007a] ATHENA, "ATHENA Interoperability Framework (AIF) Website", ATHENA IP, 2007a. [http://athena.akmodeling.com/Team/Repository/Projects/Project\\_207/Upload/website/build/site/index.html](http://athena.akmodeling.com/Team/Repository/Projects/Project_207/Upload/website/build/site/index.html)

[ATHENA 2007b] ATHENA, "ATHENA Interoperability Framework (AIF) Model", ATHENA IP, 2007b. [http://athena.akmodeling.com/Team/Repository/Projects/Project\\_207/Upload/model/index.html](http://athena.akmodeling.com/Team/Repository/Projects/Project_207/Upload/model/index.html)

[ATHENA A1 2004a] ATHENA A1, "D.A1.5.1: MCPE Specification, Version 1.0", ATHENA IP, Deliverable D.A1.5.1, September 2004a. [http://athena.akmodeling.com/Team/Repository/Projects/Project\\_216/Upload/FileSharing/Del\\_1st6Ms/04092](http://athena.akmodeling.com/Team/Repository/Projects/Project_216/Upload/FileSharing/Del_1st6Ms/04092)

[ATHENA A1 2004b] ATHENA A1, "D.A1.1.1: First Version of State of the Art in Enterprise Modelling Techniques and Technologies to Support Enterprise Interoperability, Version 1.2", ATHENA IP, Deliverable D.A1.1.1, July 2004b. [http://athena.akmodeling.com/Team/Repository/Projects/Project\\_216/Upload/FileSharing/Del\\_1st6Ms/0408](http://athena.akmodeling.com/Team/Repository/Projects/Project_216/Upload/FileSharing/Del_1st6Ms/0408)

[ATHENA A1 2005a] ATHENA A1, "D.A1.3.1: Report on Methodology description and guidelines definition, Version 1.0", ATHENA IP, Deliverable D.A1.3.1, March 2005a. [http://athena.akmodeling.com/Team/Repository/Projects/Project\\_216/Upload/FileSharing/Del\\_2nd6Ms/0503](http://athena.akmodeling.com/Team/Repository/Projects/Project_216/Upload/FileSharing/Del_2nd6Ms/0503)

[ATHENA A1 2005b] ATHENA A1, "D.A1.2.1: Model and Report – First Set of Requirements, Version 1.0", ATHENA IP, Deliverable D.A1.2.1, February 2005b. [http://athena.akmodeling.com/Team/Repository/Projects/Project\\_216/Upload/FileSharing/Del\\_2nd6Ms/0503](http://athena.akmodeling.com/Team/Repository/Projects/Project_216/Upload/FileSharing/Del_2nd6Ms/0503)

[ATHENA A1 2005c] ATHENA A1, "D.A1.4.1: Framework for the Establishment and Management Methodology, Version 1.0", ATHENA IP, Deliverable D.A1.4.1, February 2005c. [http://athena.akmodeling.com/Team/Repository/Projects/Project\\_216/Upload/FileSharing/Del\\_2nd6Ms/0503](http://athena.akmodeling.com/Team/Repository/Projects/Project_216/Upload/FileSharing/Del_2nd6Ms/0503)

[ATHENA A1 2005d] ATHENA A1, "D.A1.5.2: Collaborative Modelling Platform First Prototype, Version 1.0", ATHENA IP, Deliverable D.A1.5.2, July 2005d. [http://athena.akmodeling.com/Team/Repository/Projects/Project\\_216/Upload/FileSharing/Del\\_3rd6Ms/0509](http://athena.akmodeling.com/Team/Repository/Projects/Project_216/Upload/FileSharing/Del_3rd6Ms/0509)

[ATHENA A1 2006] ATHENA A1, "D.A1.6.1: Evaluation and Benefit Assessment, Version 1.0", ATHENA IP, Deliverable D.A1.6.1, February 2006. [http://athena.akmodeling.com/Team/Repository/Projects/Project\\_216/Upload/FileSharing/Del\\_4th6Ms/0602](http://athena.akmodeling.com/Team/Repository/Projects/Project_216/Upload/FileSharing/Del_4th6Ms/0602)

[ATHENA A2 2005a] ATHENA A2, "D.A2.1: Cross-Organisational Business Process requirements and the State of the Art in Research, Technology and Standards, Version 1.0", ATHENA IP, Deliverable D.A2.1, November 2005a.



[http://athena.akmodeling.com/Team/Repository/Projects/Project\\_216/Upload/FileSharing/Del\\_4th6Ms/0602](http://athena.akmodeling.com/Team/Repository/Projects/Project_216/Upload/FileSharing/Del_4th6Ms/0602)

[ATHENA A2 2005b] ATHENA A2, "D.A2.2: Specification of a Cross-Organisational Business Process Model, Version 1.0", ATHENA IP, Deliverable D.A2.2, June 2005b.

[http://athena.akmodeling.com/Team/Repository/Projects/Project\\_216/Upload/FileSharing/Del\\_3rd6Ms/0507](http://athena.akmodeling.com/Team/Repository/Projects/Project_216/Upload/FileSharing/Del_3rd6Ms/0507)

[ATHENA A2 2005c] ATHENA A2, "D.A2.3: Architecture for Enactment and Integration of Cross-Organisational Business Processes, Version 1.0", ATHENA IP, Deliverable D.A2.3, August 2005c.

[http://athena.akmodeling.com/Team/Repository/Projects/Project\\_216/Upload/FileSharing/Del\\_3rd6Ms/0509](http://athena.akmodeling.com/Team/Repository/Projects/Project_216/Upload/FileSharing/Del_3rd6Ms/0509)

[ATHENA A2 2005d] ATHENA A2, "D.A2.4: Architecture for Enactment and Integration of Cross-Organizational Business Processes, Version 1.0", ATHENA IP, Deliverable D.A2.4, November 2005d.

[http://athena.akmodeling.com/Team/Repository/Projects/Project\\_216/Upload/FileSharing/Del\\_4th6Ms/0602](http://athena.akmodeling.com/Team/Repository/Projects/Project_216/Upload/FileSharing/Del_4th6Ms/0602)

[ATHENA A2 2006] ATHENA A2, "D.A2.5: Validation of Research Results, Version 1.0", ATHENA IP, Deliverable D.A2.5, January 2006.

[http://athena.akmodeling.com/Team/Repository/Projects/Project\\_216/Upload/FileSharing/Del\\_4th6Ms/0602](http://athena.akmodeling.com/Team/Repository/Projects/Project_216/Upload/FileSharing/Del_4th6Ms/0602)

[ATHENA A3 2005] ATHENA A3, "D.A3.1: SoA on Ontologies and the Ontology Authoring and Management System, with Ontology Modelling Language, Version 1.0", ATHENA IP, Deliverable D.A3.1, March 2005.

[http://athena.akmodeling.com/Team/Repository/Projects/Project\\_216/Upload/FileSharing/Del\\_2nd6Ms/0504](http://athena.akmodeling.com/Team/Repository/Projects/Project_216/Upload/FileSharing/Del_2nd6Ms/0504)

[ATHENA A3 2006a] ATHENA A3, "D.A3.2: Updated version of the Ontology Authoring and Management System with semantic search functions, Version 1.0", ATHENA IP, Deliverable D.A3.2, February 2006a.

[http://athena.akmodeling.com/Team/Repository/Projects/Project\\_216/Upload/FileSharing/Del\\_4th6Ms/0602](http://athena.akmodeling.com/Team/Repository/Projects/Project_216/Upload/FileSharing/Del_4th6Ms/0602)

[ATHENA A3 2006b] ATHENA A3, "D.A3.3: Semantic Annotation language and tool for Information and Business Processes, Version 1.0", ATHENA IP, Deliverable D.A3.3, February 2006b.

[http://athena.akmodeling.com/Team/Repository/Projects/Project\\_216/Upload/FileSharing/Del\\_4th6Ms/0602](http://athena.akmodeling.com/Team/Repository/Projects/Project_216/Upload/FileSharing/Del_4th6Ms/0602)

[ATHENA A3 2006c] ATHENA A3, "D.A3.4: A system for reconciliation rules specification, storage and management, Version 1.0", ATHENA IP, Deliverable D.A3.4, February 2006c.

[http://athena.akmodeling.com/Team/Repository/Projects/Project\\_216/Upload/FileSharing/Del\\_4th6Ms/0602](http://athena.akmodeling.com/Team/Repository/Projects/Project_216/Upload/FileSharing/Del_4th6Ms/0602)

[ATHENA A3 2006d] ATHENA A3, "D.A3.5: A reconciliation and mediation engine, capable to efficiently process semantic mediation and reconciliation rules, Version 1.0", ATHENA IP, Deliverable D.A3.5, January 2006d.

[http://athena.akmodeling.com/Team/Repository/Projects/Project\\_216/Upload/FileSharing/Del\\_4th6Ms/0602](http://athena.akmodeling.com/Team/Repository/Projects/Project_216/Upload/FileSharing/Del_4th6Ms/0602)

[ATHENA A4 2005] ATHENA A4, "D.A4.1: Requirements for Interoperability Framework, product-based and process-based Interoperability Infrastructures, Interoperability Life-cycle Services, Version 1.0", ATHENA IP, Deliverable D.A4.1, March 2005.

[http://athena.akmodeling.com/Team/Repository/Projects/Project\\_216/Upload/FileSharing/Del\\_2nd6Ms/0503](http://athena.akmodeling.com/Team/Repository/Projects/Project_216/Upload/FileSharing/Del_2nd6Ms/0503)

[ATHENA A4 2006] ATHENA A4, "WD.A4.2: Specification of Interoperability Framework and Profiles, Guidelines and Best Practices", ATHENA IP, Working document WD.A4.2, 2006.

[ATHENA A4 2007a] ATHENA A4, "D.A4.3: Product-based Interoperability Infrastructure and Profiles", ATHENA IP, Deliverable D.A4.3, 2007a.

[ATHENA A4 2007b] ATHENA A4, "D.A4.4: Process-based Interoperability Infrastructure and Profiles", ATHENA IP, Deliverable D.A4.4, 2007b.

[ATHENA A4 2007c] ATHENA A4, "D.A4.5: Interoperability life-cycle Services", ATHENA IP, Deliverable D.A4.5, 2007c.

[ATHENA A4 2007d] ATHENA A4, "D.A4.6: Validation of research results - product-based and process-based interoperability infrastructures, Interoperability life-cycle Services", ATHENA IP, Deliverable D.A4.6, 2007d.

[ATHENA A5 2005a] ATHENA A5, "D.A5.3: Architecture of SOA platforms, Version 1.0", ATHENA IP, Deliverable D.A5.3, November 2005a.

[http://athena.akmodeling.com/Team/Repository/Projects/Project\\_216/Upload/FileSharing/Del\\_4th6Ms/0602](http://athena.akmodeling.com/Team/Repository/Projects/Project_216/Upload/FileSharing/Del_4th6Ms/0602)

[ATHENA A5 2005b] ATHENA A5, "D.A5.4: Execution Framework(s) for Planned and Customisable Service-Oriented Architectures, Version 1.0", ATHENA IP, Deliverable D.A5.4, November 2005b.

[http://athena.akmodeling.com/Team/Repository/Projects/Project\\_216/Upload/FileSharing/Del\\_4th6Ms/0602](http://athena.akmodeling.com/Team/Repository/Projects/Project_216/Upload/FileSharing/Del_4th6Ms/0602)

[ATHENA A5 2005c] ATHENA A5, "D.A5.2: Model and Specification of Service Descriptions and Usage as well as Advanced Concepts, Version 1.1", ATHENA IP, Deliverable D.A5.2, August 2005c.

[http://athena.akmodeling.com/Team/Repository/Projects/Project\\_216/Upload/FileSharing/Del\\_3rd6Ms/0509](http://athena.akmodeling.com/Team/Repository/Projects/Project_216/Upload/FileSharing/Del_3rd6Ms/0509)

[ATHENA A5 2005d] ATHENA A5, "D.A5.1: Perspectives on Service-Oriented Architectures and their application in environments that require solutions to be planned and customisable, Version 1.0", ATHENA IP, Deliverable D.A5.1, November 2005d.

[http://athena.akmodeling.com/Team/Repository/Projects/Project\\_216/Upload/FileSharing/Del\\_4th6Ms/0602](http://athena.akmodeling.com/Team/Repository/Projects/Project_216/Upload/FileSharing/Del_4th6Ms/0602)

[ATHENA A5 2006] ATHENA A5, "D.A5.5: Validation of Research Results, Version 1.0", ATHENA IP, Deliverable D.A5.5, February 2006.

[http://athena.akmodeling.com/Team/Repository/Projects/Project\\_216/Upload/FileSharing/Del\\_4th6Ms/0602](http://athena.akmodeling.com/Team/Repository/Projects/Project_216/Upload/FileSharing/Del_4th6Ms/0602)

[ATHENA A6 2005] ATHENA A6, "D.A6.1: Specification of a Basic Architecture Reference Model, Version 1.0", ATHENA IP, Deliverable D.A6.1, 2005.

[http://athena.akmodeling.com/Team/Repository/Projects/Project\\_216/Upload/FileSharing/Del\\_2nd6Ms/0503](http://athena.akmodeling.com/Team/Repository/Projects/Project_216/Upload/FileSharing/Del_2nd6Ms/0503)

[ATHENA A6 2006a] ATHENA A6, "D.A6.3: Model-driven and Adaptable Interoperability Framework, Version 1.0", ATHENA IP, Deliverable D.A6.3, February 2006a.

[http://athena.akmodeling.com/Team/Repository/Projects/Project\\_216/Upload/FileSharing/Del\\_4th6Ms/0602](http://athena.akmodeling.com/Team/Repository/Projects/Project_216/Upload/FileSharing/Del_4th6Ms/0602)

[ATHENA A6 2006b] ATHENA A6, "D.A6.4: Model-driven and Adaptable Interoperability Infrastructure, Version 1.0", ATHENA IP, Deliverable D.A6.4, January 2006b.

[http://athena.akmodeling.com/Team/Repository/Projects/Project\\_216/Upload/FileSharing/Del\\_4th6Ms/060221\\_AT](http://athena.akmodeling.com/Team/Repository/Projects/Project_216/Upload/FileSharing/Del_4th6Ms/060221_AT)

[ATHENA A6 2006c] ATHENA A6, "D.A6.2: Enhanced Registry/Repository Infrastructure, Version 1.0", ATHENA IP, Deliverable D.A6.2, February 2006c.

[http://athena.akmodeling.com/Team/Repository/Projects/Project\\_216/Upload/FileSharing/Del\\_4th6Ms/060221\\_AT](http://athena.akmodeling.com/Team/Repository/Projects/Project_216/Upload/FileSharing/Del_4th6Ms/060221_AT)

[ATHENA A7 2006] ATHENA A7, "D.A7.1: Analysis of Industry Best Practice in Business Documents and Protocols, Version 1.0", ATHENA IP, Deliverable D.A7.1, January 2006.

[http://athena.troux.com/Team/Repository/Projects/Project\\_216/Upload/FileSharing/Del\\_4th6Ms/060221\\_AT](http://athena.troux.com/Team/Repository/Projects/Project_216/Upload/FileSharing/Del_4th6Ms/060221_AT)

[ATHENA A7 2007a] ATHENA A7, "D.A7.2: ATHENA Approach to Business Document and Protocol Management", ATHENA IP, Deliverable D.A7.2, January 2007a.

[ATHENA A7 2007b] ATHENA A7, "D.A7.3: Business Content for Selected Industry Best Practice", ATHENA IP, Deliverable D.A7.3, January 2007b.

[ATHENA A8 2006] ATHENA A8, "D.A8.1: Analysis of Interoperability Situation and needs of SMEs and current approaches, Version 1.0", ATHENA IP, Deliverable D.A8.1, January 2006.

[http://athena.troux.com/Team/Repository/Projects/Project\\_216/Upload/FileSharing/Del\\_4th6Ms/060224\\_AT](http://athena.troux.com/Team/Repository/Projects/Project_216/Upload/FileSharing/Del_4th6Ms/060224_AT)

[ATHENA A8 2007a] ATHENA A8, "D.A8.2: Guidelines and Best Practices for applying the ATHENA Interoperability Framework to support SME participation in Digital Ecosystems", ATHENA IP, Deliverable D.A8.2, January 2007a.

[ATHENA A8 2007b] ATHENA A8, "D.A8.3: Updated ATHENA tools and infrastructures supporting B5 Sub Projects", ATHENA IP, Deliverable D.A8.3, January 2007b.

[ATHENA B3 2006a] ATHENA B3, "D.B3.1: Business Interoperability Framework, Version 1.01", ATHENA IP, Deliverable D.B3.1, 31 January 2006a.

[http://athena.akmodeling.com/Team/Repository/Projects/Project\\_212/Upload/Attachments/Results/Deliverables](http://athena.akmodeling.com/Team/Repository/Projects/Project_212/Upload/Attachments/Results/Deliverables)

[ATHENA B3 2006b] ATHENA B3, "D.B3.3: Interoperability Impact Analysis Model", ATHENA IP, Deliverable D.B3.3, 30 November 2006b.

[ATHENA B4 2004] ATHENA B4, "D.B4.1: Dynamic Requirement Definition Principles, Version 1.0", ATHENA IP, Deliverable D.B4.1, September 2004.

[http://athena.akmodeling.com/Team/Repository/Projects/Project\\_216/Upload/FileSharing/Del\\_1st6Ms/040921\\_AT](http://athena.akmodeling.com/Team/Repository/Projects/Project_216/Upload/FileSharing/Del_1st6Ms/040921_AT)

[ATHENA B4 2005] ATHENA B4, "D.B4.2: Global Processes of Reference, Version 1.1", ATHENA IP, Deliverable D.B4.2, November 2005.

[http://athena.akmodeling.com/Team/Repository/Projects/Project\\_216/Upload/FileSharing/Del\\_4th6Ms/060221\\_AT](http://athena.akmodeling.com/Team/Repository/Projects/Project_216/Upload/FileSharing/Del_4th6Ms/060221_AT)

[ATHENA B4 2006a] ATHENA B4, "D.B4.4: DRDS, Version 1.0", ATHENA IP, Deliverable D.B4.4, February 2006a.

[http://athena.akmodeling.com/Team/Repository/Projects/Project\\_216/Upload/FileSharing/Del\\_4th6Ms/060221\\_AT](http://athena.akmodeling.com/Team/Repository/Projects/Project_216/Upload/FileSharing/Del_4th6Ms/060221_AT)

[ATHENA B4 2006b] ATHENA B4, "D.B4.3: Scenarios Mapped with Interoperability issues, Version 1.0", ATHENA IP, Deliverable D.B4.3, January 2006b.

[http://athena.akmodeling.com/Team/Repository/Projects/Project\\_216/Upload/FileSharing/Del\\_4th6Ms/0602](http://athena.akmodeling.com/Team/Repository/Projects/Project_216/Upload/FileSharing/Del_4th6Ms/0602)

[ATHENA B5 2005] ATHENA B5, "D.B5.1: Methodology & Technology Testing Report, Version 2.0", ATHENA IP, Deliverable D.B5.1, July 2005.

[http://athena.akmodeling.com/Team/Repository/Projects/Project\\_216/Upload/FileSharing/Del\\_3rd6Ms/0509](http://athena.akmodeling.com/Team/Repository/Projects/Project_216/Upload/FileSharing/Del_3rd6Ms/0509)

[ATHENA B5 2007] ATHENA B5, "D.B5.3: Users Guide to Athena Prototyping Services", ATHENA IP, Deliverable D.B5.3, March 2007.

[ATHENA C1 2006] ATHENA C1, "D.C1.4d: Updated Implementation Plan M25-M36, Version 0.9", ATHENA IP, Deliverable D.C1.4d, February 2006.

[http://athena.akmodeling.com/Team/Repository/Projects/Project\\_216/Upload/FileSharing/Del\\_4th6Ms/0603](http://athena.akmodeling.com/Team/Repository/Projects/Project_216/Upload/FileSharing/Del_4th6Ms/0603)

[BEA Systems, et al. 2003] BEA Systems, International Business Machines Corporation, Microsoft Corporation, SAP AG, and Siebel Systems, "Business Process Execution Language for Web Services, Version 1.1", 2003.

<http://www.ibm.com/developerworks/webservices/library/ws-bpel/>

[Becker, et al. 1995] J. Becker, M. Rosemann, and R. Schütte, "Grundsätze ordnungsmäßiger Modellierung", *Wirtschaftsinformatik*, vol. 37, no. 5, pp. 435-445, 1995.

[Benguria, et al. 2006] G. Benguria, X. Larrucea, B. Elvesæter, T. Neple, A. Beardsmore, and M. Friess, "A Platform Independent Model for Service Oriented Architectures", in Proc. of the 2nd Conference for Interoperability for Enterprise and Software Applications (I-ESA'06), Bordeaux, France, 2006.

[ECMA 1990] ECMA, "A Reference Model for Frameworks of Computer Assisted Software Engineering Environment", Technical report ECMA/TR55, December 1990.

[Elvesæter, et al. 2005] B. Elvesæter, R. K. Rolfsen, F. Lillehagen, and D. Karlsen, "Integrated Enterprise Service Architecture", in Proc. of the 12th ISPE International Conference on Concurrent Engineering (CE 2005), Fort Worth, Texas, USA, 2005, M. Sobolewski and P. Ghodous (Eds.), International Society for Productivity Enhancement, Inc., NY, USA, pp. 129-134.

[EUP 2006] EUP, "Enterprise Unified Process (EUP) Home Page", Enterprise Unified Process, 2006. <http://www.enterpriseunifiedprocess.com/>

[Figay] N. Figay, "ATHENA Harmonization Model".

<http://nfig.hd.free.fr/ATHENA/HarmonisedModels/>

[Gruber 1993] T. R. Gruber, "A translation approach to portable ontologies", *Knowledge Acquisition*, vol. 5, no. 2, pp. 199-220, 1993.

[IBM] IBM, "Rational Software Modeler".

<http://www-306.ibm.com/software/awdtools/modeler/swmodeler/>

[IBM, et al. 2004] IBM, Adaptive, Borland, Data Access Technologies, EDS, and 88 Solutions, "Business Process Definition Metamodel - Revised submission", Object Management Group, bei/2004-01-02, 2004.  
<http://www.bpmn.org/Documents/BPDM/OMG-BPD-2004-01-12-Revision.pdf>

[IBM Corporation, et al. 2004] IBM Corporation, Adaptive, Borland, Data Access Technologies, EDS, and Solutions, "Business Process Definition Metamodel - Revised Submission to BEI RFP bei/2003-01-06", Object Management Group (OMG), Document bei/04-08-03, 2004. <http://www.omg.org/docs/bei/04-08-03.pdf>

[IDABC 2004] IDABC, "European Interoperability Framework for Pan-European eGovernment Services, Version 1.0", IDABC, 2004.  
<http://europa.eu.int/idabc/en/document/3761>

[IDABC 2007] IDABC, "European Interoperability Framework for Pan-European eGovernment Services, Version 2.0", IDABC, 2007. Forthcoming at <http://ec.europa.eu/idabc/>

[IDEAS 2003a] IDEAS, "A Gap Analysis (second version)", IDEAS, Deliverable D.3.4, 2003a. <http://isg.uninova.pt/pub.ideas/bscw.cgi/d68826/IDEAS%20D34%2bD35%2bD36.zip>

[IDEAS 2003b] IDEAS, "Required Activities in Research, Technology and Standardisation to close the RTS Gap (second version)", IDEAS, Deliverable D.3.5, 2003b.  
<http://isg.uninova.pt/pub.ideas/bscw.cgi/d68826/IDEAS%20D34%2bD35%2bD36.zip>

[IDEAS 2003c] IDEAS, "Roadmaps and Recommendations on RTS activities (second version)", IDEAS, Deliverable D.3.6, 2003c.  
<http://isg.uninova.pt/pub.ideas/bscw.cgi/d68826/IDEAS%20D34%2bD35%2bD36.zip>

[IDEAS 2004] IDEAS, "IDEAS Home Page", IDEAS, 2004. <http://www.ideas-roadmap.net/>

[IEEE 1990] IEEE, "IEEE Standard Computer Dictionary: A Compilation of IEEE Standard Computer Glossaries", Institute of Electrical and Electronics Engineers, 1990.

[IEEE 2000] IEEE, "IEEE Std 1471-2000: IEEE Recommended Practice for Architectural Description of Software-Intensive Systems", IEEE, October 2000.

[INTEROP DI 2006a] INTEROP DI, "Interoperability knowledge corpus - Intermediate report, Version 1.0", INTEROP NoE, Deliverable DI.1, 7 July 2006a.

[INTEROP DI 2006b] INTEROP DI, "Interoperability knowledge corpus - Advanced report, Version 0.5", INTEROP NoE, Deliverable DI.2, 15 December 2006b.

[INTEROP TG6 2005] INTEROP TG6, "State of the Art: Exploration of Methods and Method Engineering Approaches", Deliverable DTG 6.1, 2005.

[INTEROP TG6 2006] INTEROP TG6, "INTEROP Method Repository", INTEROP NoE, Deliverable DTG 6.2, 2006.

[ISO 1993] ISO, "The Directory: Overview of concepts, models and services", International

Organization for Standardization (ISO), ISO/IEC 9594-1, 1993.

[ISO 1994] ISO, "Industrial automation systems and integration - Product data representation and exchange - Part 1: Overview and fundamental principles", International Organization for Standardization (ISO), ISO 10303-1, 1994.

[ISO 1998] ISO, "Industrial automation systems - Concepts and rules for enterprise models", International Organization for Standardization (ISO), ISO 14258, 1998.

[ITU-TS 1995a] ITU-TS, "Basic Reference Model of Open Distributed Processing - Part 1: Overview and guide to use the Reference Model", Rec.X901 (ISO/IEC 10746-1), 1995a.

[ITU-TS 1995b] ITU-TS, "Basic Reference Model of Open Distributed Processing - Part 2: Descriptive model", Rec.X902 (ISO/IEC 10746-2), 1995b.

[ITU-TS 1995c] ITU-TS, "Basic Reference Model of Open Distributed Processing - Part 3: Prescriptive model", Rec.X903 (ISO/IEC 10746-3), 1995c.

[ITU-TS 1995d] ITU-TS, "Basic Reference Model of Open Distributed Processing - Part 4: Architectural Semantics", Rec.X904 (ISO/IEC 10746-4), 1995d.

[Jacobson, et al. 1999] I. Jacobson, G. Booch, and J. Rumbaugh, "The Unified Software Development Process", Addison-Wesley Longman, 1999.

[Jochem 2001] R. Jochem, "Integrierte Unternehmensplanung auf der Basis von Unternehmensmodellen", TU-Berlin, Dissertation, 2001.

[Li, et al. 2006] M.-S. Li, R. Cabral, G. Doumeingts, and K. Popplewell, "Enterprise Interoperability Research Roadmap, Final Version, Version 4.0", July 2006.

[Merriam-Webster] Merriam-Webster, "Merriam-Webster Online". <http://www.m-w.com/>

[OAGi] OAGi, "Open Application Group (OAGi)". <http://www.openapplications.org/>

[OASIS] OASIS, "Web Services Business Process Execution Language (WSBPEL) TC". [http://www.oasis-open.org/committees/tc\\_home.php?wg\\_abbrev=wsbpel](http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=wsbpel)

[OASIS 2005] OASIS, "eXML - Enabling a Global Electronic Market", OASIS, 2005. <http://www.ebxml.org/>

[OASIS 2006] OASIS, "Reference Model for Service Oriented Architecture 1.0", OASIS, OASIS Standard, 12 October 2006. <http://docs.oasis-open.org/soa-rm/v1.0/soa-rm.pdf>

[Object Management Group 2002] Object Management Group, "UML Profile for Enterprise Distributed Object Computing Specification", Object Management Group, ptc/02-02-05, 2002. <http://www.omg.org/technology/documents/formal/edoc.htm>

[Object Management Group 2003] Object Management Group, "Business Process Definition Metamodel - Request For Proposal", Object Management Group, bei/2003-01-06, 2003. <http://www.omg.org/docs/bei/03-01-06.pdf>



[OMG] OMG, "ORB Basics", Object Management Group (OMG).  
[http://www.omg.org/gettingstarted/orb\\_basics.htm](http://www.omg.org/gettingstarted/orb_basics.htm)

[OMG 2000] OMG, "Product Data Management Enablers Specification, Version 1.3", Object Management Group (OMG), Document formal/00-11-11, November 2000.  
<http://www.omg.org/docs/formal/00-11-11.pdf>

[OMG 2001] OMG, "Model Driven Architecture (MDA)", Object Management Group (OMG), Document ormsc/01-07-01, July 2001.  
<http://www.omg.org/docs/ormsc/01-07-01.pdf>

[OMG 2004] OMG, "UML Profile for Modeling Quality of Service and Fault Tolerance Characteristics and Mechanisms", Object Management Group, ptc/04-09-01, 2004.  
<http://www.omg.org/docs/ptc/04-09-01.pdf>

[OMG 2006a] OMG, "OMG SysML Specification", Object Management Group (OMG), OMG Adopted Specification ptc/06-05-04, May 2006a.  
<http://www.omg.org/docs/ptc/06-05-04.pdf>

[OMG 2006b] OMG, "CORBA Component Model Specification, Version 4.0", Object Management Group (OMG), OMG Available Specification formal/06-04-01, April 2006b.  
<http://www.omg.org/docs/formal/06-04-01.pdf>

[Ralyté and Rolland 2001] J. Ralyté and C. Rolland, "An Approach for Method Reengineering", in Proc. of the 20th International Conference on Conceptual Modeling (ER2001), Yokohama, Japan, 2001, LNCS 2224, H. Kunii, S. Jajodia, and A. Solvberg (Eds.), Springer-Verlag, pp. 471-484.  
[http://cui.unige.ch/~ralyte/publications/ER2001\(Ralyte-Rolland\).pdf](http://cui.unige.ch/~ralyte/publications/ER2001(Ralyte-Rolland).pdf)

[Ralyté, et al. 2006] J. Ralyté, P. Backlund, H. Kühn, and M. Jeusfeld, "Method Chunks for Interoperability", in Proc. of the 25th International Conference on Conceptual Modelling (ER'2006), Tucson, Arizona, USA, 2006, LNCS 4215, H. Kunii, S. Jajodia, and A. Solvberg (Eds.), Springer-Verlag, pp. 339-353.  
[http://cui.unige.ch/~ralyte/publications/ER2001\(Ralyte-Rolland\).pdf](http://cui.unige.ch/~ralyte/publications/ER2001(Ralyte-Rolland).pdf)

[SEI] SEI, "CMMI Web Site", Software Engineering Institute (SEI), Carnegie Mellon.  
<http://www.sei.cmu.edu/cmmi/>

[Sourceforge.net 2006] Sourceforge.net, "Platform-independent model for service-oriented architecture (PIM4SOA)", The PIM4SOA project, 2006. <http://pim4soa.sourceforge.net/>

[Stanford Medical Informatics] Stanford Medical Informatics, "Protege".  
<http://protege.stanford.edu/>

[Sun] Sun, "Java Enterprise Edition". <http://java.sun.com/javaee/>

[Trous Technologies] Trous Technologies, "Metis Architect".  
[http://www.troux.com/products/metis\\_architect/](http://www.troux.com/products/metis_architect/)

[Truex, et al. 1999] D. P. Truex, R. Baskerville, and H. Klein, "Growing Systems in Emergent Organizations", Communications of the ACM, vol. 42, no. 8, pp. 117-123, 1999.  
[UN/CEFACT 2003] UN/CEFACT, "Core Components Technical Specification - Part 8 of the ebXML Framework, Version 2.01", 15 November 2003.

[W3C 2001] W3C, "Web Services Description Language (WSDL) 1.1", World Wide Web Consortium (W3C), W3C Note, 15 March 2001. <http://www.w3c.org/TR/wsdl>

[W3C 2004a] W3C, "XML Schema Part 0: Primer Second Edition", World Wide Web Consortium (W3C), W3C Recommendation, 28 October 2004a.  
<http://www.w3.org/TR/xmlschema-0/>

[W3C 2004b] W3C, "Web Services Description Language (WSDL) Version 2.0 Part 1: Core Language", World Wide Web Consortium (W3C), W3C Working Draft, 3 August 2004b.  
<http://www.w3.org/TR/2004/WD-wsdl20-20040803/>

[W3C 2004c] W3C, "Web Services Architecture", World Wide Web Consortium (W3C), W3C Working Group Note, 11 February 2004c. <http://www.w3.org/TR/ws-arch/>

[WfMC] WfMC, "Workflow Management Coalition (WfMC)". <http://www.wfmc.org/>

## 2. Reference architecture

---

### 2.1. Reference architecture

---

#### 2.1.1. Introduction

---

This part of the framework focuses on the interoperability reference architecture and covers the following topics:

- [Interoperability reference architecture](#)
- [Interoperability classification](#)

### 2.2. Interoperability reference architecture

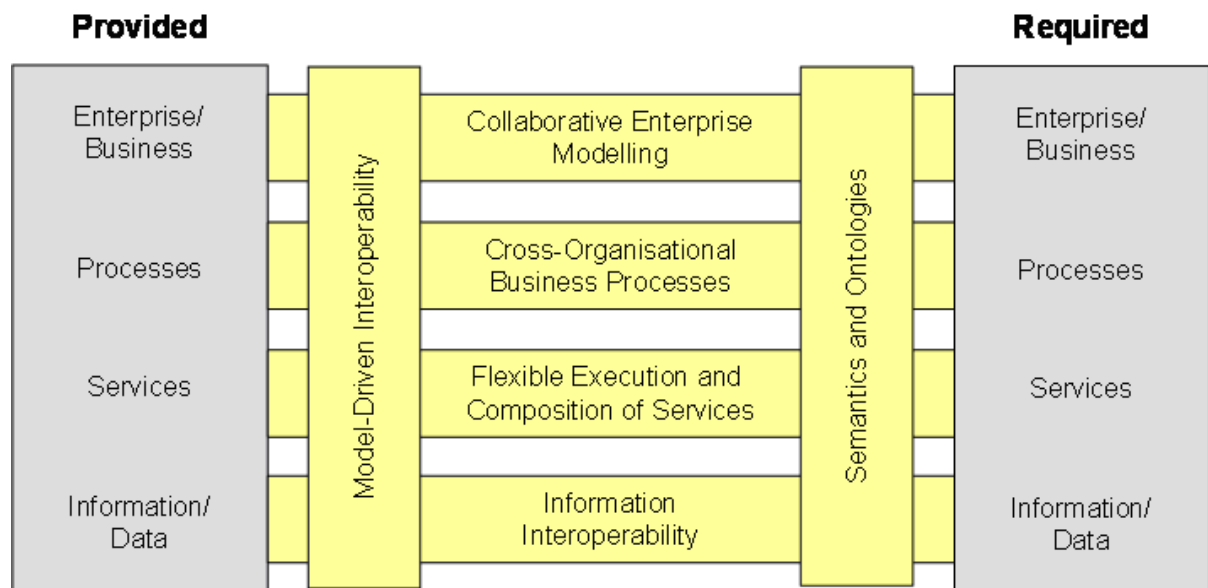
---

#### 2.2.1. Overview

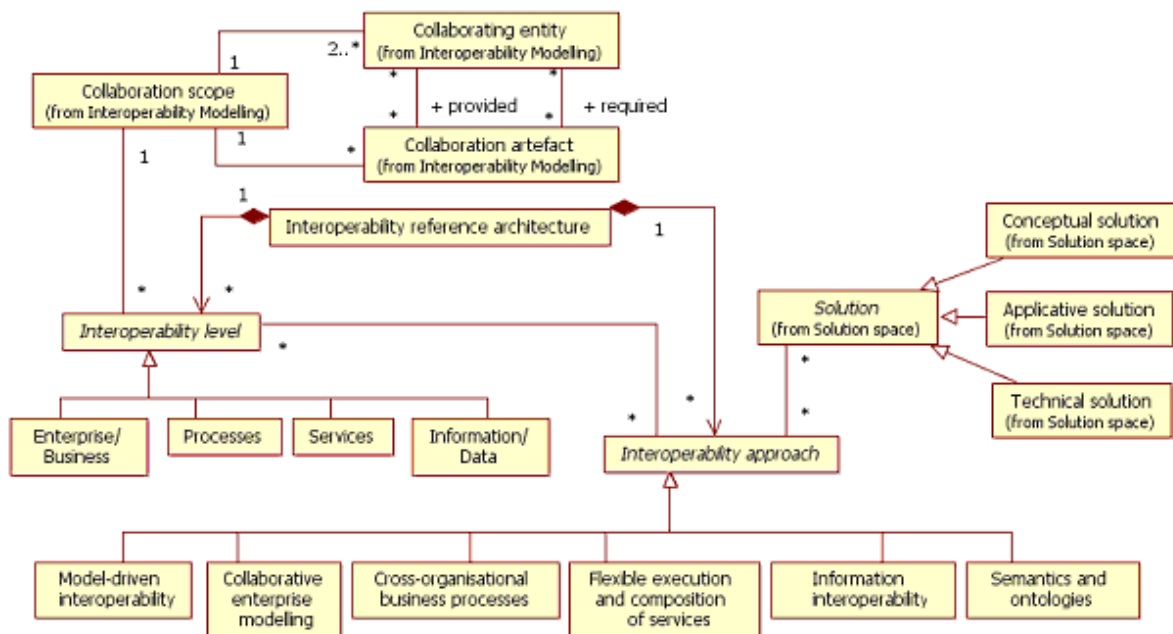
---

The **ATHENA Interoperability Framework (AIF)** defines an *interoperability reference architecture* that relates the modelling solutions coming from the three different research areas of ATHENA, namely enterprise modelling, architectures and platforms, and ontology. The figure below illustrates the reference architecture that focuses on the *provided* and *required* artefacts of two collaborating enterprises.





In this section we will formalise the representation of the reference architecture as a part of the *interoperability reference architecture* concept domain (defined in **AIF conceptual model**). A formalized view of the conceptual model is shown in the figure below and elaborated further below.



### 2.2.1.1. Interoperability levels

---

An *interoperability reference architecture* relates a set of *interoperability levels* and a set of *interoperability approaches*. An *interoperability level* denotes the capability level of two or more *collaborating entities* to support interoperation.

- Interoperability at the *enterprise/business level* should be seen as the organisational and operational ability of an enterprise to factually co-operate with other, external organisations in spite of e.g. different working practices, legislations, cultures and commercial approaches. From an ICT system perspective it means that the ICT infrastructure is flexible and adaptable to the changing business requirements and the organisational structures of the enterprise.
- Interoperability at the *processes level* is the capability to make various processes work together. A process defines the sequence of the services (functions) according to some specific needs of a *collaborating entity*.
- Interoperability at the *services level* is concerned with identifying, composing and executing various applications (designed and implemented independently). Services are an abstraction and an encapsulation of the functionality provided and/or required by a *collaborating entity*.
- Interoperability at the *information/data level* is related to the management, exchange and processing of different documents, messages and/or structures by different *collaborating entities*.

#### 2.2.1.2. Interoperability approaches

---

The concept *interoperability approach* is here used to designate the six technology approaches resulting from the research activities performed in ATHENA. The interoperability approaches help us to support interoperations at the various *interoperability levels*.

For each of these levels we prescribe a *model-driven interoperability* approach that cuts across the *interoperability levels* where models are used to formalise and exchange the provided and required artefacts that must be negotiated and agreed upon. ATHENA defines a set of metamodels and languages that can be supported by tools and methods to construct the models in question. Starting at the top:

- *Collaborative enterprise modelling* concerns the exchange and alignment of knowledge models for describing the processes, organisations, products and systems in the collaboration context. *Collaborative enterprise modelling* is supported by the POP\* metamodel [ATHENA A1 2005a].
- Modelling of *cross-organisational business processes* focuses on defining process views that describes the interactions between two or more collaborating entities. In a networked enterprise, it is also necessary to study how to connect internal processes of two companies to create *cross-organisational business process*. This is supported by the CBP (cross-organisational business process) metamodel [ATHENA A2 2005b].
- *Flexible execution and composition of services* is concerned with identifying, composing and executing various applications. Modelling *flexible execution and composition of*

*services* can be supported by the PIM4SOA (platform-independent model for service-oriented architecture) metamodel [ATHENA A6 2006b, Sourceforge.net 2006].

- *Information interoperability* is related to management, exchange and processing of different documents, messages and other information structures. The XML Schema Definition Language (XSD) [W3C 2004a] will be the foundation for ATHENA solutions at this *interoperability level*.

To overcome the semantic barriers which emerge from different interpretations of syntactic descriptions, precise, computer processable meaning must be associated with the models expressed on the different levels. It has to be ensured that *semantics* are exchangeable and based on common understanding in order to enhance interoperability. This can be achieved using *ontologies* and an annotation formalism for defining meaning in the exchanged models. The OPAL (object, process, actor modelling language) [ATHENA A3 2005] is an ontology language that offers a number of modelling notions to more precisely define the meaning of concepts. This allows us to relate concepts at the different levels (ensuring consistency amongst the levels) and relate concepts at the same level e.g. supporting *information interoperability*.

## 2.3. Characterisation of interoperability issues

### 2.3.1. Interoperability classification

Taxonomic classification is the act of placing an object or concept into a set or sets of categories based on the properties of the object or concept. *Interoperability classification* focuses on relating *interoperability issues* with *solutions*. This can be through a *classification schema* which defines a set of *categories*.

The ATHENA deliverable [ATHENA B4 2006b] identifies 31 *interoperability issues*. The interoperability issues are classified according to business management (BM), process management (PM), knowledge management (KM), information management (IM), service management (SM) and data management (DM). The table below presents an overview of the *interoperability issues* identified. The issues are further described below.

Issue	BM	PM	KM	IM	SM	DM
Business processes "hard-coded" in applications	BM1					
Manual business processes	BM2					

Linking decision-making activities	BM3					
Aggregated views of business information	BM4					
Translating strategic goals	BM5					
Inferring benefits of objectives	BM6					
Support effective decision-making	BM7					
Focus on process management		PM1				
Process interoperability		PM2				
Shorten time from order to delivery		PM3				
Supplier rating		PM4				
Product descriptions			KM1			
Product related knowledge			KM2			
Knowledge organisation based on domain standards			KM3			

Knowledge governance processes			KM4			
Enterprise description and knowledge management			KM5			
Integrated application execution			KM6			
Stakeholder involvement			KM7			
Negotiation space based on models			KM8			
Integration and federation of objectives			KM9			
Link viewpoints for decision-making			KM10			
Identities are hard-coded				IM1		
De-coupled layers					SM1	
Customisation of software products					SM2	
Auto descriptive applications					SM3	
Internal information					SM4	

model						
Documented publication of applications and services					SM5	
Data format interoperability						DM1
Distributed inconsistent data						DM2
Support for middleware framework						DM3

### 2.3.1.1. Business management

- **BM1:** Business Processes "hard-coded" in applications. Improvement of application deployment in terms of business internal "time to market" and high programming code reuse level.
- **BM2:** Repetitive manual business processes for regular bulk orders. Much of the manufactured products are generic and this involves repeated periodic processing of similar or identical orders.
- **BM3:** Link decision-making activities with strategic plans, development and operational results. Business decision-making activities are of paramount importance to enterprises, affecting the day-to-day operations as well as medium and long-term planning and execution of activities. Therefore, an integral mechanism is required to support the decision-making process at various levels with strategic plans, by considering product/project development activities and results coming out of daily operations.
- **BM4:** Provision of (near) real-time aggregated views of key business information. Related to the above business decision-making activities, these aggregated views could be provided as services to the roles and actors required, accessing and integrating data in existing legacy systems. Such aggregated views will enable actors to take more accurate and timely decisions, exploiting to the full extend the capabilities of existing ICT systems.
- **BM5:** Target setting decomposition and objectives mapping to process roles. Among the major challenges large enterprises are facing today is the capability to translate strategic goals into detailed tactical and operational objectives and targets for every business unit and major business process. The timely execution of and the ability to re-adjust and fine-tune this activity upon fluctuations of market conditions can have a significant impact on the profitability and, in some occurrences, on the survival of these enterprises

within the extremely competitive environment they operate. Therefore, there is a need to develop a mechanism to structure and facilitate that process of business strategy translation into business process objectives, attributable to the different roles of its key personnel.

- **BM6:** Objectives inferring to tangible benefits and expectations. One step further to the above issue, the detailed tactical and operational objectives per role have to be justified by the benefits and expectations one can bring back into each role, and all together into the business process/unit, thus justifying the budget and other resources to be allocated for its realisation. Therefore, there is a need to develop a similar to the above mechanism to inference role objectives into attainable benefits and expectations.
- **BM7:** Link program, product and enterprise viewpoint to support effective decision-making, strategic plans, development and operational results. Several viewpoints are structuring management of activities within an enterprise and a networked organisation. Each person has consequently different antagonist objectives, with sometimes unclear definition of priority. Linking program, product and enterprise viewpoints should allow more effective decision making, negotiation and activities within the collaborative product design within a networked organisation. It should also allow a better alignment between decision making, strategic plans, development and operational results.

#### 2.3.1.2. Process management

---

- **PM1:** Applications focus on transactions, not on processes. Usage of graphical tools to manage process parameterisations and process management in a virtual enterprise context, gaining programming activities reduction/reset (code implementation).
- **PM2:** Process interoperability. Ability of a process or application to make "visible" the requested services/interfaces and the offered services/interfaces.
- **PM3:** Lag. Time from product order to delivery could be shorter. Shortening time from ordering to receiving raw materials from the supplier has a direct effect on the delivery date of the finished product.
- **PM4:** Time spent rating supplier. Many companies conduct tri-monthly reviews of their suppliers to ensure that standards are kept.

#### 2.3.1.3. Knowledge management

---

- **KM1:** Confusion resulting from poor product descriptions. Clients very often order the wrong products!
- **KM2:** Product related knowledge sharing within and between product life cycle phases. Adequate and common understanding of product and process information is required to support knowledge sharing between different product life-cycle phases, rather than merely transferring information between them.
- **KM3:** Domain standards based knowledge organisation. The knowledge meta-meta model should allow integration of concerned industrial sector meta-models, ICT used solutions meta-models and Enterprise Modelling used solutions meta-models.
- **KM4:** Establishment of Knowledge governance process, standards and best practices for

a networked organization without governance and long term strategy, islandisation of knowledge applications will lead to non-interoperability.

- **KM5:** Enterprise description and knowledge management in various aspects and dimensions (organisation, role, decision, process, product, system) knowledge capture, assimilation and management are some of the most important assets of an enterprise, an asset that creates significant added value to any existing information and communication infrastructure.
- **KM6:** Ability of integrated applications execution via custom, adaptive and model generated environment. Legacy applications integration and interoperability: existing applications that provide access to enterprise data and facilitate analysis and decision-making should be integrated using a standard technology that allows composition at service level, thus providing the reusability and flexibility of customized services composition and deployment. Model driven generation of interoperable custom and role-based workplaces: Models mapping and integration at system level, as well as tools for the transformation of the provided models to interoperable service description interfaces would allow the interoperability of system models and model generated workplaces.
- **KM7:** Support for stakeholders' involvement and collaboration:
  - Communication / collaboration infrastructure integration: use of standard middleware and communication protocols would allow the seamless communication and interoperability of model-generated workplaces applications.
  - Shared data integration: Reconciliation of business level information exchanged between the stakeholders that would allow their collaboration and common understanding is required. This probably implies business data integration at semantic or business meta-models level with the use of reference ontologies and/or mapped meta-models.
  - Data and data access synchronization: Working concurrently on the same data requires a synchronization mechanism that preserves their consistency and validity and distributes their valid state to the interested stakeholders.
- **KM8:** Negotiation space based on objectives models used by the enterprises of the networked organisation. Enterprise Modelling tools are used to support elaboration of enterprise objectives and roles, supporting quality trends (ISO 9001, CMM, CMMI, etc), but level of maturity of enterprises for usage of such tools or quality approach is not the same. To benefits from modelling and models, but also to help the less mature enterprises to integrate such tools, some neutral standards are necessary in order to prepare negotiation workspace between the actors of Collaborative Product Design actors in a networked organization.
- **KM9:** Integration and federation of Objectives to tangible benefits and expectations used models. Different and heterogeneous tools (Activity Based Costing, System Engineering, Scorecard, etc.) are used to link objectives to tangible benefits in one hand, to the expectation and requirement in the other hand, often within the same enterprise. A negotiation and decision workspace should support quick and easy federation or integration of these tools to efficiently support enterprise and program management and decision making.



- **KM10:** Link program, product and enterprise viewpoint to support effective decision-making, strategic plans, development and operational results. Several viewpoints are structuring management of activities within an enterprise and a networked organisation. Each person has consequently different antagonist objectives, with sometimes unclear definition of priority. Linking program, product and enterprise viewpoints should allow more effective decision making, negotiation and activities within the collaborative product design within a networked organisation. It should also allow a better alignment between decision making, strategic plans, development and operational results.

#### 2.3.1.4. Information management

---

- **IM1:** Identity and identification schemes are hard-coded. This makes cooperation and collaboration process modelling and execution very difficult. Inbound and outbound logistics have to be designed from knowledge structures, and services provided to decode and align logistics schemes.

#### 2.3.1.5. Software management

---

- **SM1:** De-coupled application layer and technical layer. In order to support agility of global information system, and independence between business logic and technical solutions implementing the awaited functionality. It should allow interchange-ability of software product components and real governance of the information system by enterprise and networked organisation.
- **SM2:** Easy customisation of the software product and automatic reorganization of the technical interfaces. As enterprises are more and more using Commercial of the Shelves, the used solutions are highly generic and require an important parameterisation/customisation and administration to adapt the solution to the business context. This customisation should be as easy as possible by operators, without implying modification of technical interfaces by software engineers.
- **SM3:** Auto descriptive applications. Capability from the software product solution to extract the business logic as business or enterprise model, with a standard and open format, in order to support custom, adaptive and model generated collaboration environment parameterisation.
- **SM4:** Internal information model of software products and applications based on standardised information models capacity for the networked organisation to rely on stable and software product independent business models to establish their collaboration, with minor impact of technical solutions evolution.
- **SM5:** Documented publication of applications and software products services. The idea is to allow easy usage of these applications when willing to establish collaboration, without used solution experts. It should be done through Application Programming Interfaces and service description according the numerous interface description standards (IDL, WSDL), for IT department and ICT integrators. It should be done through well structured and agile documentation that should be reusable by knowledge models. All these interfaces should be coherent and easily reflect customisation, an automated way.

### 2.3.1.6. Data management

- **DM1:** Data Format Interoperability: ability of a process/application to exchange data with one/more partners by means of a common data format or via a mapping between the proprietary format and an intermediate common format.
- **DM2:** Distributed inconsistent data: ability of a solution to guarantee data consistency and distributed data alignment in a virtual enterprise context.
- **DM3:** Support of the main technical middleware framework a coherent way. It includes STEP ISO information technical platform, CORBA and OMA, eBusiness infrastructure (Web services), Wfmc standards, J2EE and .NET. It aims to be able to easily collaborate with partners that have made some choices based on these technical platforms a seamless way.

## 3. Interoperability methodology

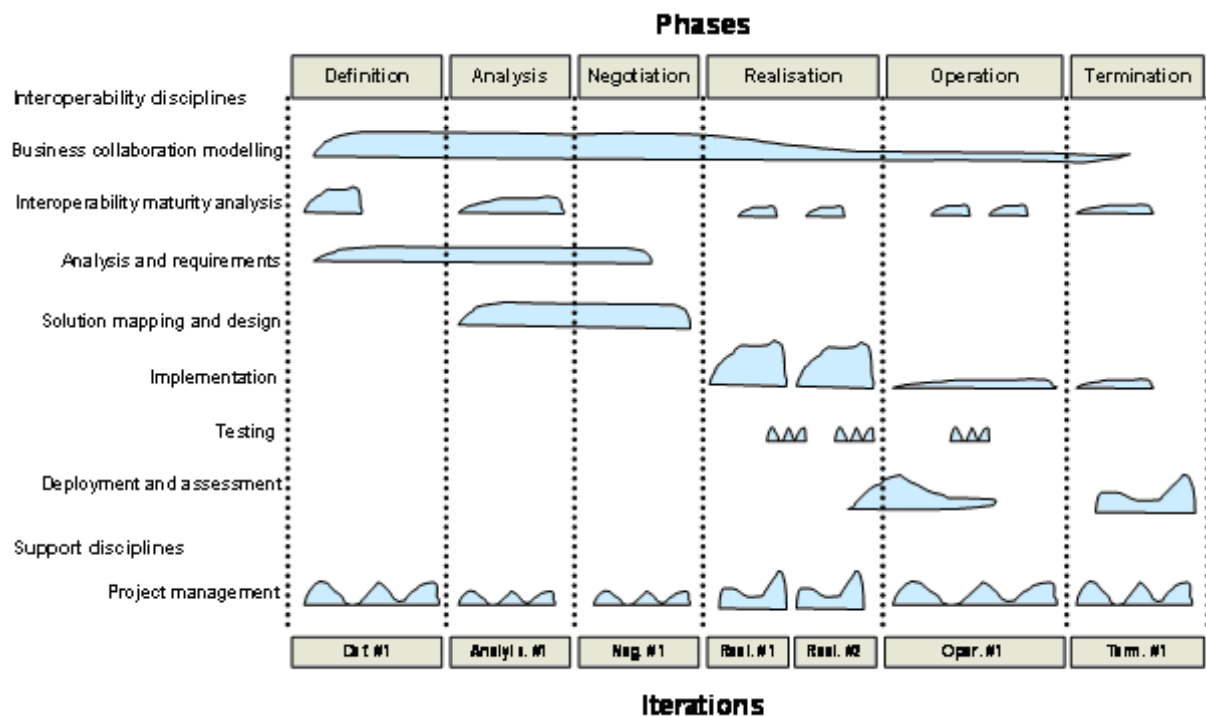
### 3.1. Interoperability methodology

#### 3.1.1. Introduction

This chapter details the applicative integration part of the AIF, specifying the ATHENA Interoperability Methodology (AIM), the AIM-related concepts, guidelines and how the AIM should be applied and how the resulting method could support different roles of an interoperability project.

#### 3.1.2. ATHENA Interoperability Methodology (AIM)

The ATHENA Interoperability Methodology is influenced by the Enterprise Unified Process (EUP) [EUP 2006]. EUP is an extension to the Unified Software Development Process (UP) [Jacobson, et al. 1999] which is a recognized and commonly used software development methodology. Whereas the UP defines a software development lifecycle, the EUP extends it to cover the entire ICT lifecycle. The AIF applicative framework builds on the EUP and extends it further by introducing new interoperability disciplines. An interoperability discipline is a group of activities within a specific interoperability field which are logically grouped together.

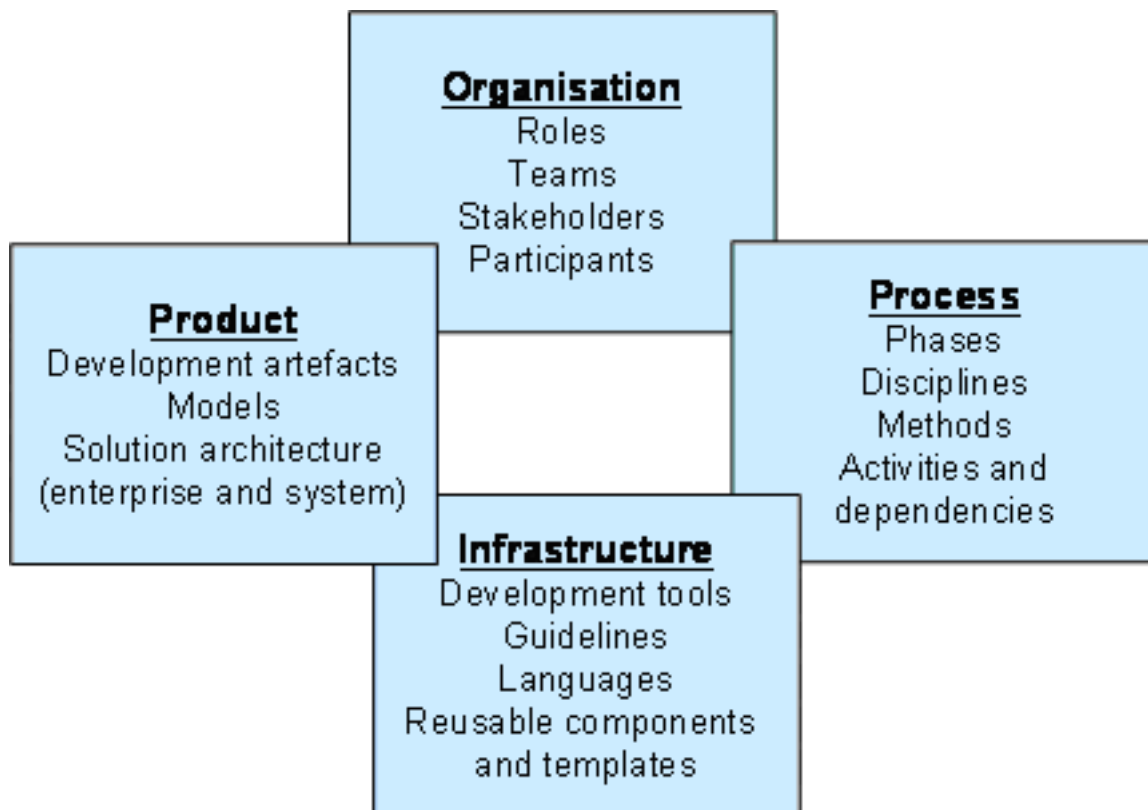


In the figure above, the ATHENA Interoperability Methodology is rendered from an overall perspective, showing the essential structure of phases and disciplines. The phases of an interoperability project life-cycle are represented by the columns. The rows in the figure outline the set of principles that characterise the mature approach for the definition, creation, operation and termination of an interoperability project. Within each discipline, the AIM recommends sets of activities to be performed in the different phases of the interoperability project. The kind of artefacts created and manipulated by the activities are varying dependent on the phase. The load of activities within a discipline will also vary dependent on the phase.

### 3.1.2.1. Architecture of the methodology framework

In order to be useful for industry, the AIF should guide companies in selecting the best ATHENA approach for their interoperability needs. For each approach, a methodology should describe:

- Which roles are involved in the project (organisation perspective)
- Which tasks they perform in which order (process perspective)
- Which tools they use for each task (infrastructure perspective)
- What the resulting artefacts and solutions are (product perspective)



Considerations related to the four perspectives are:

- **Organisation perspective:** Typical roles in an interoperability project include:
  - Business Manager with views on business project portfolio, dashboard for performance monitoring and governance, and strategic objectives and goals.
  - Customer Account Controller with views on total customer involvement, customer portfolio, status on delivery of solutions and services, and ongoing current work.
  - Chief Architect with business, EKA, and ICT descriptive views of which views are critical for the other roles and the desired solutions.
  - Model Manager with views on existing relevant models and contents, metamodels and metadata, and approach.
  - Solution Developer with views on business operations, solutions and users, logistics and maintenance.
  - Knowledge Worker or model builder with views on business network, EKA, modelling approaches, methodologies and languages, and logistics.
  - Product Designer and Engineer with views on business solution use in design and engineering, user services, and user requirements and solutions.
- **Process perspective:** Here we define the overall phases and steps in an interoperability project following the methodology.
  - We focus on methods that support the interoperability disciplines that are carried out

incremental and iterative within the phases. Each phase or step may be decomposed into activities.

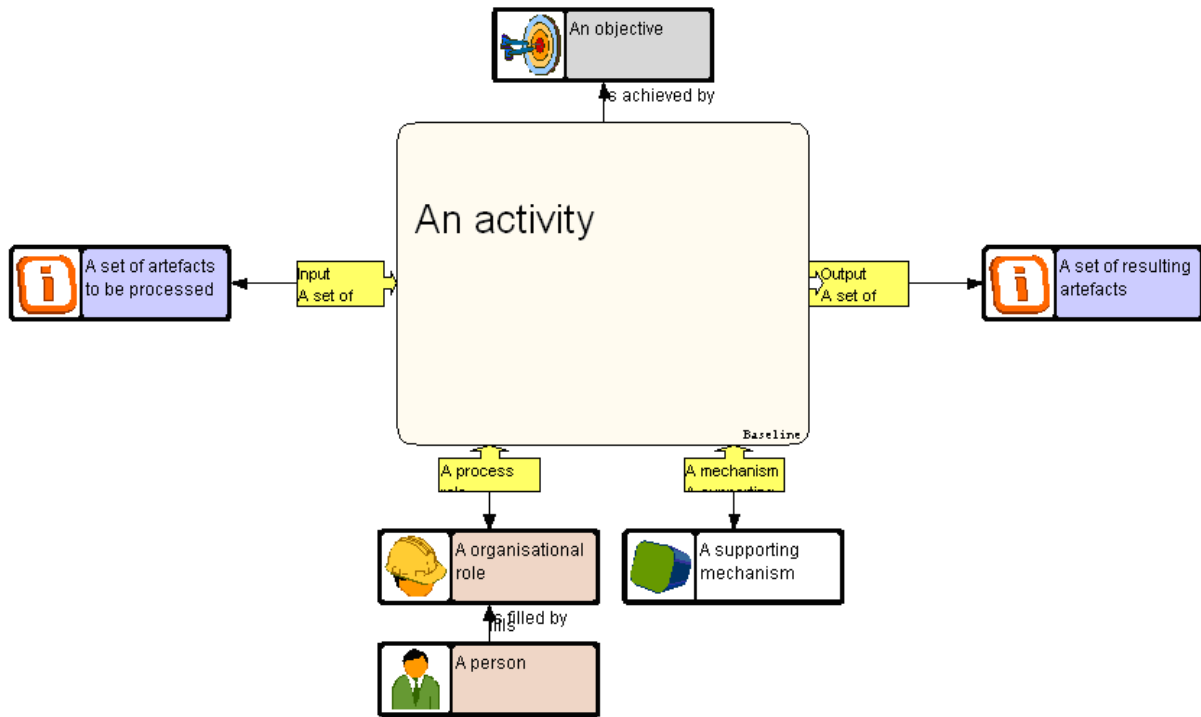
- The recommended disciplines in the AIM are described to the needed level of details sufficient to characterise the activities and the connected artefacts. The disciplines should be considered as placeholders for actual techniques, solutions and service offered by ATHENA to be used in the activities in question.
- Infrastructure perspective: Each method should define the tools that support it, which includes modelling tools, modelling languages, guidelines and documentation, reusable patterns and templates, programming tools, repositories and transformation tools.
- Product perspective: The products of an interoperability projects are the solution which consists of the technical system architecture that describes the ICT infrastructure and the enterprise architecture that describes the usage environments. The product perspective also covers artefacts that are created during the development projects such as models, documents, contracts, etc.

### 3.1.2.2. Model of the methodology framework

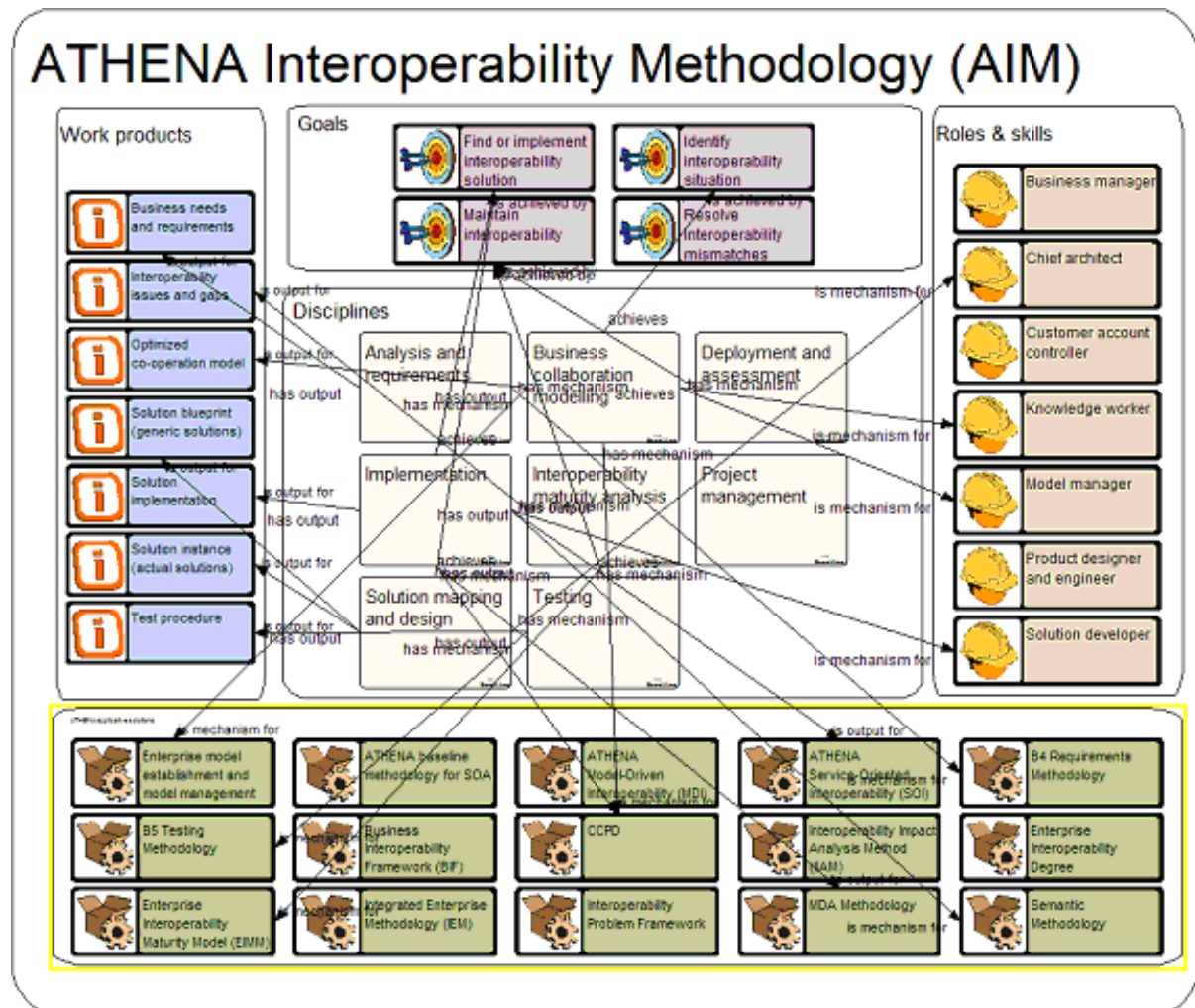
---

The AIM methodology can be made operational by implementing the different methods and activities in a method engineering platform. The recommended activities can be formalised as models that can be configured and executable in supporting the different roles of an interoperability project. The structure shown in the figure below can be used to describe the method components of the ATHENA Interoperability Methodology:

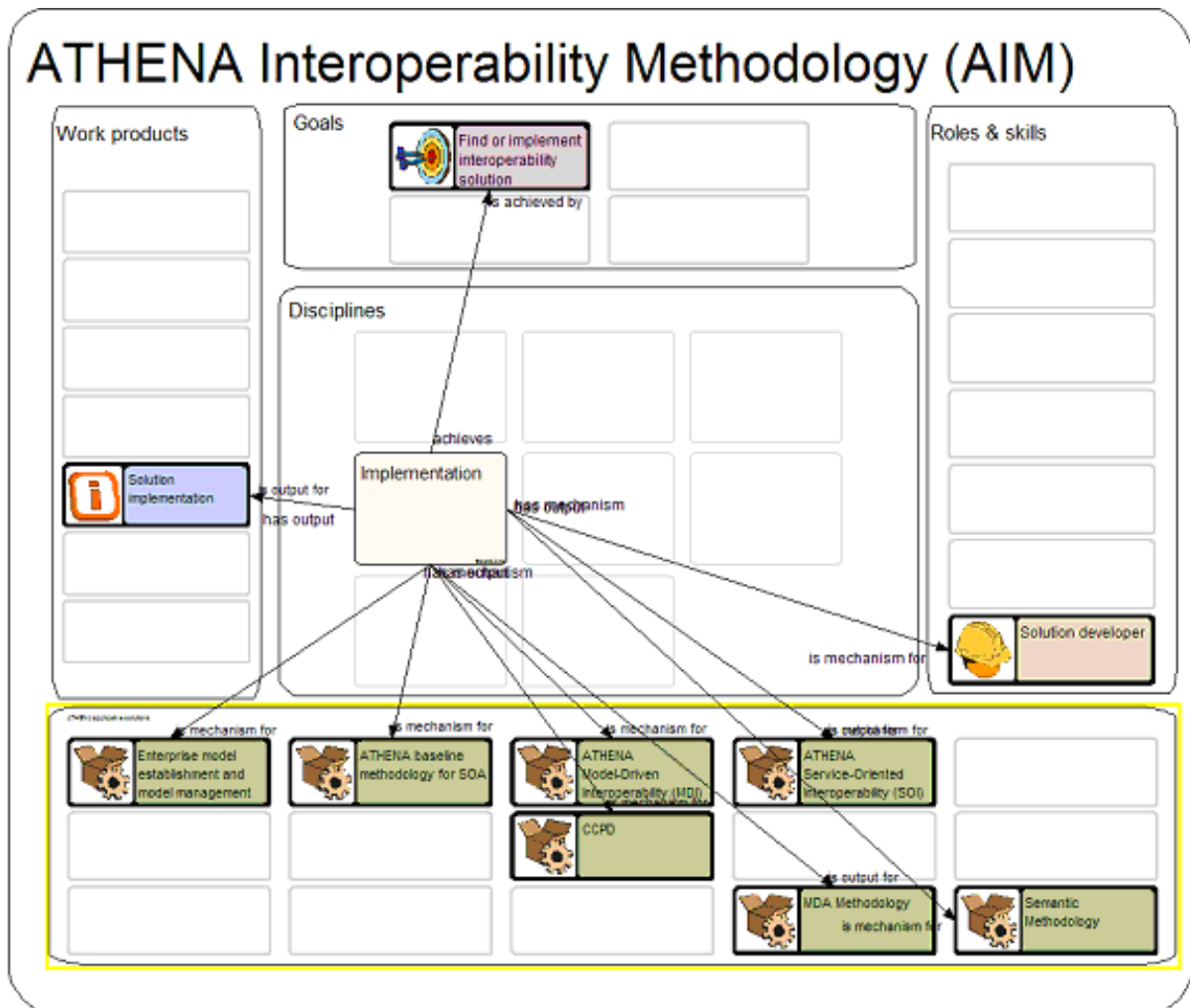
- Activity (i.e. what to do)
- Objective (i.e. what will be achieved by performing the activity)
- Input (i.e. the artefact to be processed in the activity)
- Output (i.e. the artefacts being the result of the activity)
- Roles (i.e. the way people are related to the activity; responsible, participant, customer, ...)
- Supporting Mechanisms (i.e. information, document, models, systems, processes that are supporting the activity, ...)



An example of applying the structure described above is shown in the figure below which gives a high-level overview of the AIM. The disciplines of the AIM are modelled as activities. For each discipline we describe relationships to the goals they support, the required and resulting work products (input and output), the roles that are involved, and finally the concrete methods developed in ATHENA that can be applied.



Such an approach can be used to tailor the methodology for different roles. Role-specific views can be created that filters out details that are not needed by certain roles and the different views can be consistently maintained in a method engineering platform. The figure below gives an example of the method components available to a solution developer.



## 3.2. Baseline interoperability methodology

### 3.2.1. Overview

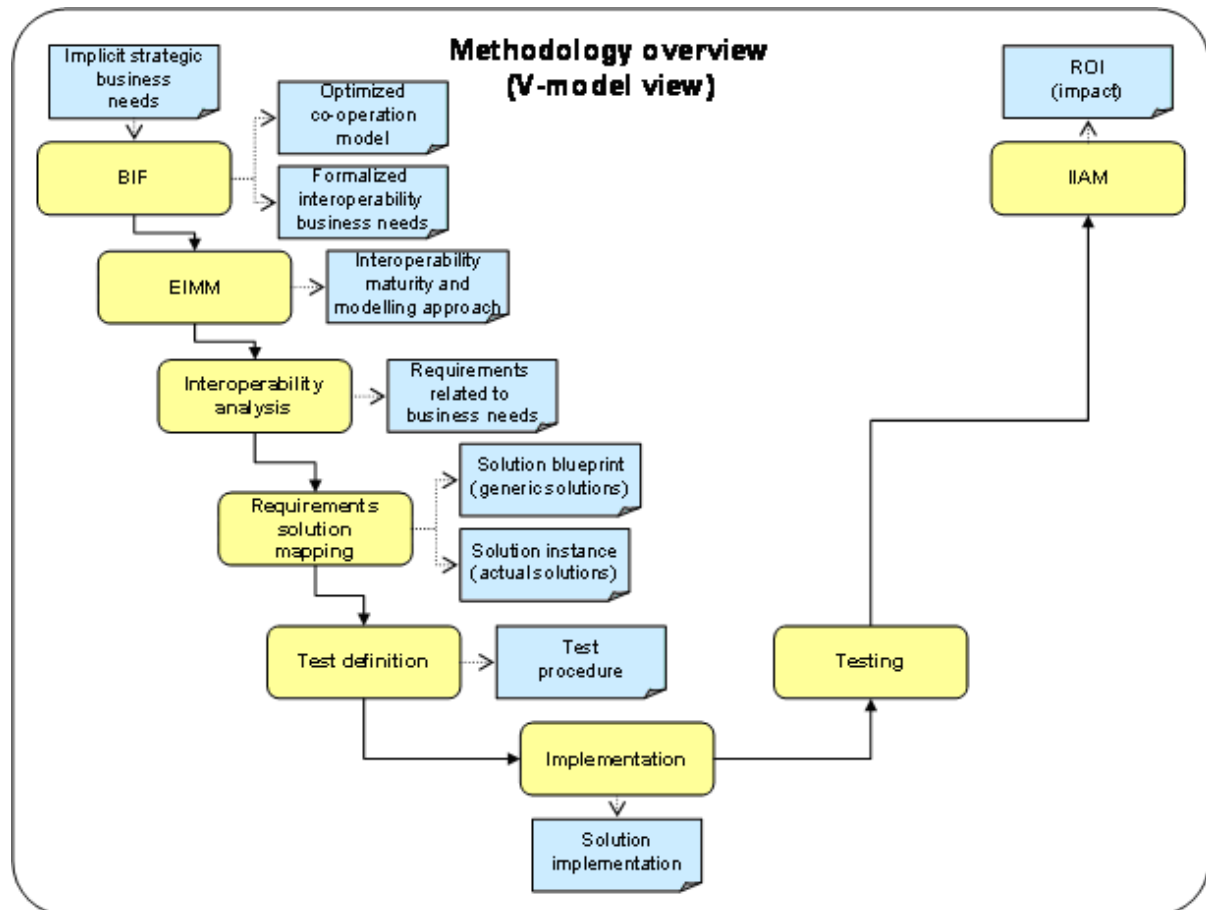
The AIM defines a baseline methodology which provides a particular integration of a set of methods developed in the ATHENA project. The baseline methodology could be used as is, or be configured and/or extended to the specific needs of the interoperability project in question.

The purpose of the methodology is to provide simple guide the users of the AIF to:

- Identify interoperability issues in their collaboration context.
- Select the appropriate ATHENA solutions and understand how to apply them.



The figure below depicts a view of the AIM according to a V-model representation.



The activities of the AIM describe the use of the following methods:

- The Business Interoperability Framework (BIF) [ATHENA B3 2006a] is a framework for determining business challenges related to interoperability according to implicit and not formalised strategic business needs. The BIF can be used to define the level of business interoperability for a given co-operation scenario. The co-operation model allows us to find optimisation potential for one collaboration and compare results with other collaborations.
- The Enterprise Interoperability Maturity Model (EIMM) [ATHENA A1 2005c] method defines a set of area of concerns and a set of maturity levels that provide the means to determine the current ability of an enterprise to collaborate with external entities and to specify the path to improve this ability. The integration matrix of the establishment methodology deduces the appropriate modelling approach for supporting analysis.
- The interoperability analysis method focus on the common understanding of the enterprise artefacts needed to achieve interoperability on the different levels. This involves understanding and relating different enterprise models, defining

cross-organisational business process models, agree on service interfaces over which the partners communicate and exchange messages.

- The requirements – solution mapping method takes as input the business needs and technical requirements identified in the interoperability analysis. The mapping methodology is helping different kinds of users to find potential ATHENA solutions and solution packages regarding their requirements. Based on annotation by contextual elements of interoperability issues (reflecting a set of specific technical requirements) and generic solutions we can support semi-automated mapping between them.
- The ATHENA Testing Framework [ATHENA B5 2005] which includes the activities test definition and testing is a framework to support conformance and interoperability testing. It describes a test architecture and how these can be combined to create a test configuration for various types of testing. It also describes the test material to be processed by this architecture, a markup language and format for representing test requirements, test cases and messages exchanged.
- The implementation activity focuses on the solution implementation. Depending on the indicated solution approach given by the requirements – solution mapping, different (and possibly alternative) implementation methods can be chosen. The implementation methods should follow a configurable and situational-based method engineering approach, where the individual method components can be characterised according to the AIF conceptual framework.
- The Interoperability Impact Analysis Model (IIAM) [ATHENA B3 2006b] method focuses on the return of investment (ROI) and the impact of the interoperability measures.

The steps of the methodology can be executed independently. This means it is possible to start not with the BIF in case of existing interoperability business needs or to select only one step in order to perform an interoperability project.

### **3.3. Business interoperability framework (BIF)**

---

#### **3.3.1. Goal**

---

Development of a framework for determining business challenges relating to interoperability. The Business Interoperability Framework can be used to define the level of business interoperability for a given cooperation scenario.

- Find optimization potential for one collaboration
- Comparison of results with other collaborations

#### **3.3.2. Description**

---

The development of the BIF is based on the assumption (1) that the maximum level of business interoperability does not necessarily represent the optimum level and (2) that business interoperability does have a direct effect on a company's performance

The BIF is structured as follows:

- A number of categories represent the fundamental concepts of business interoperability as identified in a state-of-the-art analysis.
- Each of these categories is operationalised by a set of criteria (or sub-categories) which outline the key business decisions companies have to solve when establishing interoperable electronic business relationships.
- The life-cycle aspect of the criteria is covered by the dimensions approach, deploy and assess & review.
- The interoperability levels per criteria serve as a basis for assessment and a guideline towards higher levels of interoperability.

Criteria	Definition	Life Cycle	Level of Business Interoperability						
			6 (Fully Interoperable)	4 (Qualified)	3 (Moderate)	2 (Minimum)	1 (Basic)		
Category	Management of External Relationships - "How do we manage and control external relationships?"		All strategic importance of external relationships is embedded in company strategy	Importance of external relationships is addressed in company strategy, some other selection criteria for partner selection are used	External relationships are addressed in company strategy, some other selection criteria for partner selection are used	External relationships are addressed in company strategy, some other selection criteria for partner selection are used	External relationships are addressed in company strategy, some other selection criteria for partner selection are used	External relationships are addressed in company strategy, some other selection criteria for partner selection are used	
Criteria	Plans and objectives of external relationships are documented and shared with all partners (common purpose)	Process is used in most external relationships, external relationships are actively managed	Targets are documented and shared with all partners (common purpose)	All partners are involved in external relationships	Process is used in some external relationships	Targets are specified for all partners	Targets are defined for all partners	No target setting	Each partner has individual target, which are not communicated among partners

The base for the assessment is adequate data and information about the cooperation scenario; sources can be (structured) interviews (as in case 1), questionnaires, case studies or articles (as in case 2). Finally, the result needs to be interpreted. The interpretation depends on the objective of the assessment, which could be benchmarking with other organisations or industries, or identification of potential for improvement in the design of external relationships. For the comparison of two results it is essential to consider that the contingencies influence the level of business interoperability.

### 3.3.3. Resources

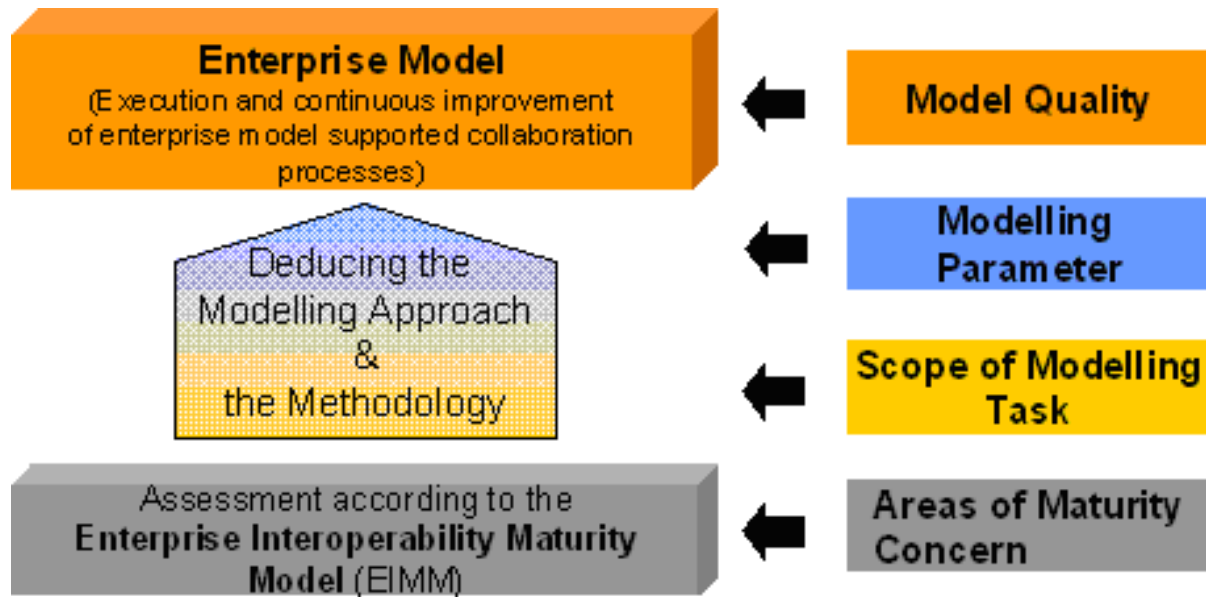
- Documentation: [Deliverable D.B3.1 - Business Interoperability Framework](#)

## 3.4. Enterprise interoperability maturity model (EIMM)

### 3.4.1. Introduction

To figure out deficits and gaps during operation leads to a serious risk related to the current business. ATHENA elaborated a maturity model and an application procedure to perform assessments for interoperability maturity.

The EIMM-based methodology step shall provide assistance in capturing the collaborative processes of the company with the support of one of several adequate modelling approaches. And further it shall support the selection of an adequate methodology into an enterprise model and establish this model in the company. Figure 38 gives a review on the whole framework [ATHENA A1 2005c]. In the following two subchapters the EIMM and the Deducing approach will be explained in more detail.



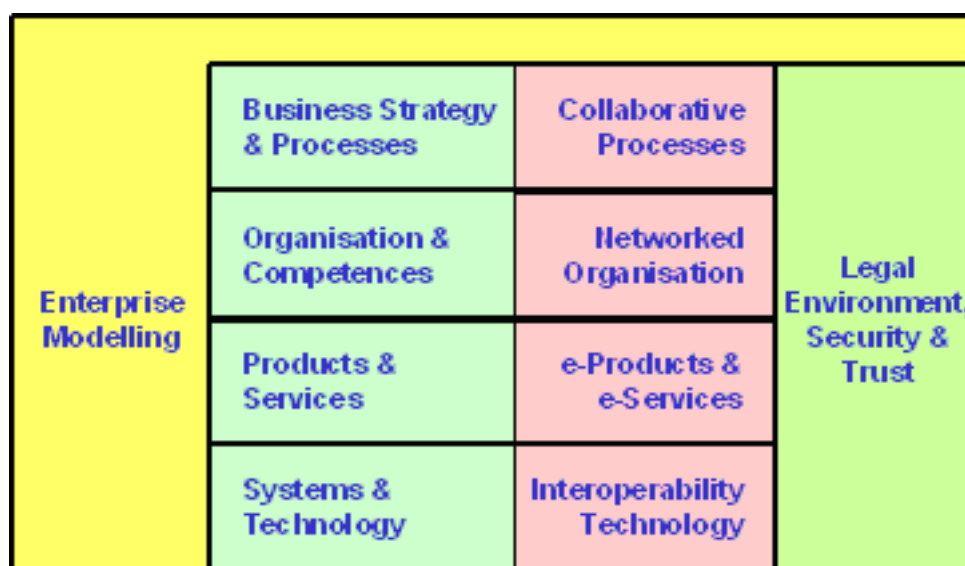
### 3.4.2. Solution - EIMM assessment

The following six Areas of Concern are captured in the assessment:

1. **Business Strategy and Processes.** This Area of Concern covers the identification, specification, execution, improvement and alignment of business strategy and processes. For the purpose of interoperability, this includes and pursues the improvement of collaborative processes, for several units within the organization as well as for external entities.
2. **Organisation and Competences.** This Area of Concern covers the identification, specification, enactment and improvement of the organizational structure, including the knowledge and skills of the identified players. For the purpose of interoperability, this includes the identification of external entities to collaborate with, the specification of the topology of a networked organization, and its deployment and improvement.
3. **Products and Services.** This Area of Concern covers the identification, specification and design of the organisation's products and services, its quality characteristics and the life-cycle strategy. For the purpose of interoperability, this includes the identification of new opportunities and specification of the same aspects for new products and services that make use of networked technologies for its delivery: e-Products and e-Services.
4. **Systems and Technology.** This Area of Concern covers the identification, specification,

design, construction/acquisition, operation, maintenance and improvement of enterprise systems. This includes the establishment of links and traceability to enterprise models, at best self-controlled. For the purpose of interoperability, this includes research and evolution of enterprise systems to apply innovative technologies that foster interoperability.

5. **Legal Environment, Security and Trust.** This Area of Concern covers the identification of legal, security and trust requirements due to the collaboration with external entities, and the provision of solutions to manage these aspects which are a key for interoperability.
6. **Enterprise Modelling.** All of the previously identified Areas of Concern are directly affected by aspects of an all embracing sixth Area of Concern. This Area of Concern covers the specification, construction, application and improvement of the enterprise models. This includes support activities such as the identification of appropriate meta-models and languages, methodologies, infrastructure, organization (people and skills), etc. for enterprise modelling. Additionally, it deals with the interoperability of enterprise models.



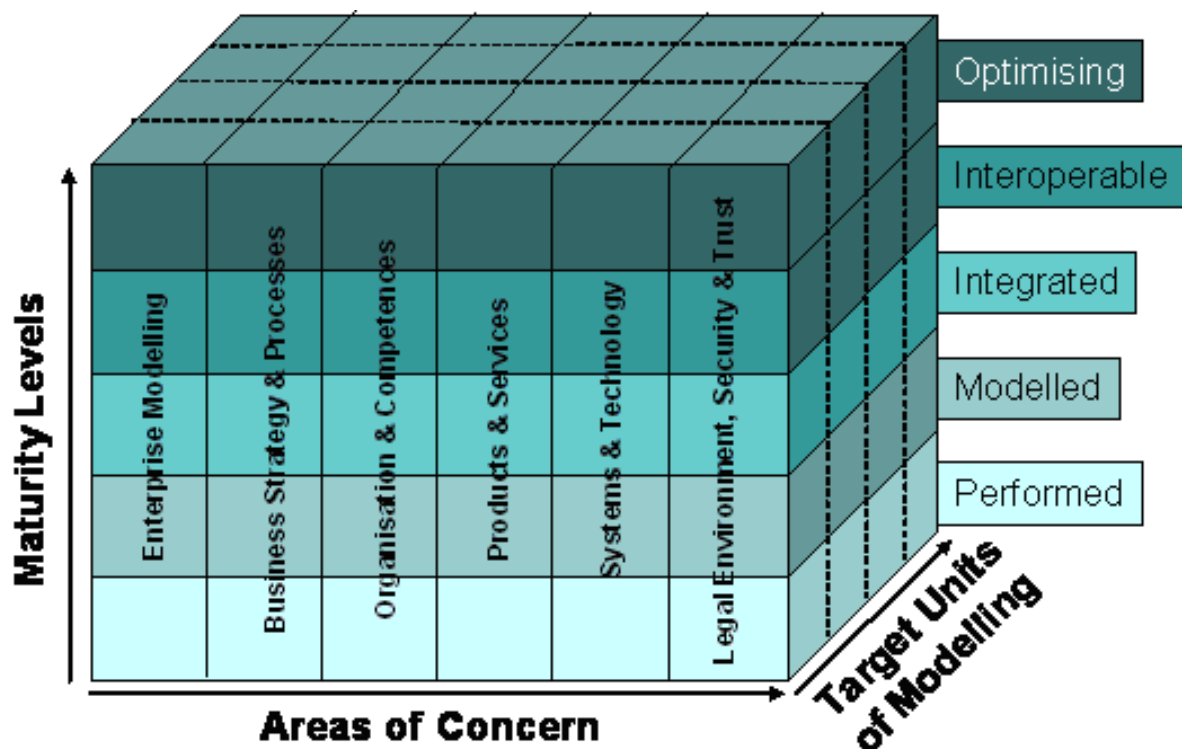
Using a five level maturity scale, the following maturity levels can be identified:

1. **Performed:** Enterprise modelling and collaboration is done, but in an ad-hoc and chaotic manner. The organization collaborates with external entities (suppliers, administration, customers), but the relationships are not planned thoughtfully. Collaborative tasks and processes usually exceed budget and schedule, their past success (usually based on the people) cannot be repeated, and the potential of the technology is not used properly.
2. **Modelled:** Enterprise modelling and collaboration is done in a similar way each time, the technique has been found applicable. Defined meta-models and approaches are applied, responsibilities are defined, people understand the enterprise model and know how to execute it, and network technologies are used to collaborate.
3. **Integrated:** The enterprise modelling process has been formally documented,

communicated and is consistently in use. The organisation uses a defined methodology and infrastructure for enterprise modelling, the different dimensions are integrated among themselves and the model is traceable to the enterprise systems, there is a knowledge base used to improve the models, and business collaboration is facilitated through interoperability technologies, use of standards, and externalisation of part of the enterprise models.

4. **Interoperable:** Enterprise models support dynamic interoperability and adaptation to changes and evolution of external entities. The workplaces of the people are seamlessly adapted to the enterprise model. Results (for organizations and persons involved) and process metrics are defined as a basis for continuous improvement.
5. **Optimising:** Enterprise models allow the organisation to react and adapt to changes in the business environment in an agile, flexible and responsive manner. Enterprise systems are systematically traced to enterprise models and innovative technologies are continuously researched and applied to improve interoperability. The use of enterprise modelling can contribute to reach the overall goals of the organization, unit, or persons involved.

The EIMM defined as a set of Areas of Concern and a set of maturity levels provides the means to determine the current ability of an enterprise to collaborate with external entities and to specify the path to improve this ability. It provides an organisational context for more specific and technical improvements. As a third dimension, the EIMM takes into account the targeted organisational units for which a maturity level needs to be assessed, or which need to be improved, in order to achieve a certain maturity level.



Each Area of Concern will be defined by a set of goals and objectives related to interoperability and collaboration issues. The level of interoperability and collaboration maturity for each Area of Concern will be defined by the presence or absence of maturity indicators. These are typical practices and work-documents, which have to be in place to achieve a determined maturity level. The specific goals and objectives of each Area of Concern, together with their indicators are described in the next section. In order to achieve a certain maturity level, each of the indicators needs to fulfil the threshold values or states that are specified for the respective maturity level. At the same time they illustrate the To-Be status that has to be realized if a company wants to reach the next maturity level.

#### **3.4.2.1. Solution modelling concept derivation**

---

The impact and the benefit of the above described criteria to Interoperability requirements can be shown, if they were mapped to the different levels of the AIF. This mapping of the criteria to the AIF gives the assessment structure and the related procedure a new tool to differentiate and to weight the interoperability requirements for Enterprise Models. In the next three subchapters the Interoperability levels and the quality criteria and the mapping with the EIMM levels will be introduced.

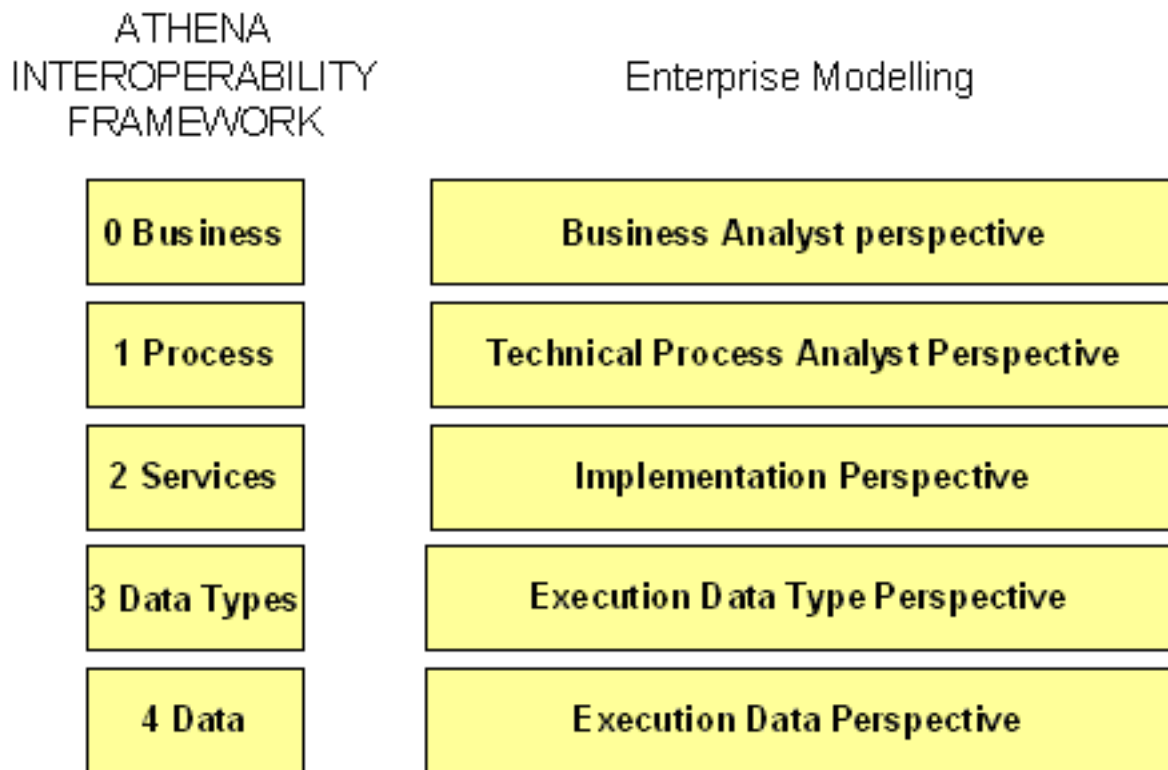
#### **3.4.2.2. Modelling levels mapped to the AIF**

---

The ATHENA Interoperability Reference Architecture as described in Figure 9 is now mapped with modelling levels in terms of formalisation. This is one key aspect in order to find the right modelling concept. This implies the following mapping items (see figure below):

- Technical Process Analyst Perspective: Collaborations on this level are characterized by the attempt of the partners to align their process with each other. The detailed business logic and the requirements for IT – Support to enable interoperability between business partners can be assessed in this level.
- In the third level the Implementation Perspective allows the invocation of existing services automatically. Collaboration can now take place on IT system level by using certain interaction protocols.
- The lowest level of granularity in performing design time modelling is represented by the Data Perspective where data formats and semantics are clarified in order to allow collaboration support with approved data and formats.
- The lowest level in terms of interoperability run-time perspective is represented by the Execution Data Perspective, where values of properties are consistent and comparable.





### 3.4.2.3. Quality criteria for enterprise modelling regarding interoperability

Quality Criteria for Enterprise Modelling regarding interoperability are derived from “Principles of methodical modelling” [Becker, et al. 1995, ISO 1998] and concepts and rules for Enterprise Models [Jochem 2001]. These basic criteria are extended and adapted regarding interoperability:

- **Correctness:** An Enterprise Model is correct, if real world elements are correctly represented in the model. It means syntactically (complete and consistent related to the Meta-Model) and semantically (structural, hierarchical and behavioural constancy related to the elements of the real world) correct.
- **Scope and Purpose orientation:** An Enterprise Model is scope and purpose oriented, if it represents only these parts of the real world which are intended by the goals, the scope and the purpose of the modeller.
- **Efficiency:** An Enterprise Model is efficient, if the creation effort is low, but the benefit regarding the intended goals, scope and purpose is high. It is also efficient when the usage duration of the model is long and itself or parts of it are reusable for other goals, scopes and purposes.
- **Conformity:** An Enterprise Model is conform, if it fulfils specific modelling language requirements, follows specific (design) rules, fulfils/covers standards, covers specific boundary conditions.



- **Clearness:** An Enterprise Model is “clear”, if on one side a common well known terminology based on an application-oriented ontology is used and on the other side it is readable based on a structured layout. This criterion depends on the model user and also on the modelling method/language which is used.
- **Comparability:** An Enterprise Model is comparable, if it fits into a common framework, uses defined levels of abstraction and a granularity based on defined scope, goals and purpose. Comparability is influenced by the use of common patterns, the grade of formalisation and the correct usage of modelling method/language.
- **Systematic Structure:** An Enterprise Model has a systematic structure, if it fits into a common framework, uses common pattern, was build with consistent, systematic applied design rules and supports the concepts of views to integrate models developed from different views.
- **Life-Cycle Support:** An Enterprise Model supports the Enterprise Life-Cycle, if it allows feeding model information forward and backward in life-cycle activities and represents recursion and iteration mechanisms. Different life-cycle phases may have different models. It enables value-added iteration of enterprise processes that improves product quality.

#### **3.4.2.4. Mapping**

---

In. the mapping between the introduced parameters is shown: Scoping Business Modelling (which are the leading parameters), Modelling parameters, required minimum EIMM Assessment result and importance of the Modelling Quality parameters.

<b>Quality Criteria</b>	Correctness	Important	Important	Important	Important	Important
	Scope and Purpose orientation	Essential	Essential	Important	Less important	Essential
	Efficiency	Less important	Less important	Less important	Important	Important
	Conformity	Important	Important	Important	Essential	Essential
	Clearness	Essential	Important	Important	Important	Important
	Comparability	Important	Essential	Essential	Essential	Essential
	Systematic Structure	Important	Essential	Essential	Essential	Essential
	Life-Cycle Support	Essential	Essential	Important	Less important	Important
<b>Modelling Parameters</b>	<b>Model granularity</b>	<b>Value Chain</b>	<b>Work Process</b>	<b>Activity</b>	<b>Properties</b>	<b>Property Values</b>
	<b>Model Completeness</b>	<b>Pragmatics</b>	<b>Syntax</b>	<b>Semantics Constructs</b>	<b>Semantics Data</b>	<b>Semantics Data</b>
	<b>Level of Formalization</b>	<b>Business Analyst Perspective</b>	<b>Technical Business Process Perspective</b>	<b>Implementation Perspective</b>	<b>Execution Data Type Perspective</b>	<b>Execution Data Perspective</b>
<b>Scoping Business Modelling</b>	Management level					
	Process level					
	Process Support level					
	IT-Execution Level (Datatype)					
	IT-Execution Data Collection and Analysis					
<b>Minimum EIMM Assessment</b>	Products & Services	<b>Level 1 Performed</b>	<b>Level 2 Modelled</b>	<b>Level 3 Integrated</b>	<b>Level 4 Interoperable</b>	<b>Level 5 Optimized</b>
	Enterprise Modelling					
	Legal Environment, Security and Trust					
	System & Technology					
	Business Strategy and Processes					
	Organisation and Competences					

The AIF Reference Architecture levels are represented as Level of Formalisation from Business Analyst perspective to Execution Data perspective. Based on the “Business Scope” the right modelling parameters can be derived in order to define an appropriate model (see mark “X” to each level). As well the required EIMM level is indicated in the same metric. In the case that an EIMM level is not achieved for a distinct modelling task, activities for the improvement of interoperability capabilities can be identified by a simple analysis of the current maturity profile. The quality parameter which represents the outcome of the modelling task has a different behaviour. The requirements level of each parameter is increasing from left to right. So for instance becomes the “Clearness” in the Execution Data Perspective the mark important whilst in the other levels it is essential.

### 3.4.2.5. Application guide

In the following two subchapters the application for applying the EIMM and to perform the deducing approach will be figured out.

### Application guide for EIMM

---

In the following the steps for performing the assessment are figured out.

**First:** The situation of the company has to be clarified:

- Assessment is important for the reputation to the business community – here an independent assessor from outside is necessary. For an outside person the major specific keys for the business and the situation of the company has to be identified and if necessary the impact for interoperability has to be clarified
- Internal assessment for improving the capability or deducing mainly the right modelling approach – the evaluation of the assessment should be done by an independent person inside the company

**Second:** A self assessment of each unit which is responsible for a certain item has to be performed. Here both a member of the management and a staff member have to fulfil the questionnaire independently. Especially for function oriented organisations (Sales, production planning, Production etc.) a distribution in bigger companies is recommended. In smaller companies the ORG/IT Department should be able to fulfil the questionnaire. In the following a reference proposal should help to distribute.

- Product /Service Development
- Production Planning
- Production
- Sales
- Purchasing
- HR
- Org/IT
- CEO level

The responsible organisational unit is normally marked with “x”. There are some alternatives for distributing. In this case an “a” means that the ticked organisational unit can answer as an alternative. Sometimes if one organisational unit is not able to have the respected knowledge (e.g. Org/IT) more than one organisational units should in parallel “mandatory” answer. This is marked with “m” (see example in the table below).

EM1	Bill of Material	x						a	
EM2	Product variants	x						a	
EM3	Structure	x						a	

EM4	Service Processes							x	
EM5	Others							x	
EM6	Strategic processes (like governanc							x	

**Third:** After the self assessment the independent person should analyse the fulfilled questionnaires in order to identify discrepancies. During the external audit these discrepancies as well between management and staff have to be moderated and verified by using operational data or reference documents. Depending on the size of the company more than one auditor may have to assess.

**Fourth:** The independent person has to perform the entire calculation and to organise a final evaluation meeting at least with the management in order to clarify the rating and to give recommendations for the entire company.

#### Application for the mapping

Based on the Scope elaborated by using the Business Interoperability Framework and the current maturity Level of Interoperability the to-Be and the As-Is columns can be derived. The level of formalization is the channel for all other modelling concept aspects like completeness, granularity and quality. These aspects will guide the modelling as well the derivation of interoperability requirements.

<b>Quality Criteria</b>	Correctness	Important	Important	Important	Important	Important
	Scope and Purpose orientation	Essential	Essential	Important	Less important	Essential
	Efficiency	Less important	Less important	Less important	Important	Important
	Conformity	Important	Important	Important	Essential	Essential
	Clearness	Essential	Important	Important	Important	Important
	Comparability	Important	Essential	Essential	Essential	Essential
	Systematic Structure	Important	Essential	Essential	Essential	Essential
	Life-Cycle Support	Essential	Essential	Important	Less important	Important
<b>Modelling Parameters</b>	<b>Model granularity</b>	<b>Value Chain</b>	<b>Work Process</b>	<b>Activity</b>	<b>Properties</b>	<b>Property Values</b>
	<b>Model Completeness</b>	<b>Pragmatics</b>	<b>Syntax</b>	<b>Semantics Constructs</b>	<b>Semantics Data</b>	<b>Semantics Data</b>
	<b>Level of Formalization</b>	<b>Business Analyst Perspective</b>	<b>Technical Business Process Perspective</b>	<b>Implementation Perspective</b>	<b>Execution Data Type Perspective</b>	<b>Execution Data Perspective</b>
<b>Scoping Business Modelling</b>	Management level	x				
	Process level					
	Process Support level					
	IT-Execution Level (Datatype)				x	
	IT-Execution Data Collection and Analysis					x
<b>Minimum EIMM Assessment</b>	Products & Services	<b>Level 1 Performed</b>	<b>Level 2 Modelled</b>	<b>Level 3 Integrated</b>	<b>Level 4 Interoperable</b>	<b>Level 5 Optimized</b>
	Enterprise Modelling					
	Legal Environment, Security and Trust					
	System & Technology					
	Business Strategy and Processes					
	Organisation and Competences					
		<b>AS-IS</b>	<b>TO-BE</b>			

In the next subchapter the modelling concept will be used to perform interoperability analysis by annotating requirements based on enterprise models.

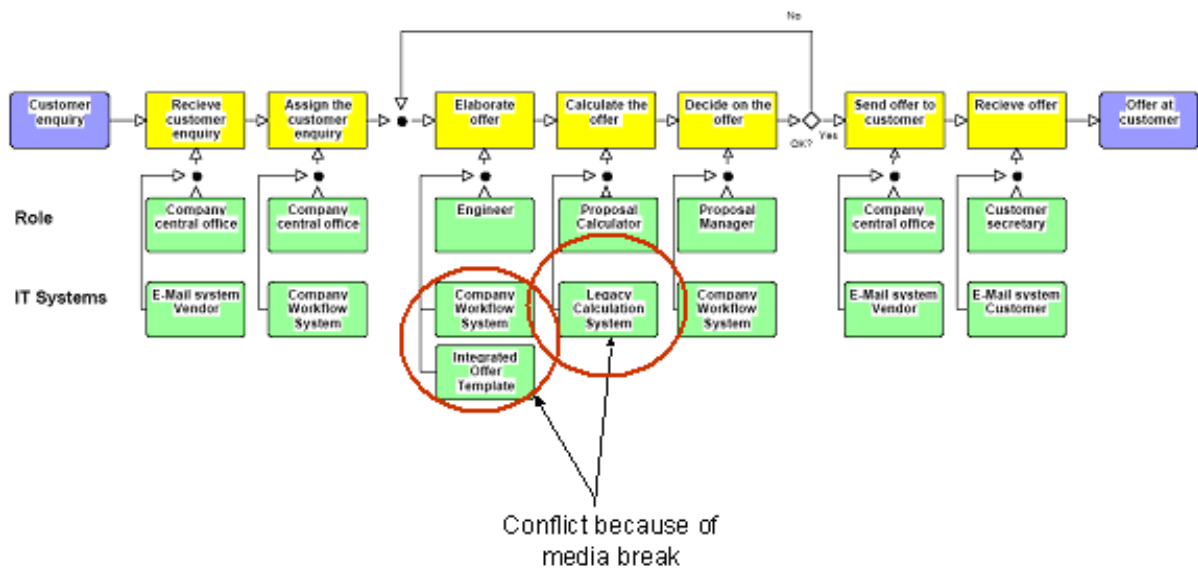
### 3.4.3. Resources

- Documentation: [Deliverable D.A1.4.1 - Framework for the Establishment and Management Methodology](#)

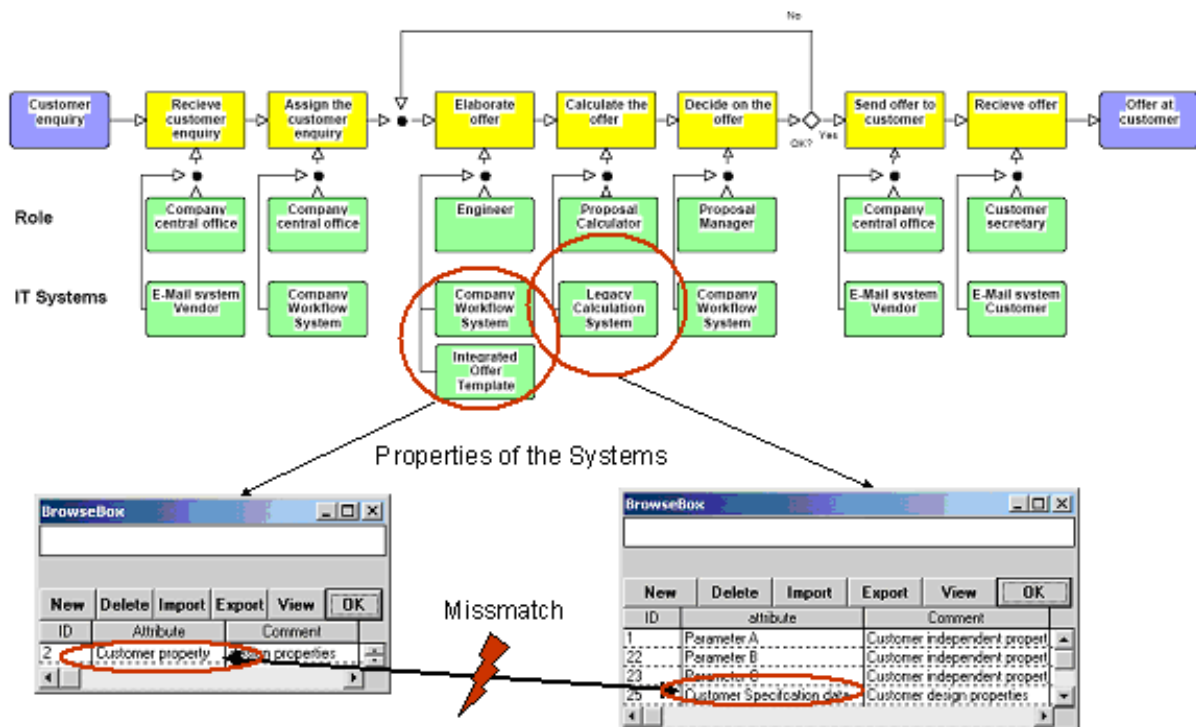
## 3.5. Interoperability analysis

### 3.5.1. Interoperability analysis

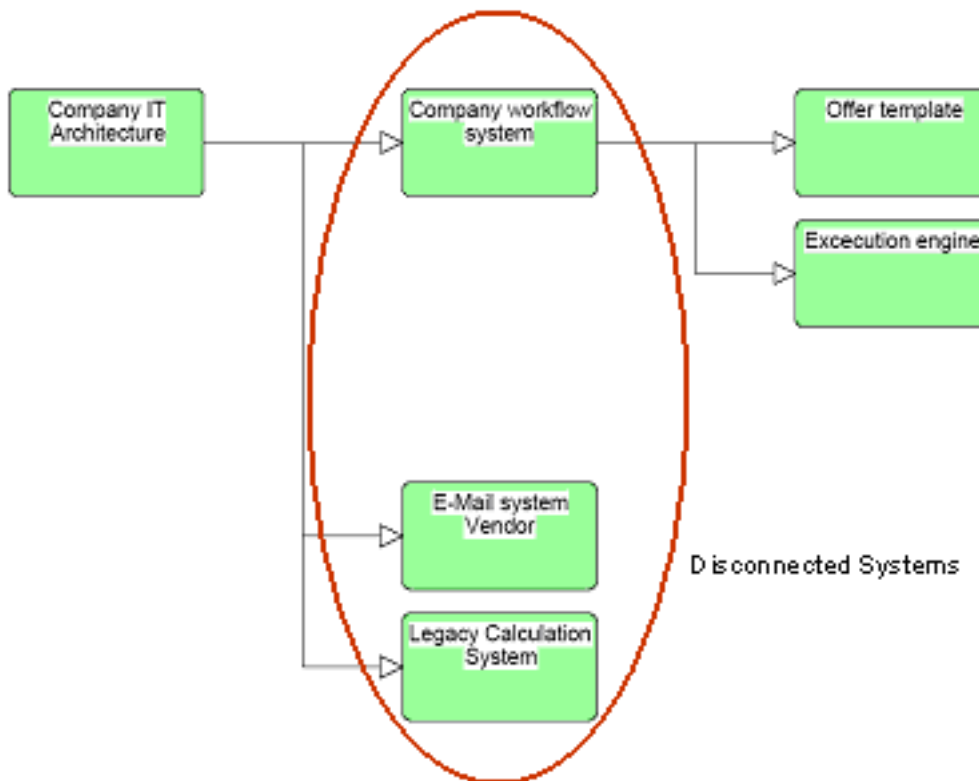
The EIMM based approach for Enterprise Modelling can be used for the derivation of an adequate elaboration of an Enterprise Model in order to determine interoperability requirements in detail. The following explanations are based on a small IEM model that addresses today not integrated legacy systems for enquiry processing in a company. So in the process view media and organisational breaks can be illustrated.



The process indicates also the relevant roles in the enterprises that are involved into the process execution. Information breaks can also be analysed in detail by showing the properties of the currently not interoperable systems.



In order to analyse the enterprise objects several views and diagrams can be used. The views should be integrated to each other in order to ensure consistent relations. As example in the next figure the IT Architecture of the company is sketched as part-of-structure. Here all disconnected systems can be identified in order to follow a holistic approach to integrate different IT Systems. Especially for business analysts the relation between all views of the enterprise model should be possible. For that also tables and textual descriptions generated from a model will be helpful.



The models should be used during all other phases described in Scenario-Specific Requirements and Generic Requirements, and Validation of Solutions according to the requirements. Here business consequences of solutions that fit to the requirements can be identified and calculated. For example, process the time reduction by reducing the information gaps or in this case by reducing the calculation activity through automatically calculation as well cost reduction or new business organisation can be simulated.

To summarise enterprise models will help in the step to perform the detailed analysis for a given interoperability situation. Here the input coming from BIF and EIMM are very helpful but not every time necessary to elaborate the requirements in an efficient way.

### 3.6. Requirements - solution mapping

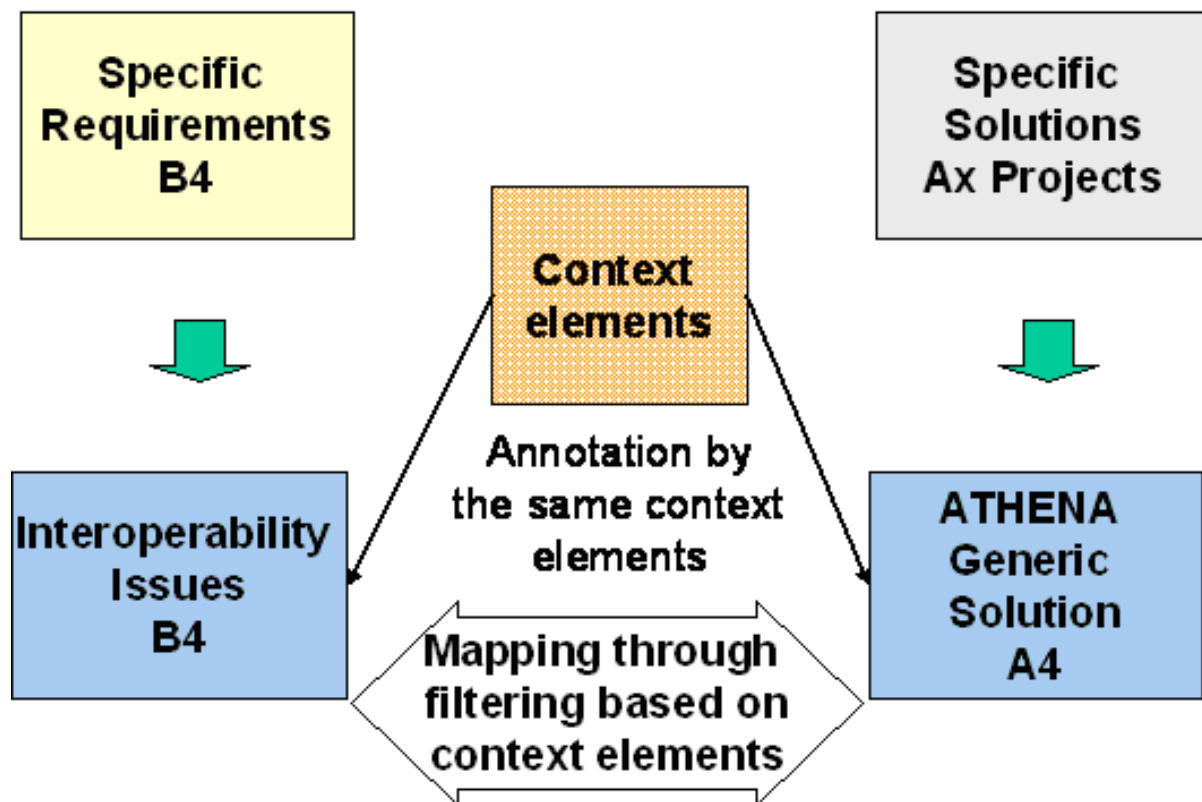
#### 3.6.1. Introduction

The objective of the “Requirements – Solution mapping is to find correspondences between solutions and requirements in order to compare expectations and requested solutions. In order to do this, similar conditions are necessary according to the interoperability context coming from different sources like role of organisation, business process type etc. For sure an 1:1 mapping is due to the fact of diversity on both sides not possible. Based on more than 600



ATHENA requirements, the mapping can be done by linking business needs and solutions. In this context requirements play the role for deeper specification of the industrial or scientific problem in order to support implementation. The approach is to map Business Needs (which are 1:1 correspondent to the interoperability issues) to solutions by using common context elements. This means if a business need has a similar context to a given generic interoperability solution, specific solutions can be selected in order being implemented by specific (ATHENA) solutions. Generic solutions can be seen as interoperability functions which can be combined for solving a more or less complex interoperability problem (challenge) as stated in business needs.

The common context plays the glue between both generic solutions and business needs. By having introduced this middle layer specific solutions and specific requirements can be related as well.



### 3.6.2. Annotation of business needs and solution

The contextualisation consist on eight elements

- AIF – Railroad levels: Framework which relates the ATHENA Solutions to interoperability levels: business, process, service, data see D.A4.2
- AREA of Concern: regarding to the EIMM levels (see chapter before)

- **Business Process Relation:** With this element we define whether a business need is specific to the business case under study (Product Portfolio Management, e-Procurement etc) or not.
- **Collaboration Type:** This element gives an idea in which organisational relations business partners are – from simple buyer and seller to virtual enterprises.
- **Interoperability life cycle step** as proposed by A4 (sub groups of run and design time) and the issues between the steps
  - **Design Time:** Analysis, Negotiation
  - **Run Time:** Realisation and Operation
- **MDA artifact:** Here the MDA levels from Computational Independent Model (CIM) down to Platform independent model (PIM), Platform Specific Model (PSM) and Source Code are considered.
- **Quality Condition:** This element is used for identifying whether the business need or a solutions addresses:
  - Quality Improvement
  - Time reduction
  - Cost reduction
  - Increase Flexibility
- **User Perspective:** is important to identify the main stakeholder of an interoperability requester and the related solution. Here the 10 types of Roles are selected which are identified by the B6 project:
  - **Business related Role**
    - Business Manager
    - Business Process Analyst
    - IT Manager
  - **IT System related Role**
    - System Architect
    - System Designer
    - System Responsible
    - Developer
  - **IT Application oriented Role**
    - Unit Manager
    - Performance Engineer
    - Software User

The Generic Solutions that were identified are clustered into four groups derived from the AIF framework and the requested solutions by B5:

- **Model related Solutions**
  - Create Model – to enable model elaboration
  - Execute Model (transform data) – solution to use a model for (automatic) data transformation

- Transform Model horizontally between different application on the same AIF railroad level (on all levels: Business to Business: Process to Process: Services to Service : Data to Data).
- Transform Model vertically between the different AIF railroad levels (from business to data or vice versa)
- Enrich models by additional information in order to improve capabilities
- Create compatible views of models in order to allow comparison between different systems which are reflected by the models
- Mapping of data in models to link data to models
- SW Component related solution
  - Searching (e.g. for a software service)
  - Selecting (e.g. by using profiles or conditions and criteria's as well in a runtime environment)
  - Invocation – into existing systems (e.g. in a runtime environment)
- Analysis and Testing
  - Assessment of the state of the art of a given system as well against a to be profile
  - Conformance test – in runtime against a given specification
  - Logic test – as well conceptually
  - Performance test – by using given parameters
  - Search for content – based on given parameters
- Connectivity
  - Naming - semantic
  - Provide Connection - physically
  - Routing (messages and models) - logically

As it has been already mentioned, the generic solutions can be seen as interoperability functions which can be combined into a holistic scenario. This means that a given “Business Need” can require several interoperability functions. For example the business need “Derive from process structure of a PLM application the business logic and compare this with other PLM application and their procedural structure” requires two different interoperability functions as can be seen in the figure below.

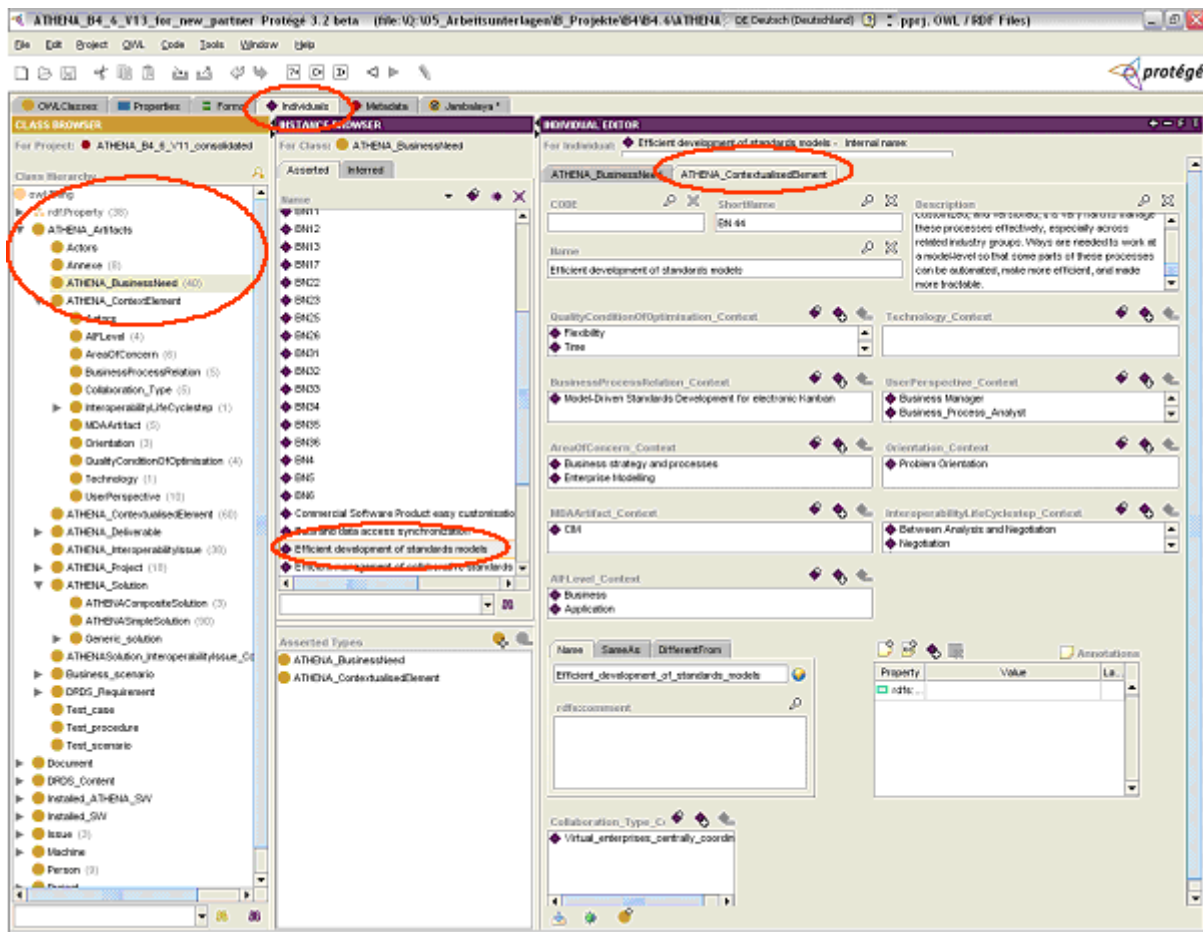


### 3.6.3. Mapping application by using the Protégé toolset

---

The contextualization of business needs and generic solutions is done by using the Protégé toolset [Stanford Medical Informatics]. Protégé was used to federate information coming from the different data repositories developed by ATHENA within a single knowledge base, called the harmonization model, which contains the different classification used within the ATHENA project. The goal was to enable mapping formalisation between specific requirements, generic requirements, generic solutions and specific solutions. In addition, advanced querying and visualization tools allowed easily analysing this mapping without any development. See [Figay] for more information about the knowledge base and how to perform queries.

In the figure below the contextualization of a business need in Protégé can be seen. The context elements are modelled as classes. Instances of the classes are the possible values. E.g. “Context Element AIF\_Level” is a subclass of “ATHENA\_Context Element”, CIM is an instance of this Class and can be used for annotating a given Business Need (here “Efficient development of a standard model”). The annotation is possible because each “Business Need” and each “Generic Solution” has the Class: “Contextualized Element” as additional Parent Class as can be seen as “Asserted Types”.



The same contextualization is done with the generic solutions as well with the related specific solutions coming from the ATHENA research and development activities but only regarding the MDA and AIF Levels. On both items an annotation is possible from Generic and Specific Solutions because, a differentiation is in some cases not applicable.

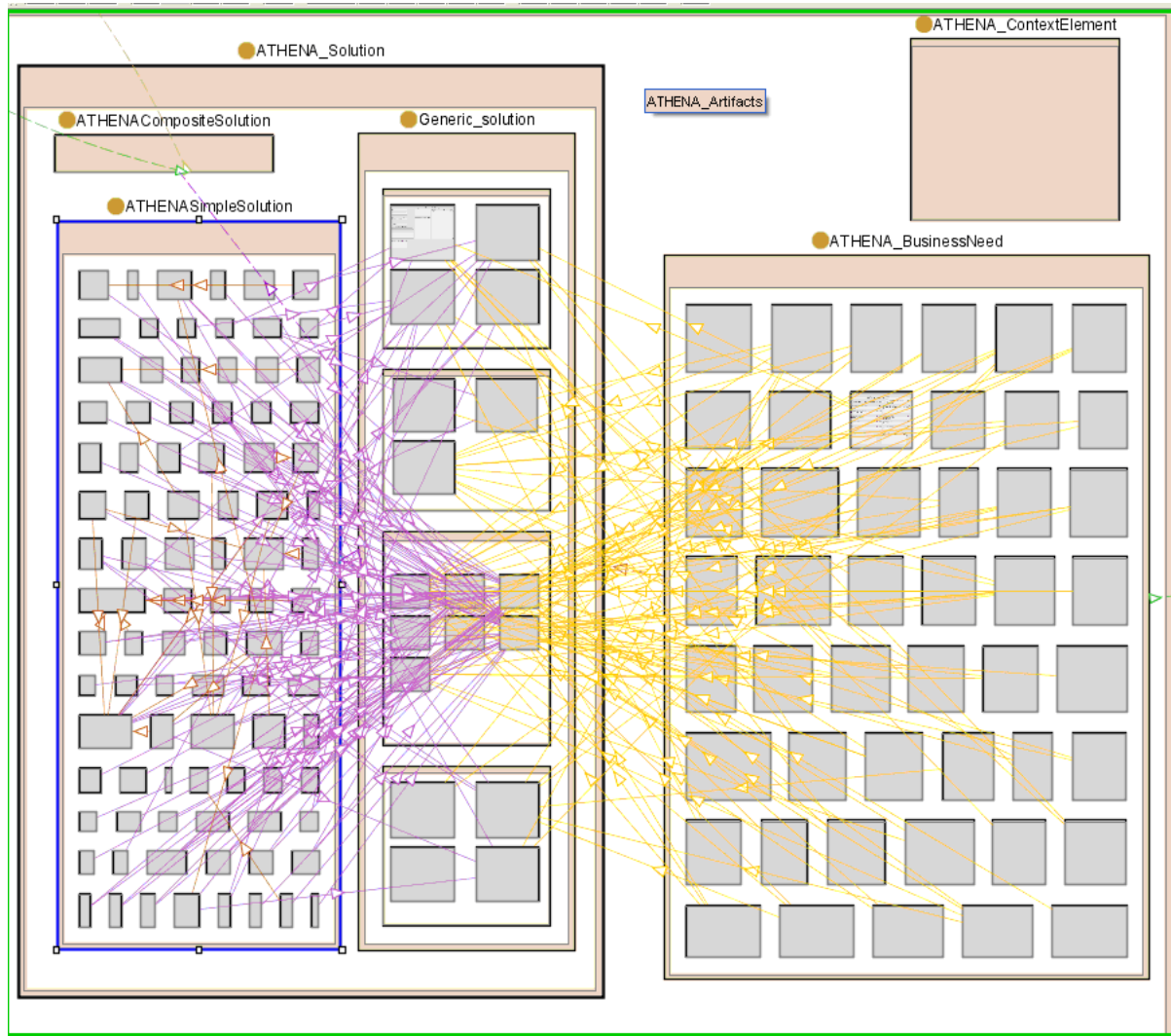
### 3.6.4. Analysis

Protégé is now used for mapping “Business Needs” and the “ATHENA Specific Solutions” based on the Generic Solutions and specifically related to the context elements as figured out in the figure below. Protégé provide two alternatives for analysis:

- Graphical analysis based on the Jambalaya plugin
- Queries for filtering and combining context elements and groups of solutions and “Business Needs”

The graphical analysis gives a very short overview about gaps between “Business Needs” related to the pilots and the required solutions coming from ATHENA.

By applying filtering, the relevant major items for mapping can be highlighted. These are: Generic Solutions as glue for combining “Business Needs” and “Simple Solutions” as well for detailed analysis the link to the “Context Elements and “Composed Solutions”. By expanding the container the relationships between the items can be identified and analysed.



Of course the first view seems to be very complex, but by zooming and filtering gaps and related issues can be pointed out very fast. So for instance clusters of a proliferation of solutions can be identified as well not used solution parts like the “Generic Solution”: “Search for SW Component” is neither implemented as a “Simple Solution” nor requested by a “Business Need”.

The Protégé 3.2 OWL plug-in includes a SPARQL query panel that allows performing SPARQL queries. SPARQL is a query language for getting information from RDF graphs. It

provides facilities to:

- Extract information in the form of URIs, blank nodes, plain and typed literals.
- Extract RDF subgraphs.
- Construct new RDF graphs based on information in the queried graphs.

SPARQL can be used with OWL. OWL is an extension of RDF (Resource Description Framework). SPARQL is a language and set of APIs specified by W3C, and that can be consulted at <http://www.w3.org/TR/rdf-sparql-query/> and is using as well RDFS (RDF-Schema).

One example is to display matching Business needs to generic solutions through the Context Element “Area of concern” The idea here is to obtain list of business needs related to generic solution through a same area of concern. We also expect to obtain the name property of needs and generic solutions.

```
SELECT ?businessNeed ?genericSolution ?areaOfConcern
WHERE {
  ?y :Name ?genericSolution.
  ?x :Name ?businessNeed.
  ?z :Name ?areaOfConcern.
  ?y :AreaOfConcern_Context ?z.
  ?x :AreaOfConcern_Context ?z.
  ?x rdf:type :ATHENA_BusinessNeed .
  ?y rdf:type :Generic_solution .
}
```

businessNeed	genericSolution	areaOfConcern
Business strategy translation into busine...	Identification_of_desired_interoperabil...	Business strategy and processes
Ability of integrated applications through...	Identification_of_desired_interoperabil...	Systems and Technology
(Near) real-time aggregated views of ke...	Identification_of_desired_interoperabil...	Systems and Technology
Inference role objectives into attainable...	Easier_planning_of_collaborations	Business strategy and processes
Seamless communication and collaboratic...	Easier_planning_of_collaborations	Systems and Technology
Data and data access synchronization	Easier_planning_of_collaborations	Systems and Technology
Automatic Generation of Workplaces	Easier_planning_of_collaborations	Systems and Technology
Real Time Knowledge aggregation	Easier_planning_of_collaborations	Systems and Technology
Workplace Management facilities	Easier_planning_of_collaborations	Systems and Technology
Business strategy translation into busine...	Easier_planning_of_collaborations	Business strategy and processes
Ability of integrated applications through...	Easier_planning_of_collaborations	Systems and Technology
(Near) real-time aggregated views of ke...	Easier_planning_of_collaborations	Systems and Technology
Inference role objectives into attainable...	Flexible_Service_Invocation	Business strategy and processes
Seamless communication and collaboratic...	Flexible_Service_Invocation	Systems and Technology
Data and data access synchronization	Flexible_Service_Invocation	Systems and Technology
Automatic Generation of Workplaces	Flexible_Service_Invocation	Systems and Technology
Real Time Knowledge aggregation	Flexible_Service_Invocation	Systems and Technology
Workplace Management facilities	Flexible_Service_Invocation	Systems and Technology
Business strategy translation into busine...	Flexible_Service_Invocation	Business strategy and processes
Ability of integrated applications through...	Flexible_Service_Invocation	Systems and Technology
(Near) real-time aggregated views of ke...	Flexible_Service_Invocation	Systems and Technology



## 3.7. Testing methodology

---

### 3.7.1. Goals

---

The objective of this Test Framework is to support conformance and interoperability testing. It describes a test architecture and its software components, how these can be combined to create a test configuration for various types of testing. It also describes the test material to be processed by this architecture, a mark-up language and format for representing test requirements, Test Cases and messages exchanged. The main purpose of the Methodology & Technology Testing Report is to provide a methodology and an IT architecture in order to validate, by means of the piloting activities, the research results coming from the AL A projects.

### 3.7.2. Description

---

The Test Framework is composed by the following elements:

#### 1. Test Components

- **Test Driver:** The Test Driver is the component that drives the execution of each step of a Test Case. Depending on the test type, the Test Driver may drive the Test Case by interacting with other components. Therefore the primary function of the Test Driver is to parse and interpret the Test Case definitions that are part of a Test Suite, as described in the Test Framework mark-up language.
- **Test Service:** it is the component that implements some test operations (actions) triggered by received messages. These actions support and automate the execution of Test Cases.

#### 2. Test Suite Documentation

- The Test Suite documentation is a collection of several OASIS IIC XML Schemas, with some added attributes required for the ATHENA test platform, containing configuration data, documentation and executable Test Cases description (XML).
- **Test Suite Metadata** provides documentation used by the Test Driver to generate a Test Report for all executed Test Cases.

#### 3. Test Platform

- From a high level description we can consider the composition of two macro blocks. The first one is the component under test (system, application or service that is being tested). The second block is included in the ATHENA Conformance/Interoperability testing platform that is the component from which the test execution is conducted.
- It is easy to recognize that the ATHENA Conformance/Interoperability testing platform plays the role of Test Driver previously described while the System under

test the one of the Test Service.

### 3.7.3. Activities

---

With the Test Framework described above it is possible to perform the Conformance and Interoperability Test. The Test Framework is intended for the following modes of operation, whether testing for conformance or for interoperability. In order for a testing process (or validation process) to conform to this specification, the following phases need to be implemented:

- Test Requirements Design: a list of Test Requirements established for the tested specification.
- Test Procedure Design: in the previous methodology the formalised procedures described a sequence of steps that the operator had to perform during the test execution. In this case the sequence of operations to be performed is embedded in an XML document
- Validation Conditions: Validation criteria have to be defined for the profile or level being tested, and expressed as a general condition over the Test Report document.

## 3.8. Implementation

---

### 3.8.1. Overview

---

After having identified the generic and concrete solutions to use, and defined appropriate test procedures, the baseline methodology needs to provide guidelines for the application of the technological interoperability approaches:

- [Collaborative enterprise modelling](#)
- [Cross-organisational business processes](#)
- [Flexible execution and composition of services](#)
- [Information interoperability](#)
- [Semantics and ontologies](#)
- [Model-driven interoperability](#)

Descriptions of applicative guidelines for these approaches are further detailed in the [guidelines and best practices](#) section.

## 3.9. Interoperability impact analysis model (IIAM)

---

### 3.9.1. Goal

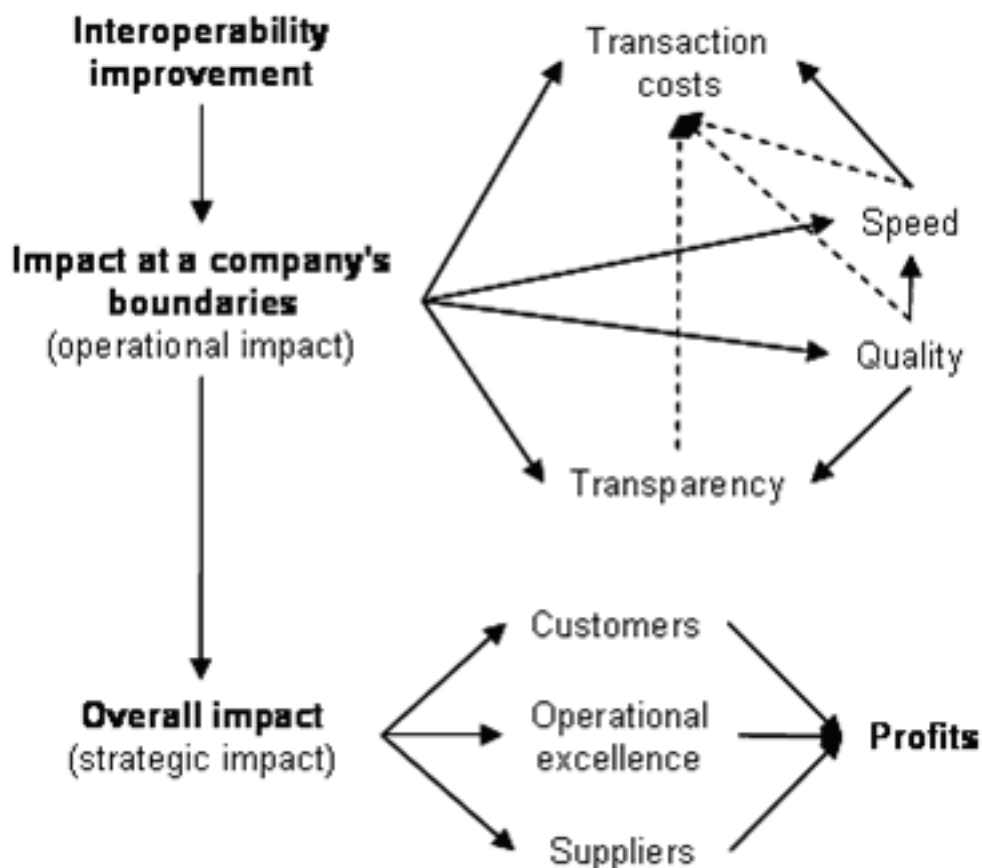
---

The aim of the Interoperability Impact Analysis Model (IIAM) [ATHENA B3 2006b] is to be a methodical framework to understand how interoperability creates value and, if possibly, quantify the benefits resulting from interoperability improvements. Together with the

Business Interoperability Framework (BIF) the IIAM can be used to assess the impact of interoperability from a business perspective.

### 3.9.2. Description

The IIAM framework builds upon transaction costs theory and causal analysis in order to identify the resulting benefits of interoperability and understand their origin. It contains a concept to integrate IT-related costs into the general transaction costs theory and a pragmatic method to operationalise transaction costs at a firm level. The impact of interoperability on businesses is further broken down into a strategic and an operational impact. Based on the case studies, we state that interoperability acts as an improvement driver at a company's boundaries (operational impacts). Nevertheless, the beneficial effects of interoperability at a company's boundaries also impact the strategic positioning of the firm. We develop therefore a method to link the direct, classical, effects of interoperability with their contribution to the achievement of a competitive strategy and identify some potential interoperability impact patterns. The figure below gives an overview of the impact analysis and its main dimensions.



The operational layer of the IIAM depicts the impacts that can be directly quantified and

provides therefore the basic figures initially fostering the investment decision. Transaction costs are broken into three quantifiable blocks: connectivity, coordination and monitoring costs, which are defined in the table below.

Transaction costs	Definition	Examples
<b>Connectivity costs</b>	nonrecurring expenses to setup or improve a business relationship	costs of negotiation costs of setting up organizational and technical connectivity (labor costs, hardware procurement, software licence fees, external consulting fees)
<b>Coordination costs</b>	costs of executing the transaction	costs of manual information processing (labor costs) costs of interacting (labor costs) infrastructure and maintenance costs (e.g. maintenance fees, communication costs) costs consequent to wrong decisions (opportunity costs)
<b>Monitoring (Control) costs</b>	costs to ensure the quality of the transaction	costs of monitoring and controlling the transaction (labor costs)

The IIAM includes a detailed questionnaire illustrating the correlations between operational key performance indicators. Given these indicators, the strategic impacts are derived (in other words their contribution to the achievement of long-term profitability). Here again, a questionnaire will enable us to understand and assess the links between interoperability actions and their consequences. These questionnaires and the application of these are further described in [ATHENA B3 2006b].

### 3.9.3. Resources

- Documentation: [Deliverable D.B3.3 - Interoperability Impact Analysis Model](#)

## 4. Guidelines and best practices

### 4.1. Guidelines and best practices

#### 4.1.1. Introduction

---

This part of the framework focuses on the guidelines and best practices and covers the following topics:

- [Collaborative enterprise modelling](#)
- [Cross-organisational business processes](#)
- [Flexible execution and composition of services](#)
- [Information interoperability](#)
- [Semantics and ontologies](#)
- [Model-driven interoperability](#)

## 4.2. Collaborative enterprise modelling

---

### 4.2.1. Introduction

---

Enterprise models have the goal to reduce the complexity and give a representation of the structure, activities, processes, information, resources, people, behaviour, and goals of an enterprise and the dependencies between them. Especially for collaboration between enterprises enterprise models are helping to understand each other, to plan, implement and to support interaction.

Today, the user of enterprise models has to deal with several problems:

1. First, too much time is needed to create a complete model, and, when finished, the developed model does not reflect the reality in a proper way anymore.
2. Second, the models often don't fit the users' requirements, e.g. the model is not detailed enough or the level of formalization is not appropriate.
3. Third, it is often not possible to use the modelling results to support the daily business of employees, because the users most of the time do not have the skills to read the models properly and to deduce the implications for their work.

Collaborative enterprises face additional problems when using the enterprise modelling and willing to interoperate seamlessly within a networked organisation. Enterprise modelling approach is different for each enterprise, depending on its current practices, systems, knowledge and culture.

### 4.2.2. Solution

---

The approach takes advantage of aspects of the Capability Maturity Model, already successfully applied to software engineering. They are applied for modelling of collaborative enterprises, in a safe and efficient mode, and independently from modelling methodologies or tools. Based on an enterprise Interoperability Maturity Model (EIMM) assessment, companies will be guided to choose the right concepts for improving their capabilities, by taking into account actual market and enterprise challenges. The approach will also be used for planning and implementing new enterprise concepts in short and mid term perspectives. Here the integration of today missing aspects like organisational capabilities and skills will

allow an easier and more sustainable application of EM.

The solution is based on three concepts:

1. **Collaboration processes and Maturity Assessment as given in the Maturity Assessment above:** The basis is given by the collaborative activities of the company (definition see in WD.A2.1.). In order to identify the correct project approach, the maturity assessment has to be performed. Using the maturity model for enterprise modelling that is described in the following section, the result of the first step is supposed to be the maturity level of the company for participating in a collaboration. The maturity level must focus on management issues as well as technological issues. Management must be aware that introducing collaborative EM technology will demand changes in their organization. It introduces an advanced form of knowledge management, and many new processes that must find responsible owners and groups of new and old categories of performers and participants.
2. **Deducing the Modelling Approach and the Methodology:** This step contains the procedure how to deduce an adequate modelling approach and methodology depending on:
  - the enterprise task
  - on the defined maturity level of the company
  - the maturity level that is needed in order to participate to the collaboration process or to improve the collaboration processes.

In this part the modelling parameters have to be specified (e.g. the right level of granularity) as well as the support level of the Model Generated Workplace have to be determined.

3. **Modelling the Enterprise and Model Generated Workplace (MGWP) application:** The result of this part is an enterprise model that follows from applying the specific modelling approach and methodology from previous part. Based on the defined model the MGWP can be generated (resp. configured). The MGWP is an application that provides a model based flexible front-end for supporting the daily business of people in different roles in the enterprise, according to the collaboration processes (e.g. operating a process or manage and control a process).

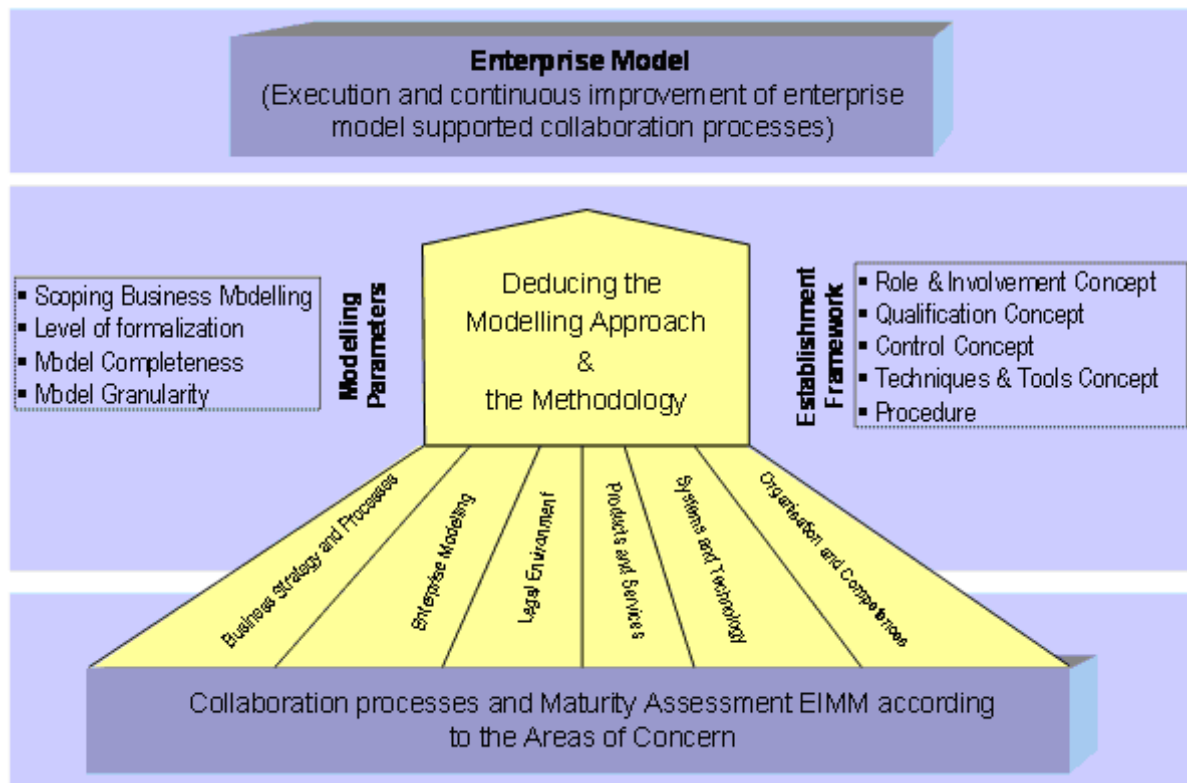


Figure: Deducing the modelling approach and the methodology

#### 4.2.3. Application

To increase the efficiency and the effectiveness of the enterprise modelling, the modelling methodology must be derived systematically from the problem definition. In general the problem areas can be classified as follows:

- Challenges coming from outside
  - Strategic (e.g. Plan new business opportunities)
  - Operational (Do transition from current situation)
  - Challenges coming from inside (EIMM Assessment)

An enterprise task is defined appropriate to these areas. This task gives requirements for the modelling. On the other hand, the general situation of the enterprise (EIMM Assessment) has a strong influence on the modelling. Both sources of impact must be considered for the definition of the modelling task. Besides, the modelling task is described with the help of the modelling parameters. The values of these parameters depend on the enterprise task as well as on the maturity of the enterprise. Afterwards based on the values of the modelling parameters and on the maturity of the enterprise the suited modelling approach for the establishment framework is customized. This procedure is represented in the figure below.

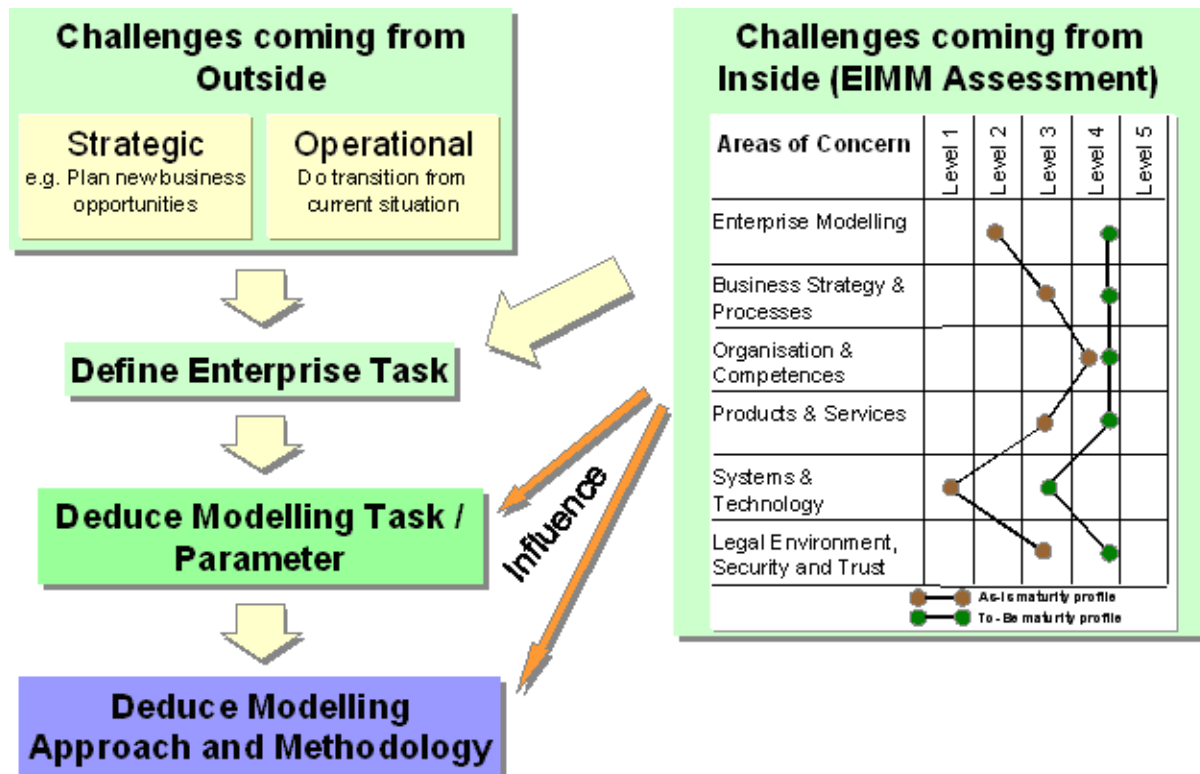


Figure: Deducing of modelling approach and methodology

### 4.3. Cross-organisational business processes - Application guidelines and design rules

#### 4.3.1. Introduction

In this chapter we first describe some general design rules that should help users to implement the architecture alternatives described in the last chapter. Furthermore, an application procedure and a CBP implementation procedure guiding the user through the necessary steps for implementing the architecture are given.

#### 4.3.2. Design rules

The design rules described in this chapter are general rules/proposals for the design of the “system” cooperation across organisations. They are not a procedure to follow during the design.

##### 4.3.2.1. Proposals for design rules for enterprise cooperation:



1. The enterprise cooperation shall have a model of the CBPs covering the workflow and the object exchange.
2. All communications between partners shall be defined within the CBP model and not hard coded in the interface components.
3. Information required for the cooperation shall not be fixed in programming code and might be configurable by a modelling approach.
4. The owner of a document shall be traceable.

#### **4.3.2.2. Proposals for design of an enterprise participating in a cooperation:**

1. Enterprises shall have a model of their view processes covering external interfaces and transformations to the internal processes.
2. Enterprises shall be able to manage local processes in a way which will allow them to coordinate data exchange with other members of the cooperation.
3. Enterprises shall be able to provide monitoring information in coherence with the CBP model.

#### **4.3.3. Application procedure**

##### **4.3.3.1. Purposes**

The purpose of this chapter is to outline the procedural steps that an organization should undertake in order to implement the CBP enactment architecture in its business environment. The objective is to highlight the relevant issues concerning the architecture implementation, and to provide guidelines on how to solve each issue using available tools and methodologies, either from ATHENA or from the state of the art.

The procedure is described abstractly, i.e., without referring to specific business scenarios and individual organization requirements. Developing detailed implementation guidelines for specific business cases is out of the scope of the present Deliverable, and will be addressed by Project A4, where implementation requirements from the pilot cases are matched with the general architecture developed in this work package.

##### **4.3.3.2. Overall Procedure**

The general implementation procedure for the CBP enactment architecture is outlined in the following Figure. The procedure consists of the following main phases:

1. **Project initialisation & specification.** This phase deals with general aspects common to most IT projects, such as strategy and requirements formulation, investment planning, activities planning, benefits estimation and performances monitoring.
2. **Modelling and Configuration.** This phase corresponds to the “design time” activities required for setting up the CBP enactment architecture. Most of these activities deal with CBP modelling and the related supporting tools. Different approaches are needed for each of the three architecture models: Integrated Engine for CBPs and Private Processes, Direct Application Integration and Private Engine.

3. **Prototype & Testing.** This phase consists of setting-up prototype cases for testing the enactment architecture. Testing considers aspects dependent on the architecture model (of the three proposed) as well as common aspects, such as CBP monitoring.
4. **Activation Phase.** This phase includes all the necessary activities to make the system available to the organization for regular usage, e.g., deployment, users training and support, roll-out across the organization.
5. **Maintenance & Change Management.** In the maintenance phase a procedure must be established to manage system failures or change requests. These have to be handled differently depending on which of the three architecture models has been implemented.
6. **Dismissal & Replacement.** This phase consists of dismissing the system at the end of its life-cycle, after an adequate alternative technology has been found to replace it. Given the level of innovation of the proposed architecture on industrial state-of-the-art, it is difficult at this stage to imagine a replacement scenario. Hence this phase will not be discussed.

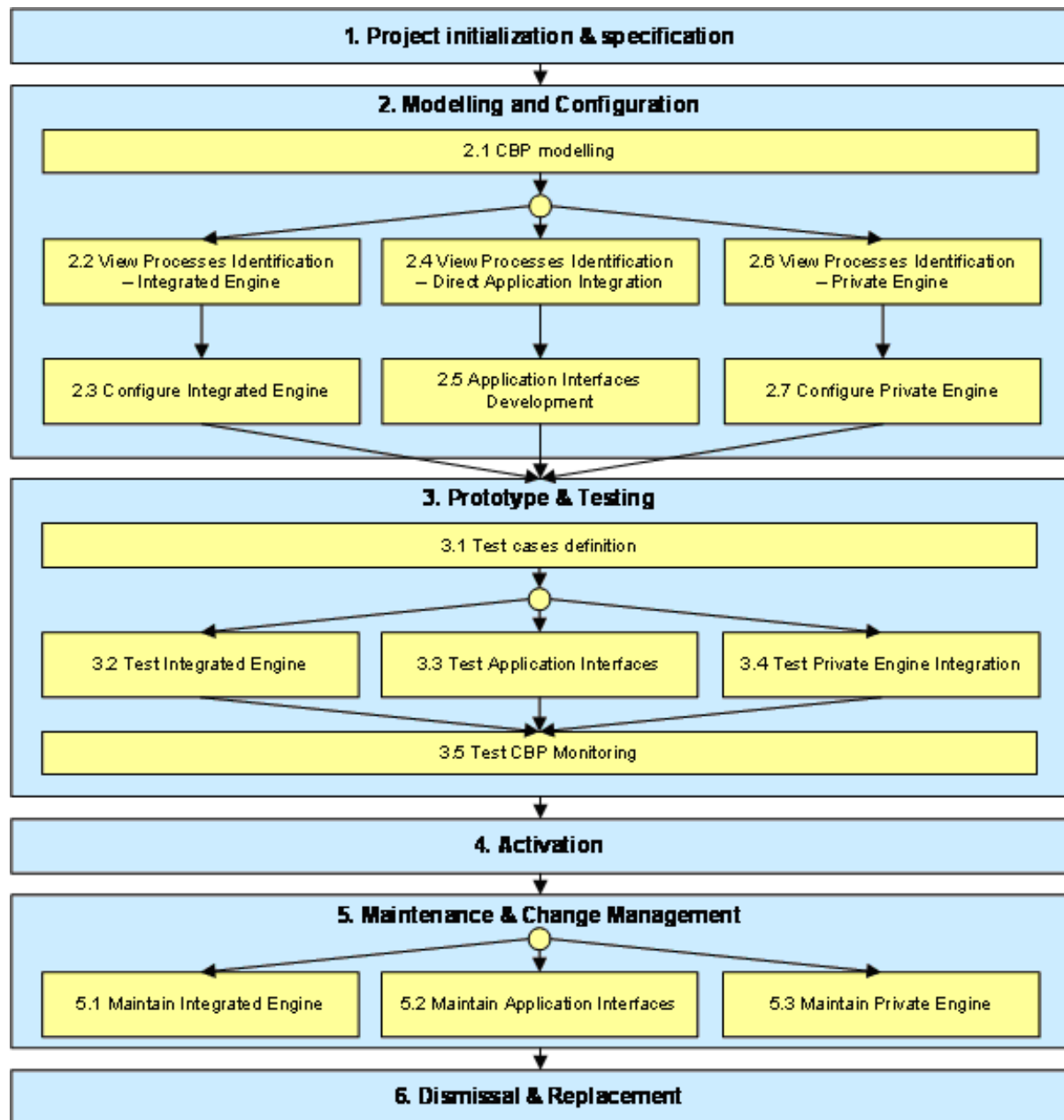


Figure: Enactment architecture application procedure

#### 4.3.4. Guidelines and relevant issues

In the following we provide guidelines for each of the phases identified above, indicating the main steps to undertake and the relevant issues and approaches to consider.

Our attention focuses solely on relevant issues from the Enactment Architecture point of view. We are aware that enabling CBPs poses important questions in other fields, e.g., about how to establish collaborative business relationships and about the content and scope of the CBPs themselves. Hence the information provided here will have to be complemented by the results of other ATHENA projects like A1 (especially A1.4), in order to develop a comprehensive CBP approach.

#### 4.3.4.1. Project initialisation & specification

---

The main architecture-related issues in this Phase are:

- Definition of an implementation plan that takes into account the various responsibilities and obligations of the different partners involved in the CBP.
- Definition of a deployment strategy, with clear identification of the physical architecture to be implemented (e.g., centralized, peer-to-peer, hybrid).
- Definition of a joint policy plan, taking into account such aspects as:
  - Service level standards;
  - Contingency planning and management;
  - Quality aspects;
  - Security aspects.
- Definition of standards to be adopted at process level and at document/event level.
- Business plan, including performance measurements, for the CBP initiative.

Support for the solution of these issues can be provided by:

- Methodological guidelines for multi-enterprise architectures implementation.
- CBP-oriented reference standards like, e.g., CPFRR and RosettaNet or other identified in work package A2.1, including contract templates and guidelines for establishing collaborative platforms.

Enterprise modelling might be an adequate instrument to support here issues such as: how IT solutions fit into the daily business or work. Starting from that enterprise modelling can be extended and used in the next phases (Modelling and Configuration).

#### 4.3.4.2. Modelling and configuration

---

##### Step 2.1: CBP modelling

---

This step consists of the definition and sharing of the Cross-organization Business Process Models, following the approach defined in the work package A2.2 and documented by Deliverable DA2.2. The mentioned Deliverable provides comprehensive guidelines and tools for these activities.

##### Steps 2.2 – 2.7

---

To design and configure the CBP enactment architecture appropriately we provide guidelines

on how to design the architecture, according to different scenarios that may be found at each partner site. In particular, rules are defined to identify which of the three architectures described in Chapter 5 is the most suitable one. Furthermore, the specific role that each component will play has to be determined.

The decisional process can be summarised with the following schema:

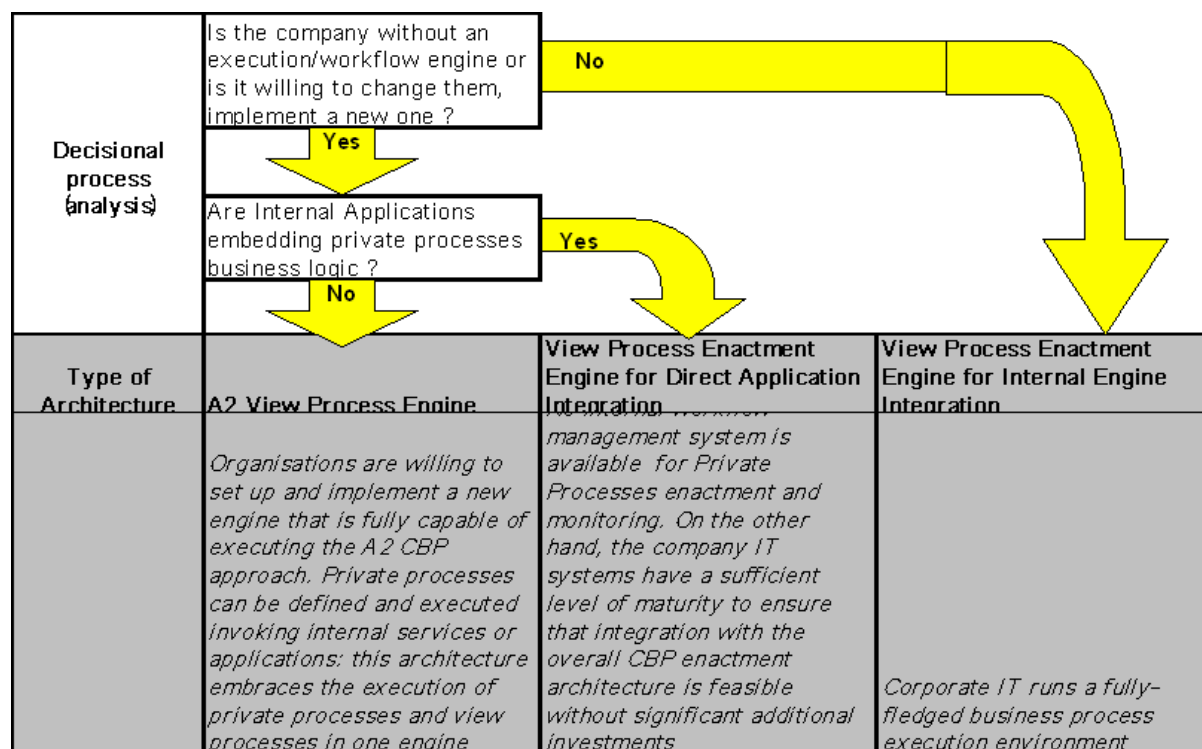


Figure: Decisional process

After having identified which of the three architectures will be targeted, guidelines will apply at different levels:

- modelling
- rules for internal components
- rules for interface components
- rules for CBP components

In the following table we have reported for each type of component and for each type of architecture, what are the most relevant issues to be taken into account for every specific case.

Moreover, according to the different situation, it may result that for each component:

- a) it is necessary to develop a component from scratch, using common guidelines
- b) it is necessary to configure an existing tool/component according to common guidelines

c) it should not be taken into account because it is not relevant or not available

According to such cases, we have painted the cells of the table with GREEN for cases a) (dark grey), with YELLOW for cases b) (light grey) or not painted for cases c).

Type of Architecture		A2 View Process Engine	View Process Enactment Engine for Direct Application Integration	View Process Enactment Engine for Internal Engine Integration
Modelling	PP	Modelling of both Public Processes and process views can be done with ATHENA A2.2 methodology & tools. Both PP and PV will be stored in the same repository.	Not relevant since PP are embedded into internal applications	PP already modelled "privately" by the partner: no assumption possible on technology/tools. Athena A2 approach not possible
	PV/CBP		PV are modelled from scratch, using ATHENA A2.2 methodology & tools	

<b>Partner Private Components</b>	<b>Private Process Modelling Tool</b>	ATHENA A2.2 tools approach	n/a	Probably already existing with specific technology, for this reason not to be taken into account
	<b>Private Process Monitoring &amp; Analysis</b>	Process monitoring facilities offered by engine developed in A2	Not relevant	Probably already existing with specific technology, for this reason not to be taken into account
	<b>Private Processes Repository</b>	see "View Processes Repository"	n/a: private processes are embedded into internal applications	Probably already existing with specific technology, for this reason not to be taken into account
	<b>Internal Engine</b>	n/a	n/a	An enactment engine running the company's private processes is already in place: it needs to be taken into account and analysed as it will need to be integrated with the Internal Application Gateway through the development of two connectors for SEND and RECEIVE. The two connectors will be provided according to a schema defined by ATHENA A2.4, and specifically implemented according to the different internal enactment engines
	<b>Internal Applications</b>	Internal applications will be directly integrated	Internal applications will be integrated with the	Internal applications are not accessible to the

	into the A2 CBP Enactment Engine. It is necessary to identify API or Web services to be further integrated with Nehemiah	Enactment Engine through API or Web Services. It is necessary to identify event-based interfaces.	Interface Components, since they are handled by the private enactment engine
--	--	---	--



<b>Interface Components</b>	<b>View Processes Repository</b>	This repository will need to store both private and public views models and instances. ATHENA A2 approach can be used	Repository to be developed according to ATHENA A2 approach, containing VP models and instances	Repository to be developed according to ATHENA A2 approach, containing VP models and instances
	<b>Internal Application Gateway</b>	Direct integration of Internal Applications through "Core Engine" of Nehemiah	The gateway must handle the enterprise applications APIs, and the standard Enactment Engine interface. It should include: <ul style="list-style-type: none"> <li>- Applications Integration Repository, which stores all the relevant information necessary to map enterprise applications interfaces with View Processes</li> <li>- Enterprise Applications Interface. On event notification from the Enactment Engine, this interface calls the corresponding API of the involved enterprise application</li> <li>- Enactment Engine Interface. Actions notified by internal applications, once translated into View Process events, are then passed on to the Enactment Engine for process advancement</li> </ul>	The gateway will communicate with the Send & Receive connectors at partner's private side. It should offer: <ul style="list-style-type: none"> <li>- listener &amp; receiver (according to connectors technology, either implemented as socket or web server, RMI, SOAP, Corba)</li> <li>- forward of business objects to the Enactment Engine</li> </ul>
	<b>Enactment Engine</b>	Use of Nehemiah	The basic set of	In this architecture,

	(ATHENA A2 CBP Enactment Engine)	functionalities that the Enactment Engine should provide is: <ul style="list-style-type: none"> <li>- View processes instantiation</li> <li>- View processes advancement</li> <li>- Events dispatching via External Gateway</li> <li>- Events dispatching via Internal Application Gateway</li> </ul>	the enactment engine will not execute views, since they are already executed by the partner's private engine. So its main functionalities should be: <ul style="list-style-type: none"> <li>- receive business objects from the Internal Gateway</li> <li>- Securely send them to the External Gateway</li> </ul>	
<b>External Partner Gateway</b>	The External Partner Gateway is implemented by the "Coalition Engine" of Nehemiah, and communication will be implemented through web services.	The basic set of functionalities that the External Partner Gateway should provide is: <ul style="list-style-type: none"> <li>- Messages dispatching to the other partners involved in the CBP</li> <li>- Receiving messages from other partners involved in the CBP</li> </ul>	The basic set of functionalities that the External Partner Gateway should provide is: <ul style="list-style-type: none"> <li>- Messages dispatching to the other partners involved in the CBP</li> <li>- Receiving messages from other partners involved in the CBP</li> </ul>	
<b>Event &amp; Document Correlation Generic Design</b>	The Event and Document correlation component must be designed in order to support: <ul style="list-style-type: none"> <li>- "message to process instance mapping", which identifies the relevant process instance</li> <li>- semantic document mapping which maps incoming messages into a format that the partner can understand. It also maps outgoing messages into a format the receiving partner can interpret appropriately.</li> </ul>			
<b>Event &amp; Document Correlation Specific Design</b>	In this architecture View Correlation IDs and Private Processes ID should be handled and mapped by the architecture	Correlation information must be associated to event types that trigger messages exchanged via the External Gateway. Messages to/from external partners must contain	View Correlation IDs are mapped to PP Correlation ID and then passed to the internal private engine	

		Correlation ID. There is no mapping of ID to/from PP and VP	
<b>CBP Components</b>	<b>CBP Repository Models</b>	A repository for both, storing the CBP definition and storing the execution data (process instances). The process definition is used for a CBP driven configuration of the engine. The information for the instance repository is derived from the execution of the view engine	
	<b>CBP Repository Instances</b>		

**Step 2.2: View processes identification – Integrated engine**

This step consists of identifying or defining the View Processes and linking them to the CBP, using an Integrated Engine which also handles the Private Processes.

The main issues for this activity are:

- Support both inside-out (from the Private Process) and outside-in (from the CBP) definition of views.
- Support pattern-based recognition of View Processes to be exposed from the existing Private Processes. CBP patterns are discussed in Section Error! Reference source not found. and in the Working Document WD.A2.1.

**Step 2.3: Configure integrated engine**

This step consists of configuring the Integrated Engine so that it can run consistently both the internal processes and the related process views.

The main issues for this activity are:

- Consistent mapping of states between internal processes and public views, and the related communications, following the approach introduced in Section Error! Reference source not found..
- Mapping of messages correlation IDs, following the approach presented in Section Error! Reference source not found.

**Step 2.4: View processes identification – Direct application integration**

This step consists of identifying or defining the View Processes and linking them to the CBP, using a View Engine which only handles View Processes.

The main issues for this activity are:

- Map internal application functions that are not necessarily process-oriented, onto CBPs in order to derive the Views that have to be exposed.

- Map View IDs, provided by the external gateway, onto internal IDs handled by the applications.

Two kinds of support are basically needed:

- Methodological support is required:
  - To map the applications functional view into a process-oriented view,
  - To define process views at a proper granularity level which is sufficient for CBPs but can also be mapped to applications.
- To handle physical communication, a high level interface should be available to expose the applications business logic and business documents. Existing state-of-the-art tools like those discussed in Section Error! Reference source not found. can provide support functionalities to facilitate the definition and maintenance of the Views/Application mapping.

#### **Step 2.5: Application interfaces development**

---

In case of a direct link to Enterprise Application, the Internal Gateway cannot remove the need for these applications to provide a programmable interface that must be made compatible with the View Process concept.

The main issues are:

- Providing an event-notification interface for all applications handling internal processes exposed as View Process, to inform the CBP about the progress of internal processes.
- Handling incoming messages for information provided by business partners via the External Gateway. This interface will also take care of ID correlation on the application side.

Such an interface can hardly be developed from scratch, and will rely on the APIs which are already present in most enterprise systems. Anyway, adaptations will be needed even for modern, web-service based interfaces, to meet the Internal Gateway requirements. Even in this case, usage of a state-of-the-art EAI platform would facilitate development and maintenance of the interfaces.

#### **Step 2.6: View processes identification – Private engine**

---

Similarly to 2.5, this step consists of identifying or defining the View Processes and linking them to the CBP, using a View Engine which only handles View Processes. The difference is that these View Processes must be mapped onto Internal Processes handled by the Private Engine.

The main issue here is to map processes already defined for different purposes (internal vs. external) and using different notations.

In this case, the main support needed is represented by consistent and interoperable process-modelling standards, as have been identified in both the Projects A1 and A2.2.

#### **Step 2.7: Configure private engine**

---

This step consists of configuring the Private Engine so that private processes and view processes can correctly communicate with each other.

The main issues for this activity are:

- Consistent mapping of states between internal processes and public views.
- Mapping of messages correlation IDs, following the approach presented in Section Error! Reference source not found.

#### **4.3.4.3. Prototype & testing**

---

##### **Step 3.1: Test cases definition**

---

Within this step a proper set of test cases is identified to support prototyping and testing of the CBPs being implemented. Test cases are defined according to the requirements of the specific organization and application sector.

No specific issues have been identified from the enactment architecture point of view.

##### **Step 3.2: Test integrated engine**

---

Specific issues related to testing the Integrated Engine architecture model are:

- Verify the consistency of the mapping and communication between View Processes and Internal Processes. This should be facilitated by the fact that both run on the same engine, and so it should be possible to verify them, e.g., through a unique simulation run.
- Verify correct messages correlation in both directions, from the Internal Process ID to the external partner View Process ID and vice-versa.

##### **Step 3.3: Test application interfaces**

---

Specific issues related to testing the Direct Application Integration architecture model are:

- Verify the correct communication between View Processes and the internal applications interfaces. This activity is critical and must be handled carefully, since it is not possible to circumscribe the test within one single system.

Indeed, a proper testing procedure will require cross-system test runs, that will have to be properly scheduled and monitored, taking the possible impact of external components into account (e.g., other application functions or the network infrastructure). For example, a problem caused by an application unexpectedly working on the same document ID can be

very hard to detect.

- Verify correct messages correlation in both directions, from the application-handled ID to the external partner View Process ID and vice-versa.

---

#### **Step 3.4: Test private engine integration**

Specific issues related to testing the Direct Application Integration architecture model are:

- Verify the correct communication between View Processes and the Internal Processes handled by the Private Engine. This activity should be facilitated by the fact that both environments offer monitoring functions to help problem identification. Anyway, coordinated tests should be run simultaneously on the two engines.
- Verify correct messages correlation in both directions, from the Private Process ID to the external partner View Process ID and vice-versa.

---

#### **Step 3.5: Test CBP monitoring**

No specific issues have been identified concerning how to test the CBP components of the architecture. The system should be tested with standard procedures, taking into account functionality aspects, i.e., fulfilment of the requirements, as well as technical aspects such as responsiveness and reliability.

---

#### **4.3.4.4. Activation**

The deployment of the architecture should follow the usual conventions applied by the involved organizations for their IT projects. Information support and training of users should take into account the following issues into account:

- Final users, involved in the normal enterprise activities through their internal workflow engines or enterprise applications should not suffer dramatic changes of their way of working. If this is not the case, then the whole project should be reconsidered, since one of our inspiring principles is to make cross-organization integration as transparent as possible.
- Technical users, such as workflow system administrators, should be trained on the new system and on the various interfaces with their existing systems, according to the three implementation models.

---

#### **4.3.4.5. Maintenance & change management**

---

##### **Step 5.1: Maintain integrated engine**

Maintenance of View Processes in the case of the Integrated Engine should be facilitated by the fact that they are managed in the same environment as the Private Processes. In this way,

inside-out changes, originating on the internal side, should have a direct impact on the View Processes and potential problems should be easily detected. Vice-versa is also true.

#### Step 5.2: Maintain application interfaces

---

This is the case which is more likely to generate problems on the maintenance side:

- Outside-in changes, from the CBP to View Processes down to the enterprise applications, might pose unexpected requirements to the interfaces that have been developed. For example, a new state or a new event type must be handled that requires a new call to be exposed by the underlying application. In the worst case, the application itself might have to be customized.
- Inside-out changes might cause unexpected problems on already running CBPs. For example, an application functionality change might impact the interface exposed to a View Process in unexpected ways. These problems are usually very hard to detect and may cause serious system disruptions.

#### Step 5.3: Maintain private engine

---

The management of change is not as complex as in the previous case, but still it requires a careful synchronization of changes between the two engines. In particular, the administrator of the Private Engine must be aware of changes on Private Processes that may impact the view processes, and take the necessary alignment actions. Automatic or semi-automatic alignment, via process model exchange tools, would improve this process.

## 4.4. Flexible execution and composition of services

---

### 4.4.1. ATHENA Service-Oriented Interoperability (SOI) Framework

---

According to W3C, a service-oriented architecture (SOA) specifies a set of components whose interfaces can be described, published, discovered and invoked over a network. SOA aims to promote software development in a way that leverages the construction of dynamic systems which can easily adapt to volatile environments and be easily maintained as well. The decoupling of system constituent parts enables the re-configuration of system components according to the end-user's needs and the system's environment. Furthermore, the use of widely accepted standards and protocols that are based on XML and operate above internet standards (HTTP, SMTP, etc) enhances interoperability.

The [ATHENA Service-Oriented Interoperability \(SOI\) Framework](#) provides guidelines for developing and integrating software services in service-oriented architectures (SOAs) using Web services and agent technologies.

## 4.5. Information interoperability

---

### 4.5.1. Introduction

---

Mapping and transformation of business documents are a core part of work being performed in action line A7. An overview of the A7 results will be incorporated in a later revision of this document. Existing ATHENA solutions focusing on information interoperability are:

- [Semantics and ontologies](#)
- [Model transformations as part of the model-driven interoperability approach](#)

## 4.6. Semantics and ontologies - Application guidelines

---

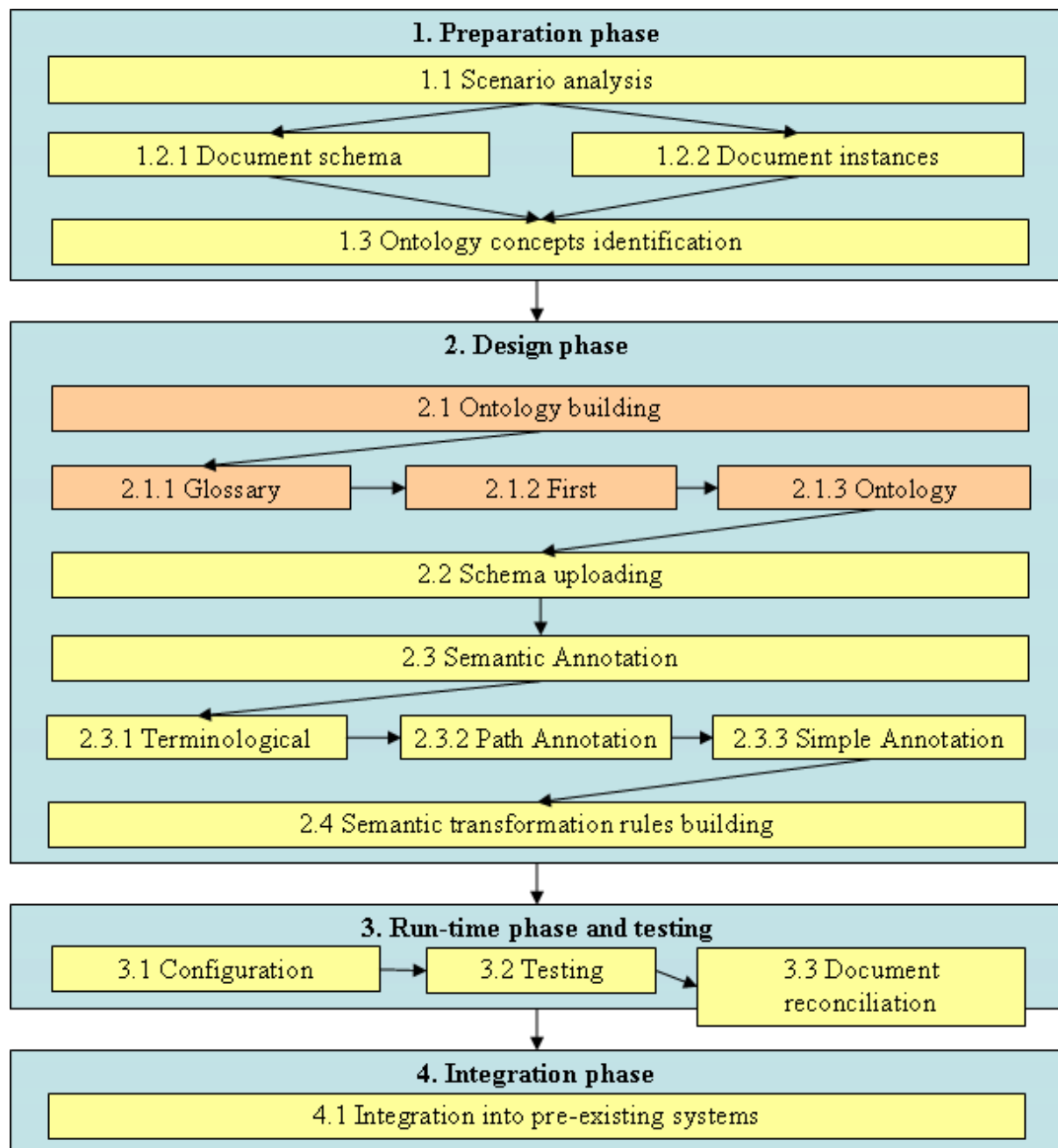
### 4.6.1. Introduction

---

This section collects guidelines and procedures that can help users to deploy and use the A3 solutions for the semantic reconciliation in a real scenario. Three main phases are described below:

- **Preparation phase:** Analysis of the business scenario and preparation of the necessary material for the deployment.
- **Design phase:** After the creation of the reference Ontology, using the A3 tools each partner design its own environment, generating semantic annotations and semantic reconciliation rules.
- **Run-time phase:** The actual reconciliation of documents is performed and tested.
- **Integration phase:** The run-time reconciliation services provided by A3 are integrated into legacy systems.





#### 4.6.2. Preparation phase

This phase should be performed, mainly, by functional analysts and domain experts in order to prepare the field to the specialist who will design and deploy the scenario.

#### **4.6.2.1. Task 1.1: Scenario analysis**

---

This phase should include a first analysis of the functional requirements which can help the technical analyst to figure out all the correct artefacts that should be deployed in the environment. Besides, the identification of all the details related to the business scenario and an agreement on a common understanding, at business level, shared by all the partners involved in the semantic integration allows a better identification of the main concepts for the development of the Reference Ontology.

#### **4.6.2.2. Task 1.2: Document analysis**

---

##### **Task 1.2.1: Document schema identification**

---

The first technical task is to collect the schemas of the documents that should be exchanged and reconciliated at runtime, in order to allow, in the design phase, the definition of the semantic annotation and rules on these schemas.

##### **Task 1.2.2: Document instances identification**

---

Since the actual reconciliation works on instances of the schema for which reconciliation rules have been previously defined, a set of such documents needs to be identified. These documents should be used also in the testing phase.

#### **4.6.2.3. Task 1.3: Ontology concepts identification**

---

After the identification of the basic artefacts that will be used in the design and test phase, it is possible to perform a deep analysis of the concepts found in those resources in order to get a better understanding of the overall scenario we are dealing with. This analysis can help the identification of all the concepts needed in the ontology and can help the analysts to understand if they need to build a new ontology or how they can use an existing one.

#### **4.6.3. Design phase**

---

All the tasks included in the design phase aim to design all the information needed for performing the run-time phase.

##### **4.6.3.1. Task 2.1: Ontology building**

---

This task is accomplished using the Athos tool. Starting from the analysis performed before, the semantic specialists collect all the information and start to build the ontology, first with the definition of a glossary of common terms, then adding a first enrichment in order to get a taxonomy and at last finalizing the ontology including the relationships among concepts. Both, business analysts and semantic experts should cooperate during this phase in order to follow the functional aspects and maintain semantic consistency on the final ontology.

This task should not be always needed because after a scenario is covered, similar cases can use the same ontology.

---

**Task 2.1.1: Glossary definition**

This is the first step towards a stepwise construction of an ontology. At this stage, only the terminological level is addressed. Structuring of concepts, as well as relationships among them are not considered. The definition of a glossary requires first the identification of the relevant terms in the domain, gathered in a lexicon; then the latter is progressively enriched with definitions, yielding the glossary.

---

**Task 2.1.2: First enrichment**

A glossary is a powerful tool, since it establishes definitions of terms. But since such definitions are expressed in natural language, a glossary is more suitable for human being than automatic reasoning. For this reason, the second step in the construction of an ontology is the definition of taxonomies, usually specialization and decomposition. Taxonomies are a useful way of linking concepts, since they identify hierarchies among concepts.

---

**Task 2.1.3: Ontology finalization**

The real ontology needs the identification and establishment of other semantic relationships, such as attributes, relatedness, in order to build a complete semantic net as an ontology is.

---

**4.6.3.2. Task 2.2: Schema uploading**

The annotations and the semantic rules are strictly related and performed on RDF schemas of the business document that should be exchanged at run-time. For this reason both A\* (for annotations) and Argos (for the reconciliation rules) share common RDF schemas using Themis. The Themis repository is also used as common reference for all the tools of the A3 framework and it provides functionalities for adding relations between document schemas and services used for exchanging message instances based on those schemas.

The analysts should upload RDF schemas of the business documents retrieved in the preparation phase. Often, in existing scenarios, the documents are defined as XML schemas which should be translated into RDFS using semi-automatic translators or following the rules of the RDFS syntax. The tool itself lets the users to upload only valid schemas in order to ensure quality check even during the design phase.

---

**4.6.3.3. Task 2.3: Semantic Annotation**

This task aims to enrich the knowledge on the business resources (in our case mainly RDF schemas of the business documents) allowing the creation of relations between the ontology concepts and the business elements. The Semantic Annotation is accomplished by the A\* tool. The methodology behind the tool identifies 4 increasing levels of annotation:

terminological, path, simple and full. At least the third level of annotation should be reached for reconciliation purposes. Like the task for the creation of the ontology, also this phase should involve both domain experts and semantic specialists, in order to bridge the gap between the business scenario and the semantic needs.

---

#### **Task 2.3.1: Terminological annotation**

It consists on a keyword-based annotation. Terms appearing in the schema that is being annotated are simply associated with a set of terms from the ontology.

---

#### **Task 2.3.2: Path annotation**

At this level, also the structure of the schemas and the ontology are taken into consideration. Complete paths, from the root element to the leaves, on the annotated schemas are associated with a set of paths from the ontology.

---

#### **Task 2.3.3: Simple annotation**

Using the simple annotation, it is possible to specify the type of mismatch that each annotation intends to cover. Such an information will be used also during the development of the semantic reconciliation rules (in the next task).

---

#### **4.6.3.4. Task 2.4: Semantic transformation rules building**

During the run-time phase the Ares reconciliation engine use semantic rules defined again document schemas to apply the reconciliation on the related document instances. These reconciliation rules are written by semantic specialists starting from the annotations made with A\* and using the web interface provided by Argos. The advantage is that the users do not have to write rules for each direct transformation from a document schema to another but only from their own schemas to the ontology format. In particular, for each schema a set of rules needs to be built for incoming messages (Backward rules) and another set of rules for outgoing messages (Forward rules). In this context each partner has to care just about its own part and can easily plug its infrastructure into existing semantic-enabled environments.

Argos uses the syntax of the Jena library for specifying the rules in order to use the Jena reconciliation engine. Using that syntax the user can cover many kinds of transformations. The possibility to use also Java methods for applying specific actions gives an idea of the wide range of opportunities given to the users. Of course the usage of that syntax is not so user-friendly, so Argos provides a web interface to design the rules starting from a set of pre-existing rule templates which cover all the most common transformations:

- *Map*: to map a concept of the source model into a concept of the target model
- *Map Table*: to map the values of a concept having an enumeration type into the values of another enumerated concept in the target model
- *Convert*: to convert a concept of the source model to a concept of the target model by using a conversion parameter

- *Sum*: to sum 2 or more concepts of the source model to a concept of the target model
- *Mult*: to multiply 2 or more concepts of the source model to a concept of the target model
- *Split*: to split a concept of the source model to 2 or more concepts of the target model
- *Merge*: to merge 2 or more concepts of the source model into a concept of the target model

#### 4.6.4. Run-time phase and testing

---

This include a first configuration phase, followed by the testing phase applied in order to check the consistency of rules provided in the design stage and at the end the reconciliation of real documents.

##### 4.6.4.1. Task 3.1: Configuration of the reconciliation engine

---

A first configuration of Ares is needed in order to set up all the necessary parameters for finalizing the integration with the other tools of the framework. In particular, each installation of Ares has to be linked to a single ontology in order to define the context in which the tool is working on and it has also to know the address of the service provided by Themis, for retrieving schemas information, and Argos, for retrieving the reconciliation rule sets.

##### 4.6.4.2. Task 3.2: Testing

---

A first testing phase is needed in order to check the actual reliability of the details provided during the design phase. In particular the reconciliation rules have to be tested on real document instances. Ares provides a web interface which allows the users to test directly particular sets of rules on RDF instances.

##### 4.6.4.3. Task 3.3: Document reconciliation

---

If the testing phase produces reliable results it is possible to start the real reconciliation between systems using the services provided by Ares. This step is completely automatic and performed by the Ares reconciliation engine with the support of the other tools of the semantic suite. It provides a set of services that take as input the document instance to be reconciled written in RDF format and the endpoint URLs of the services from and to which the message is sent. Automatically, the tool retrieves, using Themis as reference, the related sets of reconciliation rules from Argos and applies the transformation giving as output a new RDF instance which follows the final format.

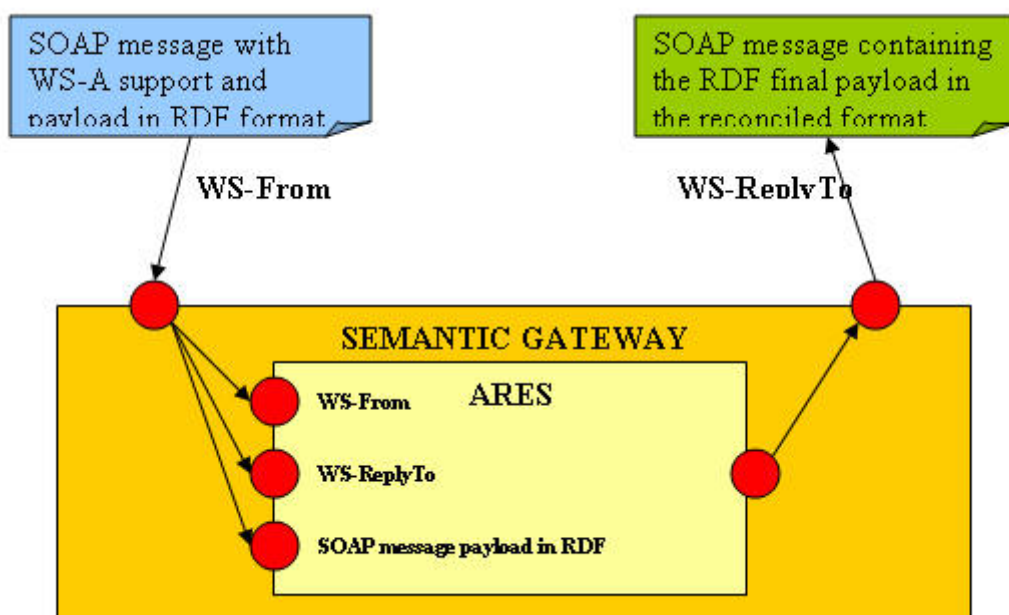
#### 4.6.5. Integration phase

---

The entire semantic reconciliation process can include also this phase which should provide a deep integration of the semantic framework with pre-existing infrastructures. Ares is the only run-time interface of the semantic framework with other systems and it is based on Web services standards. It provides also Java API for who looks for more performance and a more

coupled integration. All these aspects should help the integration but sometime could not be enough. In particular for quick deployments it is necessary to support common message exchange standards. For this reason ATHENA is providing the reconciliation service integrated in a component for the message management called Semantic Gateway.

The Semantic Gateway uses Johnson as message gateway which ensures the possibility to use, easily, common standards such as the WS-Addressing or the WS-ReliableMessaging in order to deploy straightforwardly the reconciliation service into infrastructure that use those standards. This configuration allows also to manage different aspects related to the message management and to develop advanced functionalities and different business scenarios.



## 4.7. Dynamic architectures

### 4.7.1. ATHENA Model-Driven Interoperability (MDI) Framework

Model-driven development (MDD), and in particular OMG's Model Driven Architecture (MDA), is emerging as the state of practice for developing modern enterprise applications and software systems. The MDD paradigm provides us with a better way of addressing and solving interoperability issues compared to earlier non-modelling approaches.

The [ATHENA Model-Driven Interoperability \(MDI\) Framework](#) provides guidelines for how model-driven development (MDD) approaches can be applied in developing interoperable enterprise software systems.

The MDI framework aims at providing guidelines for the following topics:

- Model-driven architecture (MDA) and interoperability
- Metamodelling
- UML profiles and domain-specific languages (DSLs)
- Model transformations
- Method engineering

## 5. Technical architecture

---

### 5.1. Technical architecture

---

#### 5.1.1. Service-oriented architecture

---

The technical architecture proposed by ATHENA follows the principles of a service-oriented architecture (SOA). SOA refers to the latest trend in system architectures where typically Web services and technologies play an important part in achieving interoperability.

Service-oriented development emerged as an evolution of the component-based development and among its goals is to support the loose coupling of system parts in a far better way than existing component-based technologies.

The OASIS reference model for SOA [OASIS] defines SOA as:

*“Service-oriented architecture (SOA) is a paradigm for organising and utilising distributed capabilities that may be under the control of different ownership domains.”*

According to this definition SOA differs in organising and understanding information communication technology (ICT) related to previous approaches:

- First, SOA reflects the reality that ownership boundaries are a motivating consideration in the architecture and design of systems.
- Second, SOA applies the lessons learned from commerce to the organisation of ICT assets to facilitate the matching of capabilities and needs.

The value of SOA is that it provides a simple scalable paradigm for organising large networks of systems that require interoperability. The ramifications of service-oriented development can be observed both at the system and the business level. Having systems composed of services offered by various service providers provides the basis for supporting new business models, such as “virtual organisations”. Services can be seen as business capabilities that support the enterprise. From the ICT perspective, a service is an ICT representation of business functionality that is implemented via multiple messages that return state and/or change state of an associated entity.

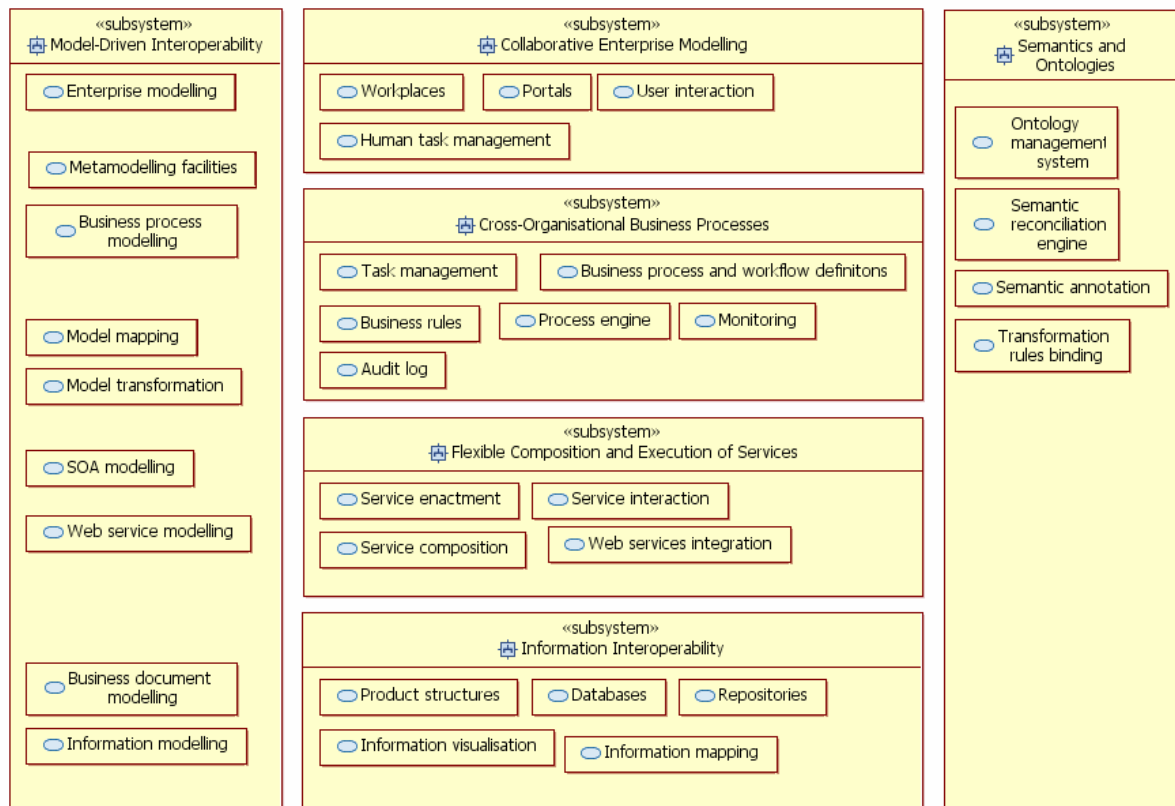
SOA also allows us to integrate more adaptive and dynamic architectures such as enterprise knowledge architectures, business process management suites, agent architectures and peer-to-peer (P2P) architectures. SOA aims to promote software development in a way that

leverages the construction of dynamic systems which can easily adapt to volatile environments and be easily maintained as well. The decoupling of system constituent parts enables the re-configuration of system components according to the end-user's needs and the system's environment. Furthermore, the use of widely accepted standards and protocols that are based on XML and operate above internet standards (HTTP, SMTP, etc) enhances interoperability.

### 5.1.2. Technical architecture of the ATHENA platform

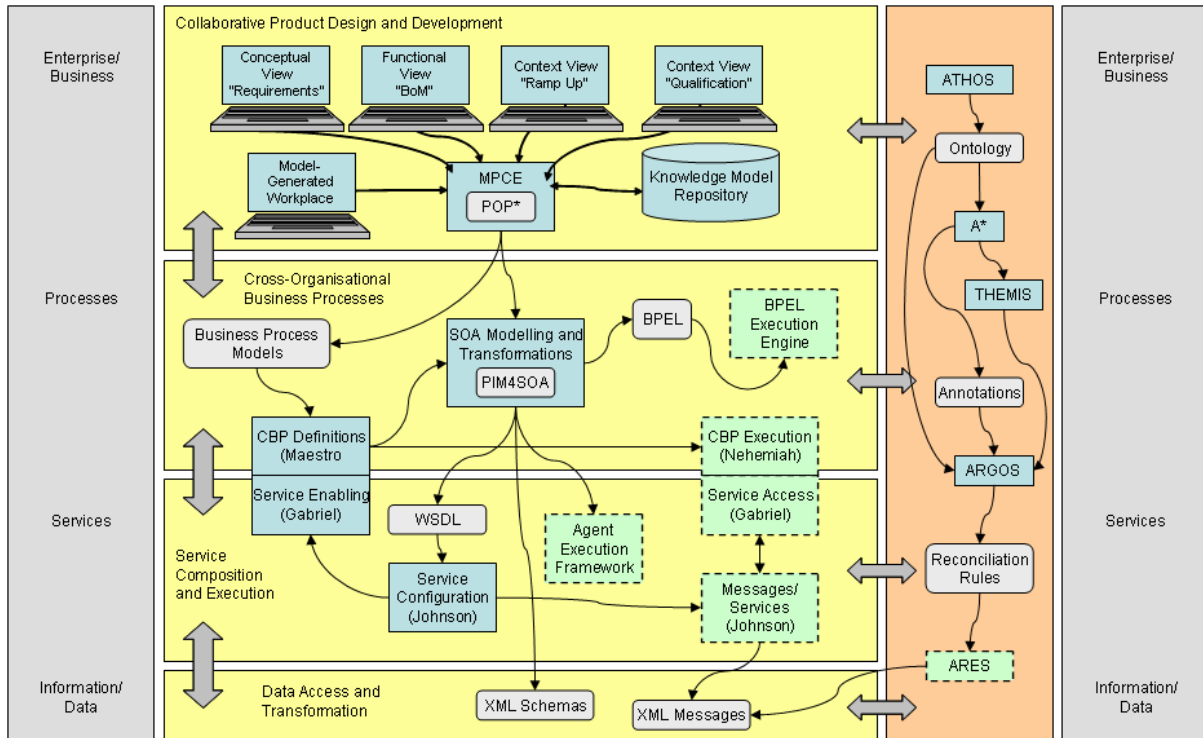
A SOA platform provides all the necessary services to develop new software capabilities and integrate existing enterprise applications and systems. There is a trend towards SOA platform consolidation which includes data and information integration services, service communication and application integration services, business process and composite application services, and interaction and workplace services. In addition, modelling capabilities are being included as part of modern SOA platforms.

The ATHENA platform is illustrated in the figure below according to the ATHENA interoperability reference architecture. It shows some of the main functionality offered by software tools and infrastructure components developed (or offered by partners) in ATHENA.





The figure below gives a high-level description of one possible implementation of the technical architecture using and configuring technical solutions components developed in ATHENA.



## 5.2. Collaborative enterprise modelling

### 5.2.1. Introduction

Enterprise modelling is a technology that many companies would say they are applying. The application of EM in industry is predominantly for Enterprise Architecture definition and Business Process Management, and the comprehension of what it is and how to use it is linked to consultancy and modelling expertise. Industrial use spans from the definition of business plans to the definition of the ISO 9000 compliant quality system. Many practices are considered as enterprise modelling by the industry.

EM is rapidly developing and transforming into providing visual languages, best-practice patterns and visual knowledge spaces. The new era for EM is driven by the advent of more advanced approaches, methodologies, infrastructures and platforms and a need for families of solutions that allow predictable customisation.

Enterprise Modelling aims to support enterprises by dealing with the several aspects of interoperability:

- *Heterogeneity*, incommensurable knowledge and information perspectives, systems and software infrastructures, working practices, etc. among the partner companies.
- *Need for Flexibility*, due to need for innovation, learning, change and exception handling;
- *Complexity*, the richness and uncertainties of interdependencies within and among partners, their activities, resources, skills and products.
- *Heterogeneity*, need for flexibility, and complexity must be managed at different levels:
  - *Knowledge*, approaches, methods and skills needed for innovation, problem solving and work performance, the shared language and frames of reference needed for communication, etc.
  - *Process*, the planning, coordination and management of cooperative and interdependent activities and resources;
  - *Infrastructure*, the information formats, software tools, and interoperability approaches of the participating companies.

However a lot of EM languages and tools are developed in the meantime to support enterprises for defining their own entire architecture. In order to collaborate enterprises have to share their models across modelling languages.

### 5.2.2. Solution

The POP\* language (stands for Process, Organisation, Products and other enterprise dimensions like Systems) defines a core set of enterprise issues to be defined in an enterprise model as a flexible intermediate language to facilitate model exchange between different enterprise modelling tools. The guideline for applying POP\* enables companies to share knowledge in a structured way.

The Modelling Platform for collaborative enterprises (MPCE) supports the POP\* language and provides model management and model exchange services. The MPCE can be used as a web-service hosted somewhere or can be locally installed. In Figure 11 the conceptual solution is shown.

The major advantage of the POP\* concept and the MPCE is the capability to keep models consistent even by using different modelling tools. So modelling elements which do not exist in one tool will be not destroyed to be used in a different tool. POP\* had already influenced the work on ISO 19440.

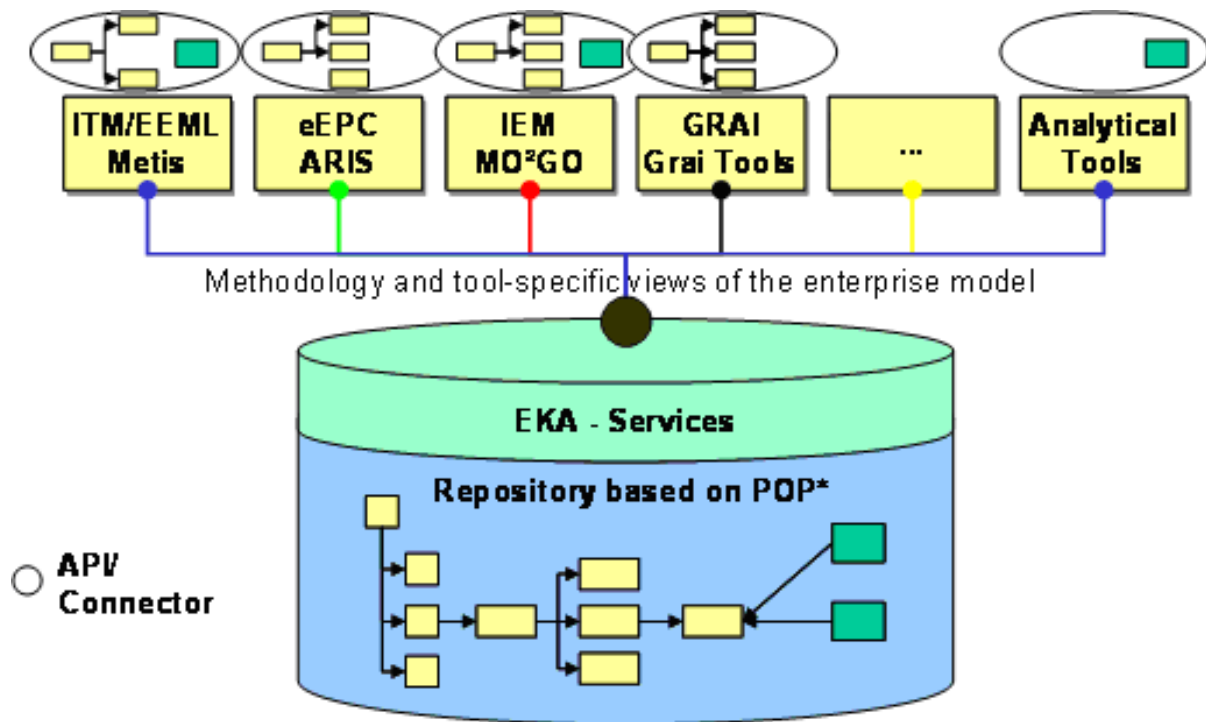


Figure: POP\* - EM exchange concept

### 5.2.2.1. Involved tools

In alphabetical order

- ARIS – IDS Scheer (Germany)
- GRAITools – ITREC (France)
- METIS – Troux (Texas, USA)
- MO²GO – Fraunhofer IPK (Germany)
- POP\* UML Plugin for Rational Rose Software Modeller – ESI (Spain)

### 5.2.3. Application case

In a furniture supply chain the manual ordering system has to be reengineered. For the first analysis a model was elaborated. But then for the definition and implementation of the to-be scenario some difficulties happen. Stakeholders like the supplier, the IT department, consultants and the enterprise management needs their views and adapted modelling languages for work support. In order to allow the partners to stay in their corner the MPCE was installed as model repository which allows to connect to the different modelling tools. In Figure 12 the architecture is given.

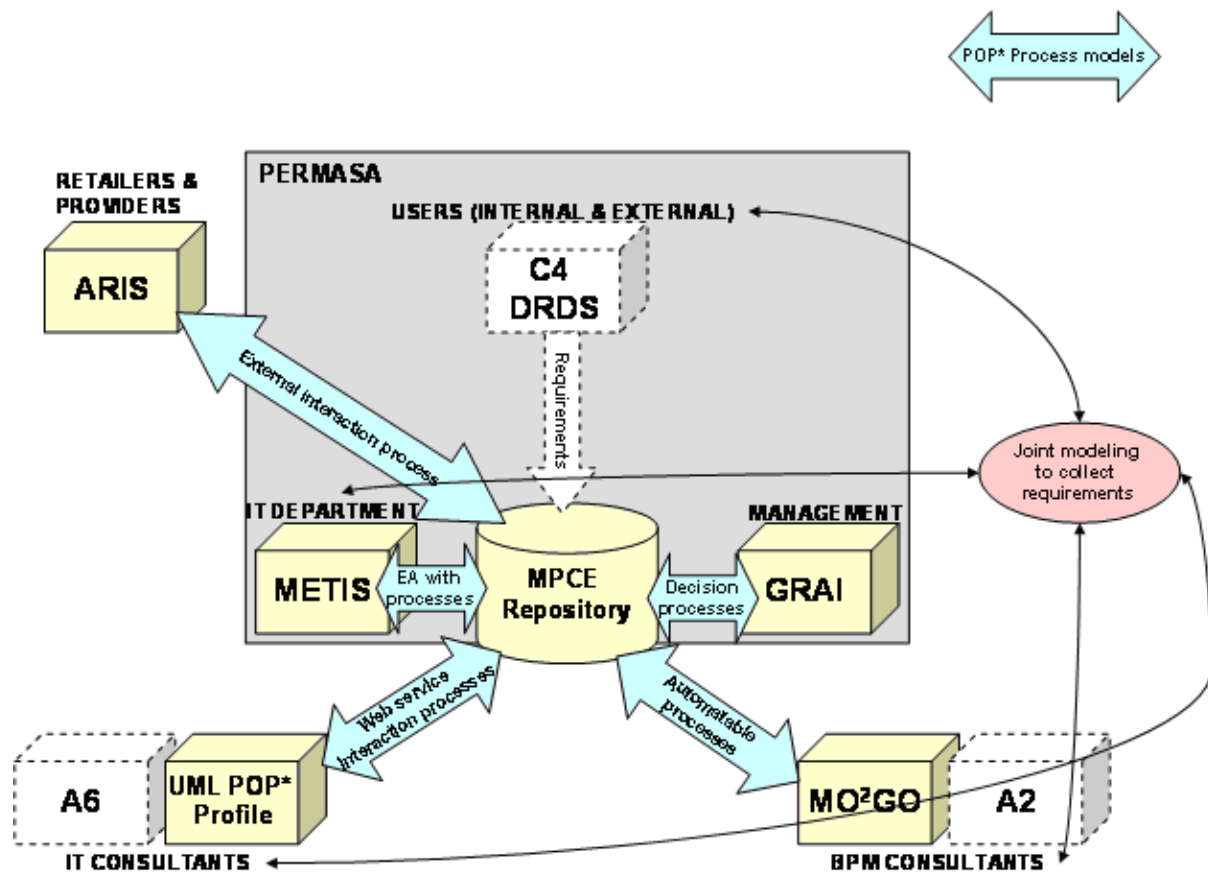


Figure: Application of MPCE and POP\* - allow model-based communication between different stakeholders

### 5.2.4. EM market

The market penetration of EM is about 8% in the US market and 7% in Europe according to Gartner Group, and the EM markets are still perceived and measured as separate markets with approaches, methodologies, tools and solutions separate from the operational enterprise systems and solutions. Most EM projects are performed disjoint from the operational environment and solutions being modelled. So the purpose of EM is mostly for creating improved insight, overview and common understanding across disciplines and processes.

The dominant market in the US is the Enterprise Architecture (EA) Market, while in Europe the EA market is developing rather slowly, but for a few exceptions. In Europe the Business Process Modelling market has so far been the dominant market.

### 5.3. Cross-organisational business processes

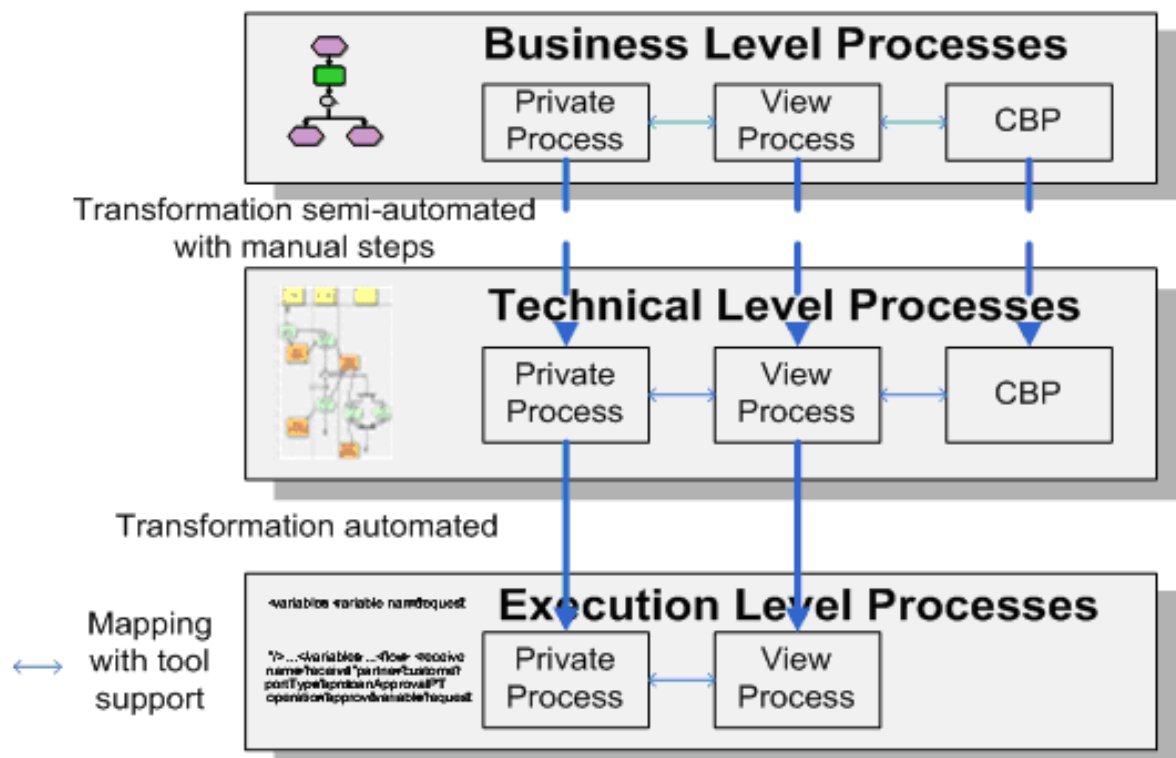
### 5.3.1. Introduction

The ATHENA CBP modelling approach combines two ideas:

- Different user groups and modellers are involved in modelling cross-organizational business processes. Their different perspectives and needs have to be reflected in the modelling method.
- The modelling method should allow for selectively hiding internal process steps while offering a mechanism to expose CBP relevant information to partners.

Therefore we propose a CBP modeling framework in the form of a matrix. The different levels on which CBP modeling is performed (business level, technical level, implementation level) are represented on the vertical axis. On the horizontal axis the different model types of the process view concept are shown. At each intersection of a vertical and horizontal axis, we can identify a possible process model to capture tasks and relationships of cross-organizational interactions. Thus it is ensured that all relevant perspectives on CBP models as well as the processes required for the view concept are properly captured and modeled.

Transformations between the different modeling levels are necessary. Between the business level and the technical level they can be executed semi-automatically, between the technical level and the execution level they can be automated.



### 5.3.2. Tools

---

The following ATHENA tools are available for designing and implementing cross-organisational business processes:

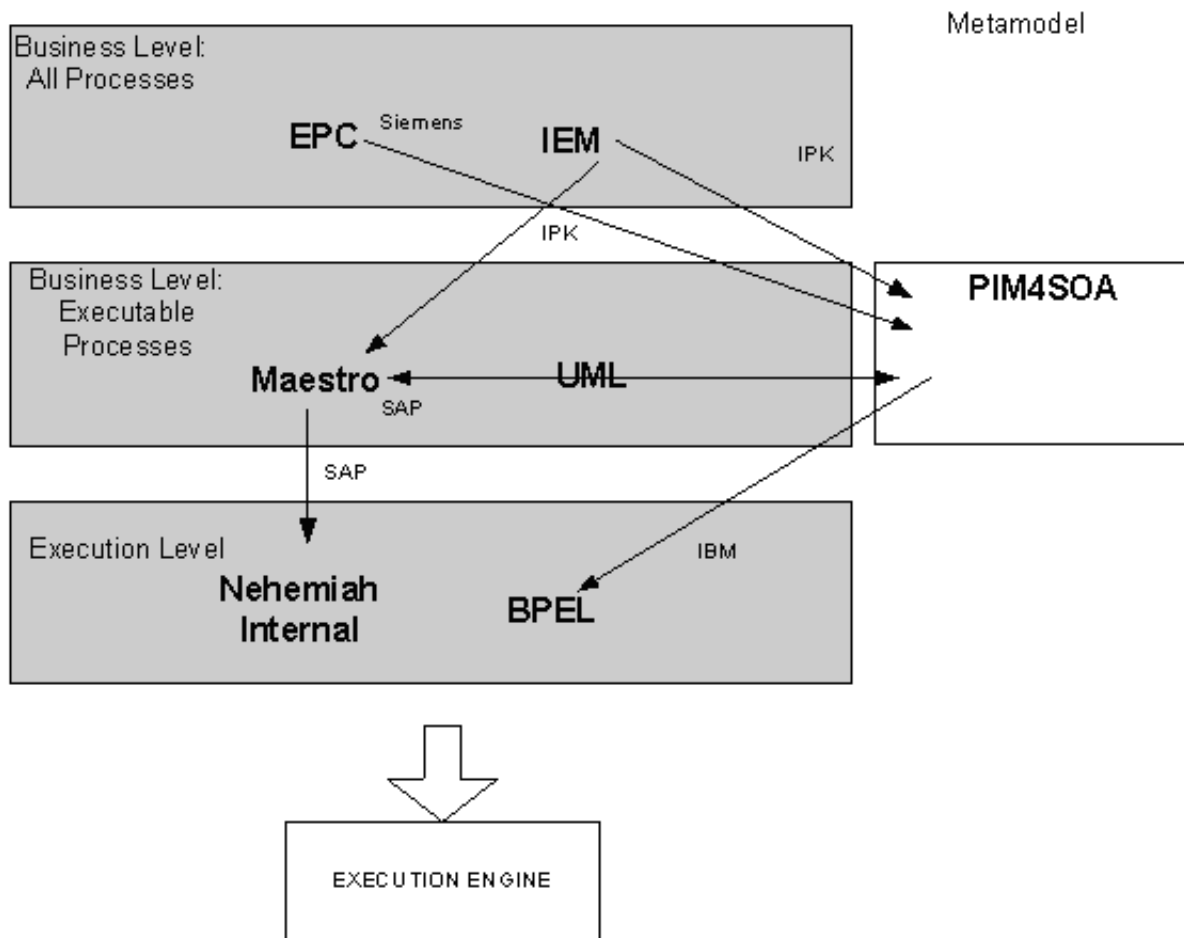
- MO<sup>2</sup>GO is an enterprise modelling tool. MO<sup>2</sup>GO supports the integrated enterprise modelling (IEM). MO<sup>2</sup>GO NG has as well been extended to support modelling of CBPs on the business level. It also provides export functionality to transform process models from the business level to the technical level. This supports re-use of process models so that users do not have to completely re-model processes when enriching them with information relevant for execution.
- ARIS is an enterprise modelling tool. The Architecture of Integrated Information Systems (ARIS) supports the modelling of Event-driven Process Chains (EPC). ARIS has been extended to support the methodology for modelling of CBPs on the business level. To model CBPs each partner starts from a private process describing the steps executed in its organisation. Then a view process is created that provides a process-oriented interface to the partners whilst at the same time hiding internal process steps that should not be published. The CBP then links the view processes of all partners and defines at which steps data and messages are exchanged between partners.
- Maestro is a Business Process Modeling Tool on a technical level that allows for modeling of private processes, view processes, CBPs and their links. Processes modeled in Maestro can be exported into the Nehemiah enactment engine for execution. Maestro also offers functionality to manage business partners that provide view processes to be added to a common CBP. Partners can be added and changed in Maestro and are directly updated in the Nehemiah repository.
- Nehemiah is an implementation of the ATHENA Process Engine. Nehemiah is a Business Process Management Engine that executes cross-organisational business processes in a distributed environment and supports the process view approach. Nehemiah has a Web front end for controlling and monitoring the execution state of the CBP in a Web browser.
- ATHENA Event and Document Correlation (AEDoC): Each process and resource involved in the execution of a Cross-Organisational Business Process (CBP) has to be exclusively identified. This identification is used to link together CBPs, process instances and message payloads. In order to sustain this duty, the ATHENA Event and Document Correlation (AEDoC) provides a set of basic services for matching documents to process instances. The execution of a CBP requires that private documents and events are continuously linked to the correct process instances that can run all the different process engines in the whole architecture of the collaborating parties.

### 5.3.3. Model transformations

---

In order to facilitate the CBP modelling method and to end up with executable models, various model transformations are necessary. The figure below shows all necessary and implemented model transformations within ATHENA and the responsible partner:

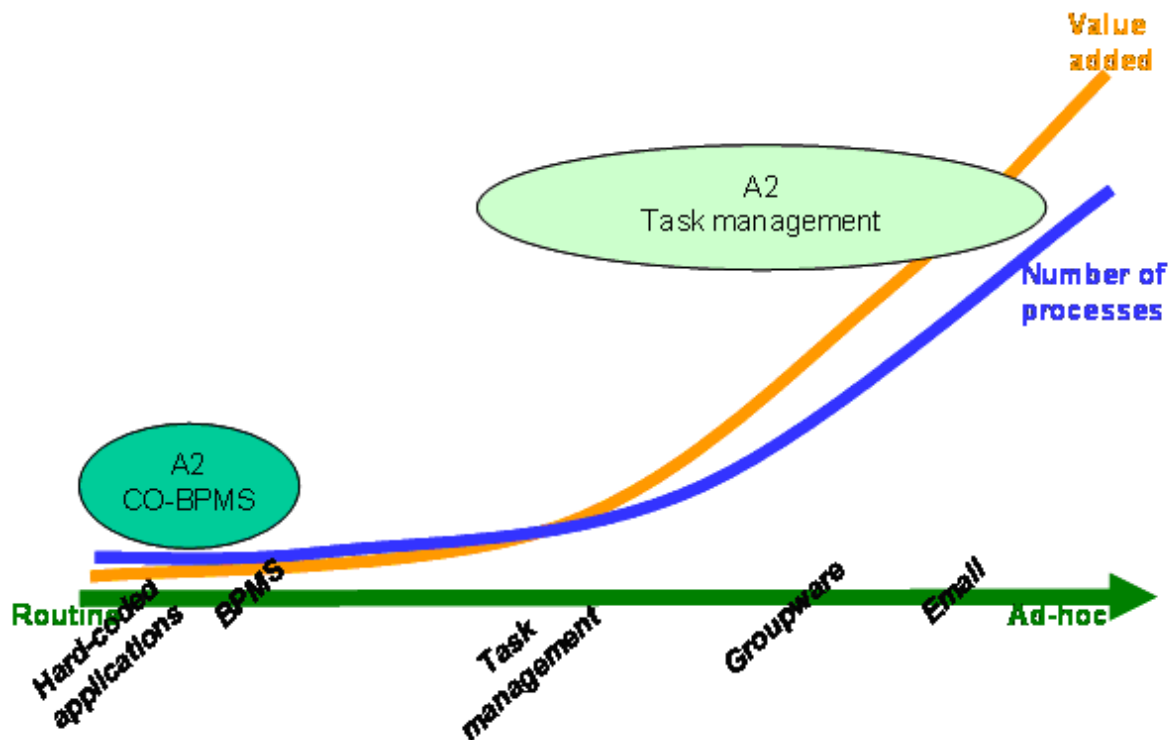
- **ARIS EPC to PIM4SOA:** The ARIS EPC to PIM4SOA transformation tool provides means to transform ARIS business level description to platform independent, service-oriented PIM4SOA models. ARIS XML export (AML) is transformed to a XMI serialisation of PIM4SOA models. The transformation makes use of and is based on the ARIS modelling style for cross-organizational business processes (CBPs). It translates business process models to platform independent ICT models based on a service-oriented architecture.
- **Mo<sup>2</sup>Go IEM to Maestro and PIM4SOA:** Mo<sup>2</sup>Go also provides export functionality to transform process models from the business level to the technical level. To directly link to Maestro a specialized IEM–BPDM export is available. The output file format is XMI 1.2 and uses UML 1.4. The resulting files can be imported in UML tools and into Maestro for further use. Additionally an export to PIM4SOA models as an intermediate format is available. In particular the view processes are transformed. The processes in MO<sup>2</sup>GO can be annotated to indicate executable, non-executable processes or processes requiring user interactions. This information is used to transform only execution relevant processes.
- **PIM4SOA to BPEL:** Allows users to take a PIM4SOA model (e.g. generated from higher level tooling) and convert to an execution platform (BPEL). Rather than a direct model to text transformation, the web service layer PSM transformations make use of platform specific models. For the BPEL transform, an Ecore / EMF model of BPEL has been created to manipulate the transformed process. The EMF implementation of BPEL is generated directly from the XSD (BPEL schema).



### 5.3.4. Need for task management

Distributed, knowledge based cooperation must be supported by flexible information systems (IS). Business process systems such as workflow management, enterprise resource planning, and supply chain management, apply models to facilitate work performance, control, management and coordination. In these systems, process modelling and execution are separated, performed at different times, by different people, using different tools. While capable of automating routine procedures (the left end of the process spectrum depicted below), such systems cannot handle ad-hoc, evolving processes (the right end). These processes are becoming increasingly important in the global, networked economy, so the scope of process support should be extended.





## 5.4. Flexible execution and composition of services

### 5.4.1. SOA framework

The framework for Rapid Prototyping of SOAs presented here is composed of three parts: a *modelling* part, a *service* part and an *autonomous agent* part.

The *modelling* part is concerned with applying Model-Driven Development (MDD) techniques and tools to the design of SOAs. It defines models and *modelling* transformations that are specific to the concepts used for SOAs, such as Web Service descriptions and plans for autonomous agents. The *service* part provides a highly flexible communication platform for Web services. The *autonomous agent* part deals both with designing and enacting service compositions as well as performing mediation, negotiation and brokering in SOAs.

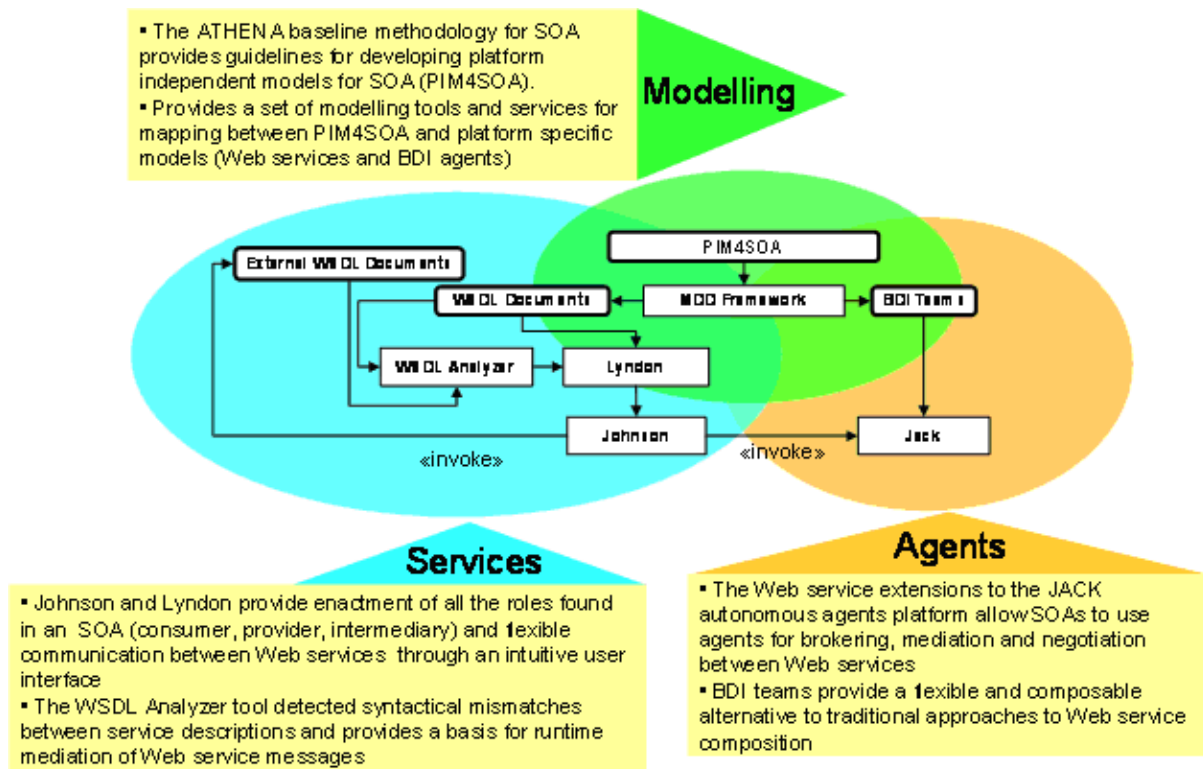


Figure: SOA framework

### 5.4.2. Modelling

The *ATHENA baseline methodology for SOA* introduces a model-driven development (MDD) approach to specifying interoperable service-oriented architectures realized as Web services. In model-driven development are used models to describe business concerns, user requirements, activities, information structures, components and component interactions. These models govern the system development in that they can be transformed to program code. We aim to develop tools to automate model transformations for service-oriented architectures. Hence, the term model-driven development in our context encompasses both the development of models, and tools for model transformation. *ATHENA baseline methodology for SOA*

The models are expressed in UML, and supported by UML profiles for SOA and Web services. The baseline methodology provides guidelines for how to develop the different kinds of models recommended for SOA. Some of them lay the basis for automated code generation; all of them contribute to the understanding and specification of the system or services to be developed.

The PIM4SOA metamodel defines modelling concepts that can be used to model four different aspects or views of a SOA:

- **Service:** Services are an abstraction and an encapsulation of the functionality provided by an autonomous entity.
- **Information:** Information is related to the messages or structures exchanged, processed and stored by software systems and components.
- **Process:** Processes describe sequencing of work in terms of actions, control flows, information flows, interactions, protocols, etc.
- **Quality of service (QoS):** Extra-functional qualities that can be applied to services, information and processes.

Model transformations are developed according to the OMG Model Driven Architecture (MDA) (Soley 2000) approach between a Platform Independent Model (PIM) for SOA (PIM4SOA) and Platform Specific Models (PSMs) for describing Web services (XSD and WSDL), Jack BDI agents and BPEL (Andrews et al. 2003) processes. PIM4SOA is a visual PIM which specifies services in a technology independent manner. It represents an integrated view of the SOA in which different components can be deployed on different execution platforms. The PIM4SOA model helps us to align relevant aspects of enterprise and technical IT models, such as process, organisation and products models. This model allows us to raise the abstraction level at which we can talk about and reason on the architecture we design.

### 5.4.3. Services

---

The part of our SOA Rapid Prototyping framework that deals with the enactment of Web services is composed of three tools which are arranged along a value chain: the WSDL Analyser, the Lyndon tool and the Johnson tool.

- The WSDL Analyzer is a tool for detecting similarities between Web service descriptions. The tool can be used to find a list of similar services and produces a mapping between messages, thereby enabling brokering and mediation of services. The algorithm of the WSDL Analyzer improves over an algorithm for finding structural similarities proposed by Wang and Stroulia (Wang et al. 2003) by taking into account additional features of the WSDL structure. More specifically, we make use of the *tree-edit distance* measure (Shasha et al. 1997) and the concept of a *weak subsumption relation* (Nagano et al. 2004). *tree-edit distance* *weak subsumption relation*
- Johnson is a runtime tool that enables users to enact most of the roles typically found in an SOA, thereby enacting complex SOA scenarios by sending real SOAP messages between Web services without having to write a single line of code. Johnson features a Web-based user interface designed to closely resemble Web-based email applications, with the only difference that SOAP messages and Web Services endpoints are used in place of email messages and email addresses. The user can see incoming SOAP messages in the Inbox and create outgoing SOAP messages in the Outbox that will be sent to external Web services. A powerful user-interface generator relieves the user from having to deal with XML documents by generating forms for displaying and editing any XML-based data type.

- The Lyndon tool can be seen as the design-time counterpart of the Johnson tool. It analyses WSDL files and automatically configures Johnson for playing either the role of consumer or provider of the service described. Lyndon parses a WSDL file and determines which endpoints need to be created, and which processing chains need to be assigned to them. Determining which processing modules to include in the processing chain takes into account information extracted from the WSDL file as well as options set by the user. The user may, for example, specify whether Johnson should be configured as a service consumer or a service provider, or whether messages sent to or from the service should be logged. Some configuration information can be extracted from the WSDL file, such as the need for implementing the WS-Addressing specification, which is specified as part of the description of the bindings of a Web service.

#### 5.4.4. Agent

---

The aim of the extended JACK agent framework for Web Services is to provide a goal-oriented service composition and execution module within an SOA.

Following the Belief Desire Intention (BDI) model, agents are autonomous software components that have explicit goals to achieve or events to handle (desires). Agents are programmed with a set of plans to describe how to go about achieving desires. Each plan describes how to achieve a goal under varying circumstances. Set to work, the agent pursues its given goals (desires), adopting the appropriate plans (intentions) according to its current set of data (beliefs) about the state of the world. The combination of desires and beliefs initiating context-sensitive intended behaviour is part of what characterises a BDI agent.

BDI agents exhibit reasoning behaviour under both pro-active (goal directed) and reactive (event driven) stimuli. Adaptive execution is introduced by flexible plan choice, in which the current situation in the environment is taken into account. A BDI agent has the ability to react flexibly to failure in plan execution. Agents cooperate by forming teams in order to achieve a common goal.

The JACK agent platform is not inherently ready for interaction within a Web service environment. Additional steps are necessary for enabling interactions between the agent platform and Web services, especially when the agents themselves offer services. In this case, some tools are needed for generating the server and client-side code for using JACK inside a Web server.

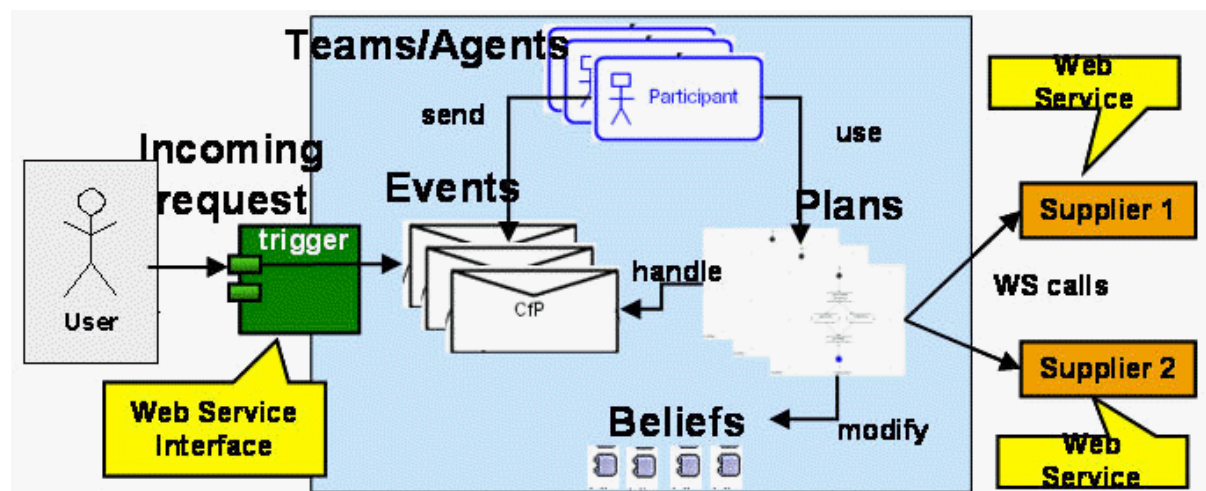


Figure 23: Extended JACK framework for service composition and execution

Figure 23 is an overview of the extended JACK architecture for Web service composition and plan execution, with at its core the JACK agent framework with plan library and knowledge base. Following the MDA approach, a modeller specifies at design time a set of plans (PSM level) that constitute the workflow library of the agents. Web service calls are integrated as steps into plans. Workflows are modelled graphically and most of the common workflow patterns are supported.

In order to prepare for a transformation from a PIM4SOA model to the JACK PSM, service providers are mapped to Jack agents/teams. The parts of the PIM which define the processes involved are mapped to agent/team plans and correlated events, whereas the parts which define the interfaces are mapped to the modules which provide the client- and server-side code for the JACK agent platform.

Just like BPEL, our framework supports fixed composition, where the structure and the components of the composition are statically bound, and semi-fixed compositions, where the structure is statically bound but the actual service bindings are performed at runtime. More explorative compositions, where both structure and components are created at runtime, are beyond what BPEL or BDI agents can offer.

However, there are several advantages to BDI agent, especially when it comes to handling failures at runtime. A plan is executed in a context which specifies conditions for plan instances and also other applicable plans. An exception in one plan instance then leads to the execution of another plan instance for the next known service. The BDI agent approach supports this adaptive behaviour in a natural way, whereas a BPEL process specification which attempts to provide the same behaviour would require awkward coding such as nested fault handlers.

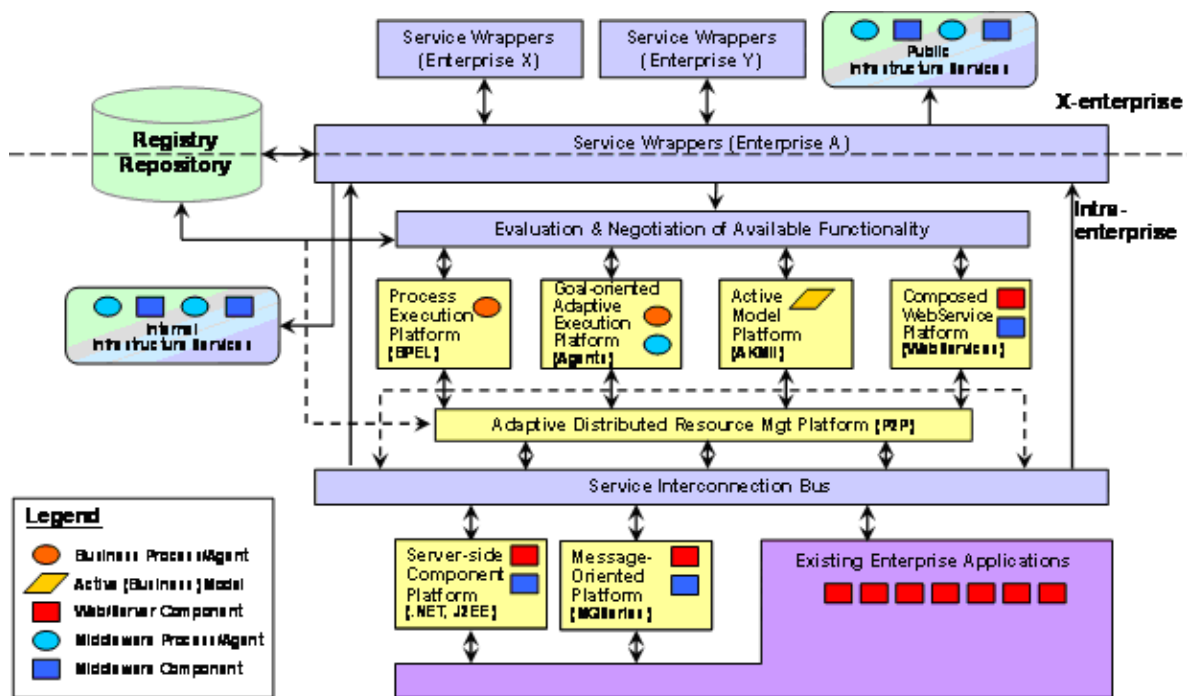
Another advantage is that extending the behaviour by adding a new plan for a specific task simply means adding it to the plan library for it to be executed at the next opportunity. Similarly, customizing the composition is facilitated since the different plans clearly structure

the alternatives of possible actions. Since the control structure is implicit, changes in a plan do not have impact on the control structure.

### 5.4.5. Integrated execution platform

Development of the integrated execution platform and corresponding infrastructure services will be one of the focus areas of action line A5 and A6. The service bus consists of three main components:

1. Service wrappers will provide a standardised way of accessing and using services. A first version of the service wrapper will be based on WSDL technology.
2. Evaluation & Negation of Available Functionality
3. Service Interconnection Bus provides middleware services for integrating the various execution platforms.



Services can be provided for internal use to support interoperability between business units within an enterprise, and for external use to support interoperability between enterprises. In addition to the middleware services provided by the service bus, there may be a need to develop or acquire specific infrastructure services within an organisation.

## 5.5. Information interoperability

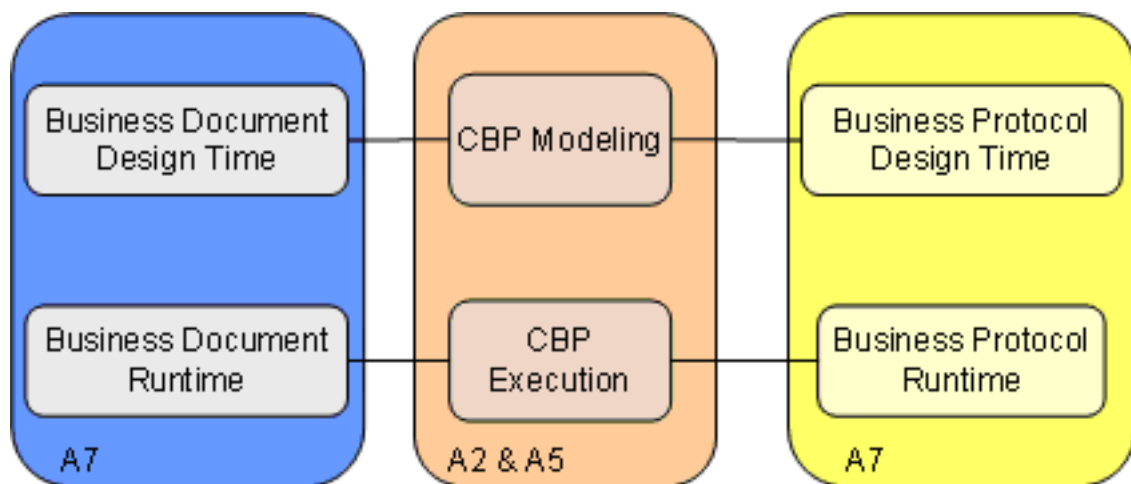
### 5.5.1. Introduction

In this section we describe the overall architecture of the ATHENA approach to the handling of business documents and protocols in the context of modeling and execution of cross-organizational business processes, which extends the semantics and ontologies approach to information interoperability.

We consider a business document as information entities that are exchanged between and referred to by business partners during the enactment of business processes. Depending on the stakeholder that models a business document; different representation for the business object can be defined:

- on the business level stakeholders will typically talk about business documents in the form of business relevant documents, e.g., a purchase order
- on the execution level stakeholders will agree on XML messages that are exchanged and whose payload contains the business documents

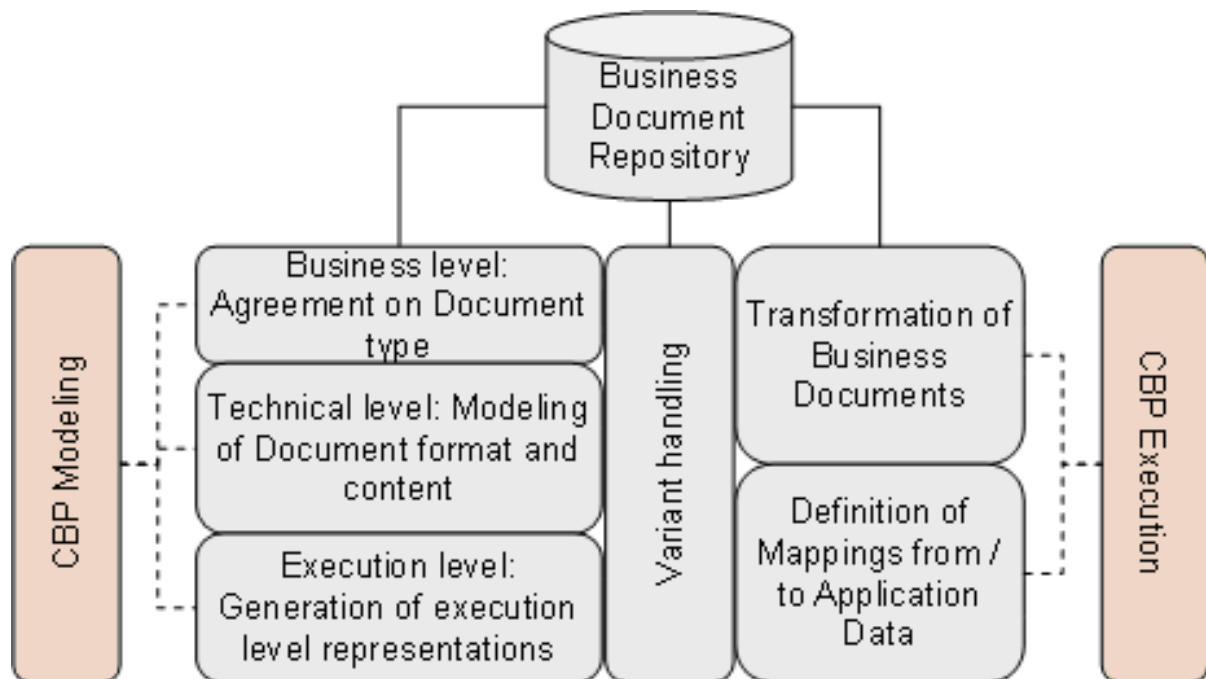
The following figure illustrates on a very high level the main building blocks of the overall architecture.



### 5.5.2. Design-time architecture for business documents

This section describes the elements of the design time architecture for business documents.



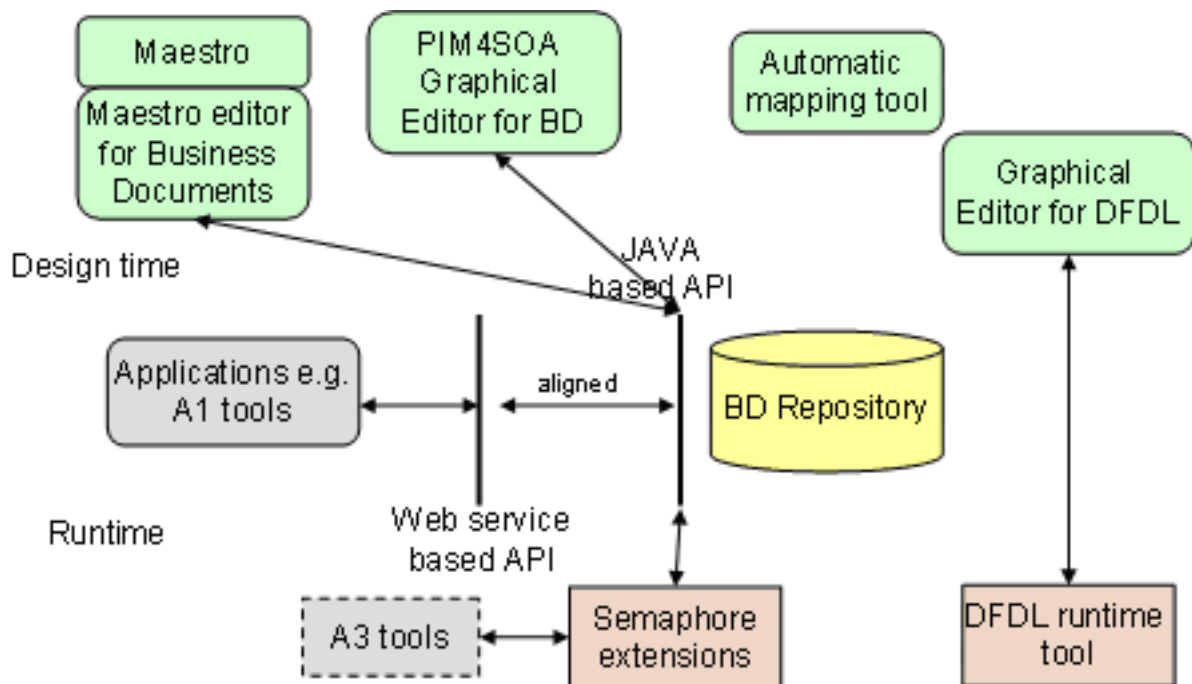


Modeling of business documents is closely linked to the modeling of the cross-organizational business processes defining the interaction between business partners. Thus, we also have to consider the different modeling levels as they are contained in the business process modeling framework. At business level the content of the business document is determined. The requirements are formulated from a business perspective and form the general structure of the business document. Through an assembly-based approach, components are utilized to construct concrete business documents from smaller components or singular items. The detailed instantiation of these items (e.g., "title") by assignment of further attributes (e.g., to provide semantics in form of a reference to an ontology) or by the restriction of the assigned data types is part of the technical level of business documents. The specification of the business document in a certain schema (e.g., UBL or OAGIS) in a certain syntax (e.g., XML or UN/EDIFACT) wrapped in business messages constitutes the execution level of the business document specification.

### 5.5.3. Business document and mapping tools

The approach is supported by a number of tools that complement the tools business process modeling and execution that have been developed in ATHENA. The figure below gives an overview of the different solutions and highlights their relationships.





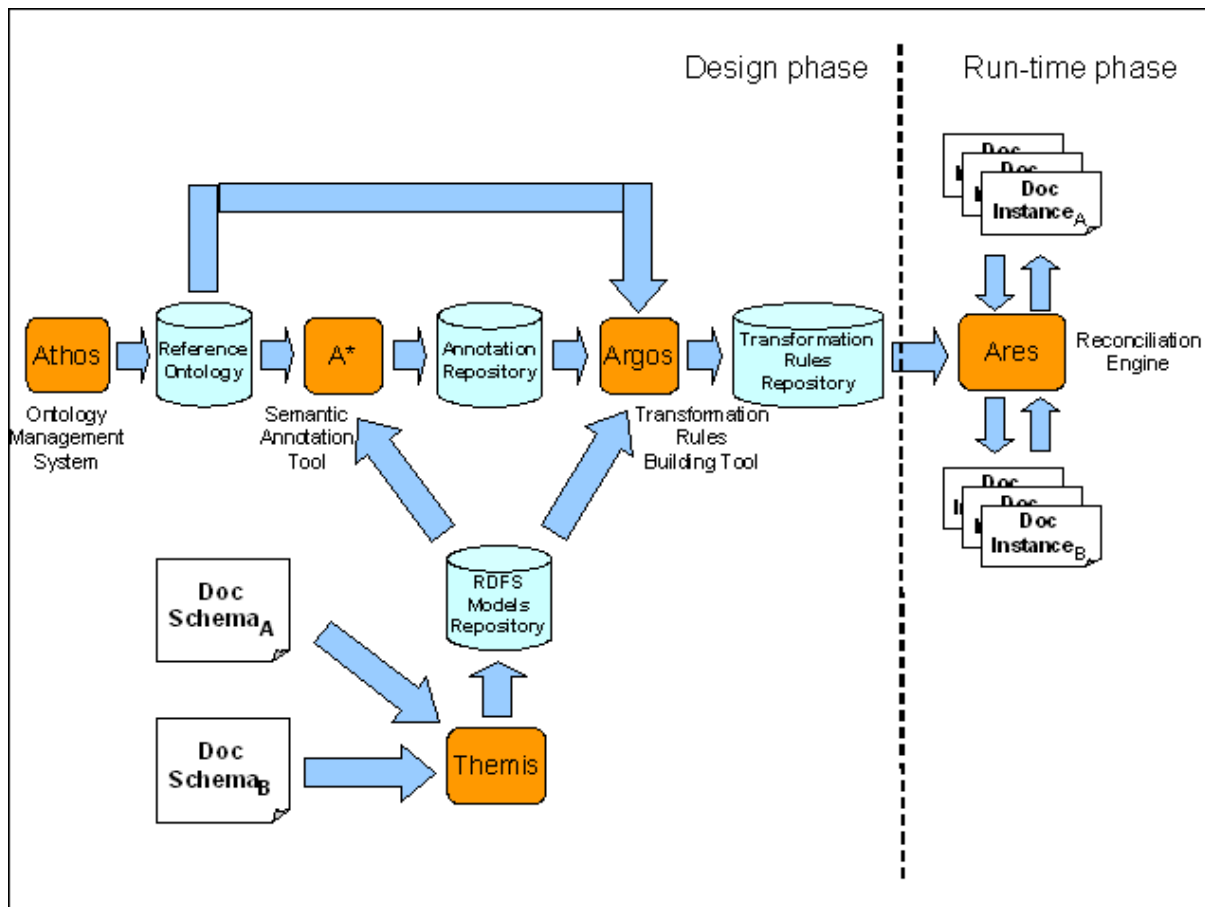
## 5.6. Semantics and ontologies

### 5.6.1. Introduction

One of the aims of ATHENA was to provide a basic set of semantic tools and services which can be used by other components in order to include semantic support for solving interoperability issues. Some of the solutions, such as the ontology management system, are more general purpose and can be used as basis for achieving different solutions, while other tools are oriented to solve specific semantic goals. In particular the semantic reconciliation allows to reconcile documents and messages, starting from a common ontology and semantic annotations, without the human intervention at run-time.

### 5.6.2. Semantic reconciliation suite

The figure below shows the ATHENA framework for semantic reconciliation as a whole and it clarifies the relationships between the different tools:



All the tools described have been developed within the ATHENA project:

1. **ATHOS:** It is the Ontology Management System. It provides a web user interface for helping the users in the process of building and managing reference ontologies.
2. **THEMIS:** It is the common RDFS repository. The other tools of the suite share common resources using Themis as central point and it provides basic functionalities for supporting the integration of the semantic components and existing SOA platforms. Both, A\* and ARGOS, use the repository for accessing the resources to be annotated and ARES retrieves logical links between message instances of the run-time level with all the needed resources of the design phase (in particular the rule sets related to particular message instances).
3. **A\*:** It is the semantic annotation tool. Semantic annotation aims at giving a non ambiguous meaning to digital resources and represents a conceptual correspondence between resources and concepts in the ontology. The semantic annotation process results in semantic annotation expressions stored inside the tool itself and used by Argos as starting point for the creation of specific reconciliation rules.
4. **ARGOS:** The usage of this tool is strictly related to the issue of semantic document reconciliation. Starting from the knowledge captured by the semantic annotations, the tool

provides a user friendly web interface for writing reconciliation rules. These rules can be used at run-time in order to apply forward and backward transformations among business documents.

5. **ARES:** It is the only tool of the framework which works at run-time. Its objective is to provide the semantic reconciliation service to external run-time environments. It is built for supporting SOA environments and the integration in pre-existing legacy systems.

### 5.6.3. Semantic enhancement for SOA and UMT2OWLS

---

UMT2OWLS is a tool to model visually the semantic enrichment of Web service interfaces description. It is based on the existing UMT-QVT tool. This extension allows to transform UML-based models for describing Web services directly in their WSDL and OWL-S representation. In this manner a real model oriented development is achieved. Obviously the resulted WSDL and OWL-S documents can be used in a Web services registry for the description of the service interface including all the necessary business and technical information.

The development of the tool has involved two different tasks. The first has been the definition and the extension of the pre-existing ACE-GIS standard for the service description in order to incorporate specific functionalities for the definition of all the parameters needed by the OWL-S description. The second task has been the real implementation of the plug-in of the UMT tool in order to realize the automatic transformation in the computer understandable format (WSDL and OWL-S).

## 5.7. Model-driven interoperability

---

### 5.7.1. Introduction

---

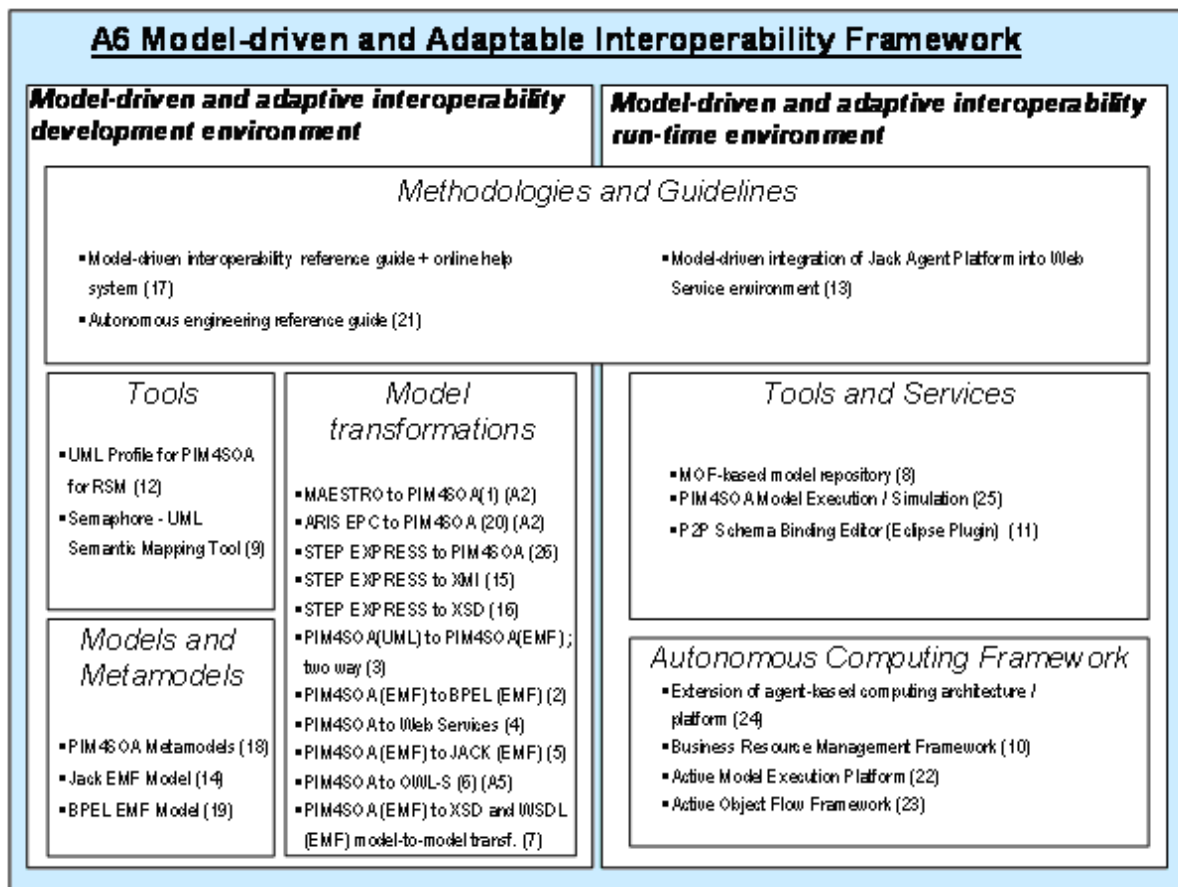
The objective of model-driven interoperability is to integrating principles of model-driven development and adaptable interoperability architectures.

- Model-driven development focus on design-time aspects of system engineering. Supporting methods describe how to develop and utilise (visual) models as an active aid in the analysis, specification, design and implementation phases of an information and communication technology (ICT) system.
- Adaptive interoperability architectures focus on run-time aspects of system engineering. Agent and P2P technologies enrich an ICT system with dynamic and adaptive qualities.

The model-driven interoperability approach has resulted in the model-driven and adaptive interoperability framework (Figure 138) which consists of two parts:

1. Model-driven and adaptive interoperability development environment
2. Model-driven and adaptive interoperability runtime environment

An overview of the framework is described below. For further details please consult the deliverable [ATHENA A6 2006b]./p>



### 5.7.2. Model-driven and adaptive interoperability development environment

The Model Driven and Adaptive Interoperability Development Environment provides a set of models, model transformations and tools that enable interoperability between modelling tools.

- The PIM4SOA meta-model is the primary vehicle for enabling tool interoperability. It provides a platform independent model of documents, services, processes and non-functional requirements.
- A number of model transformations are provided that allow business models defined in a number of modelling tools to be targeted, via PIM4SOA, at a number of different runtime environments. This allows different styles of business modelling tools to be independent of the eventual runtime environment rather than closely associated together as is most the case today.
- An example of such a tool has been provided by the UML Profile for PIM4SOA which allows business models to be defined by users who are familiar and comfortable with UML.
- In addition a UML semantic mapping tools is provided that UML users to develop

conversions of business data from one format to another. For example, converting the external form of a purchase order to an internal form understood by internal applications.

### 5.7.3. Model-driven and adaptive interoperability runtime environment

---

The Model Driven and Adaptive Interoperability Runtime Environment provides the ATHENA Autonomous Computing Framework (ACCF) and a set of runtime tools. The AACF framework consists of three integral parts, which are described in detail in this document:

- 1. Autonomous Service and Information Infrastructure:** This part addresses basic methods, tools, models, and protocols to support dynamic and distributed information sharing, provisioning, and management, as well as flexible self-organizing service environments. In this document, we describe two instances of components supporting an autonomous service and information infrastructure:
  - The P2P Business Resource Management Framework
  - The Active Model Execution Platform
- 2. Autonomous Behaviour and Process Infrastructure:** This section of the framework provides architectures, methods, tools, and protocols geared to describe and enable dynamic system behaviour and adaptive business process composition and management. This document illustrates two results achieved in ATHENA to this end:
  - A meta-model, method and tool to support the modelling and execution of business processes devised by software agents.
  - The Active Object Flow concept, which extends the Active Object Space by a process description based on UML activity diagrams.
- 3. Autonomous computing engineering reference:** It was recognized early in the project that a methodological framework was needed to support the design of systems relying on principles of autonomous computing. The reference guide describes some key aspects of such a design methodology, resulting in a reference guide for designers of autonomous systems, relying on holonic multiagent concepts.

## 6. Interoperability profiles

---

### 6.1. Interoperability profiles

---

#### 6.1.1. Introduction

---

ATHENA partners defined the concept of *interoperability profile* at the very beginning of the project. They were used by the different ATHENA projects, in order to support collaboration between ATHENA partners and in order to be refined and improved for usage as part of the ATHENA Interoperability Framework. But from concrete experience of usage of profiles in one hand, from piloting activities results analysis in the other hand, it appears that the profiles

as initially defined were not always appropriate.

### 6.1.2. Interoperability profile definition, usage and lessons learnt

The concept of an *interoperability profile* was initially defined when preparing the description of work of the ATHENA project [ATHENA 2003], on basis of a categorisation per application domains (initially *Supply Chain Management, Product Portfolio Management, Collaborative Product Development, and e-Procurement*) and industry sectors (initially *Automotive, Aerospace, Furniture and Telecommunication*).

The profile concept aimed first to facilitate coordination and collaboration between ATHENA projects and ATHENA involved communities. It aimed second to be a reusable component of the ATHENA Interoperability Framework, validated through concrete usage and lessons learnt from its application by the ATHENA partners through research and piloting activities.

Interoperability profiles were used by piloting activities:

- as categorisation of requirements that supports commonality analysis and generalisation of requirements
- as context element to establish relationships between business needs/interoperability issues and solution components

Interoperability profiles were used by research activities in order to package integrated set of solutions. An interoperability profile consists of interoperability guidelines, specifications and solutions on the conceptual, the applicative and technical level, specifically selected, grouped and configured for the enterprise.

Profiles are derived by the ATHENA Interoperability Framework and can be used to support an industrial sector community through establishment of a Web-based portal supporting each application domain's viewpoint for a given industry sector. They can also be used to support a community dedicated to an application domain, through establishment of a Web-based portal supporting each industrial sectors' viewpoints. Finally they can be used to allow these different communities to share their experience and efforts to address common interoperability issues and to reduce development of similar and overlapping solutions that are themselves a source of non-interoperability.

ATHENA initial aim was to create four ATHENA Interoperability Profiles (AIPs) for selected scenarios covering the application domains as described in the table below.

<b>Business domain Industry sector</b>	<b>Supply-chain management (SCM)</b>	<b>Collaborative product development (CPD)</b>	<b>Electronic procurement</b>	<b>Product portfolio management (PPM)</b>

<b>Aerospace</b>	Where stable supply chains and dynamic supply networks will be considered	In which cross-functional and cross-organisational teams collaborate in product development.	Focusing on electronic purchasing and selling of goods and services.	Focusing on project classifications, selection, prioritisation, and resource allocation.
<b>Automotive</b>				
<b>Furniture</b>				
<b>Telecom</b>				

For each of these application domains, the proposed approach was the following:

- to identify domain-specific dictionaries, thesauri, nomenclatures and coding that will have impact on the development and usage of domain-specific reference ontologies
- to also take into consideration industry standards, and legislations and regulations given by the national legislative assemblies
- to prioritise, for each of these domains, specific software concerns and aspects differently for each specific context, as a specific context always required custom-tailored views or models
- were more related to a one-to-one collaboration with specific non-open solutions (in such a case usage of standards is useless, and it is required to have a deep and detailed analysis of business process, objects and specific applications – without being able to use any standards)
- were addressing an industrial sector or a domain where no standards are defined nor used
- were issued from a context where organisations were not using any standards
- were issued from a context where it was just required to compose services for a non-repeatable process

For these kinds of situations, the more appropriate technologies and solutions are those related to fast integration of legacy technologies (such as enterprise application integration (EAI), CORBA and Web services) and applications without any support of well-structured automated business process (such as workflow process model). Within the context of an enterprise, flexibility, fast development and reconfiguration are properties that are very important.

Consequently, AIF had to adapt the initial approach with results coming from piloting activities and their analysis in order to propose guidelines for profile development, considering that some other important factors may impact the mapping that were not already discovered. So the proposed approach should allow to discover and to enrich continuously the proposed profiles.

## 6.2. Guidelines for interoperability profiles

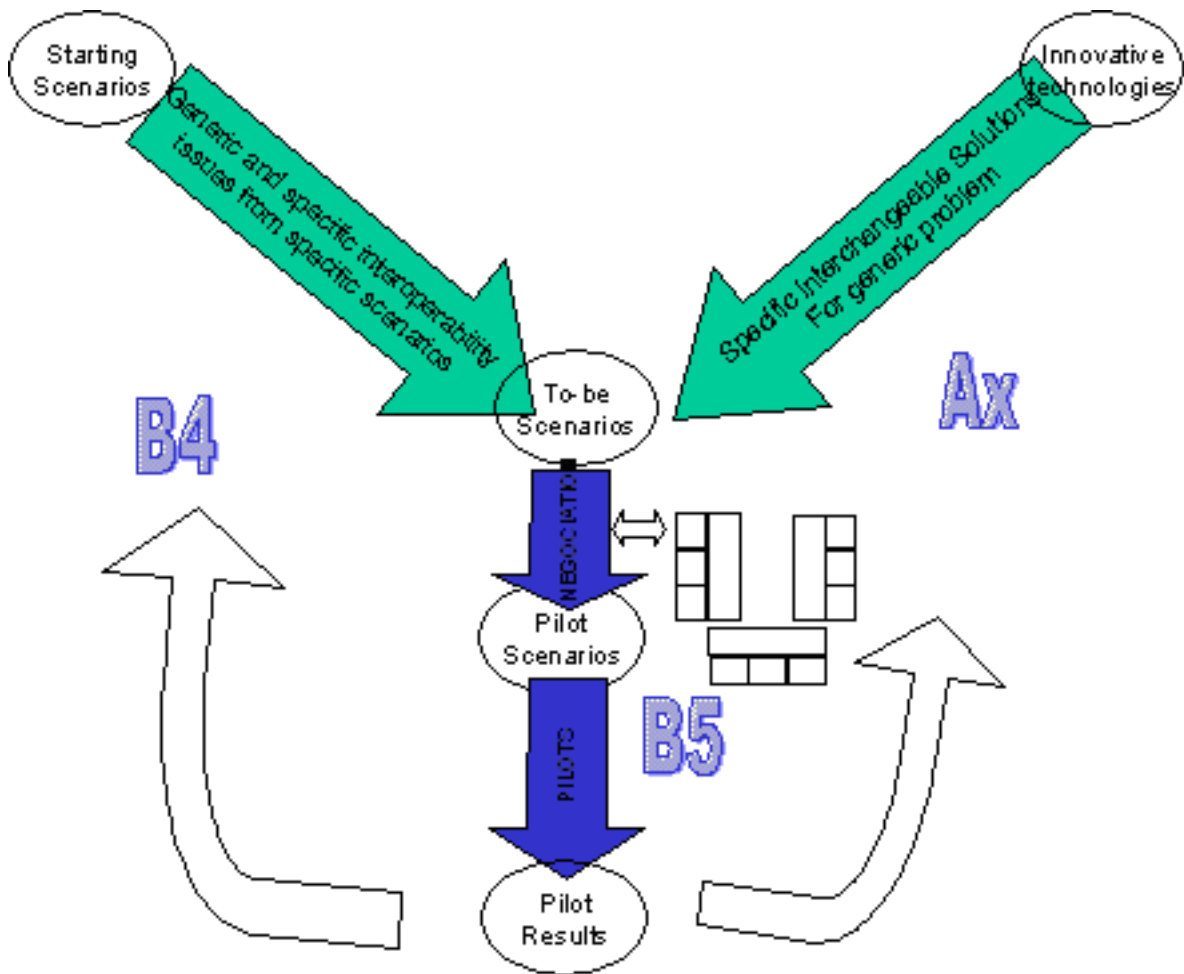
### 6.2.1. Procedure

The revised process followed within ATHENA is outlined in this section. An interoperability profile means a collection of ATHENA generic solutions that work together to solve a set of meaningful interoperability generic problems (interoperability issues). In addition, requirements were categorized using initial classification based on industrial sectors (e.g. Automotive, Aeronautic, Telecom or Furniture) and application domains (e.g. Product Development, Portfolio Management, Supply Chain Management or e-Procurement).

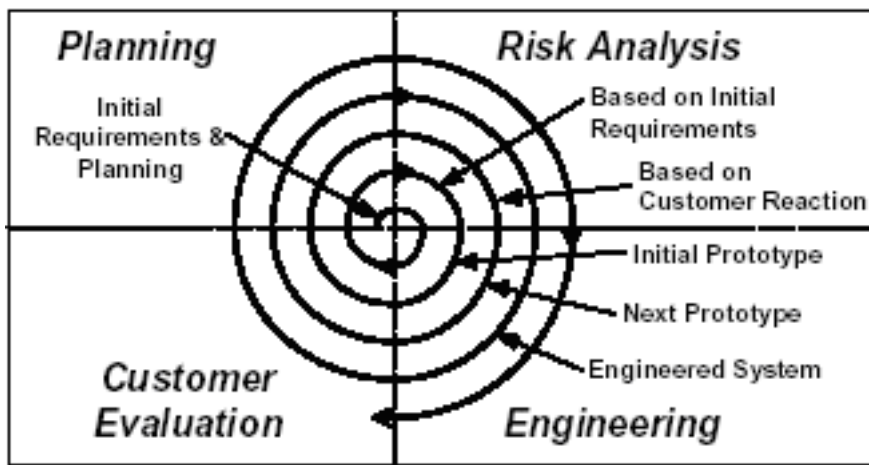
From the experience gained within ATHENA, in conjunction with some other standardisation initiatives, it appeared that a relevant approach to define profiles for interoperability of enterprise applications should be established through program iterations, and distinguishing three main phases or steps that are independent of iterations.

During the first step, a bottom-up approach for initialisation of profiles, in a second step a top-down approach for validation and improvement of profiles. As soon as the profiles are robust enough, the third step consists in a pattern-like approach allowing effective usage of knowledge gain during the two first steps to fasten identification and validation of relevant solutions for new business cases and scenarios. Robustness is related to maturity of technologies but also to maturity of the network in charge of the interoperability profiles and of its members.

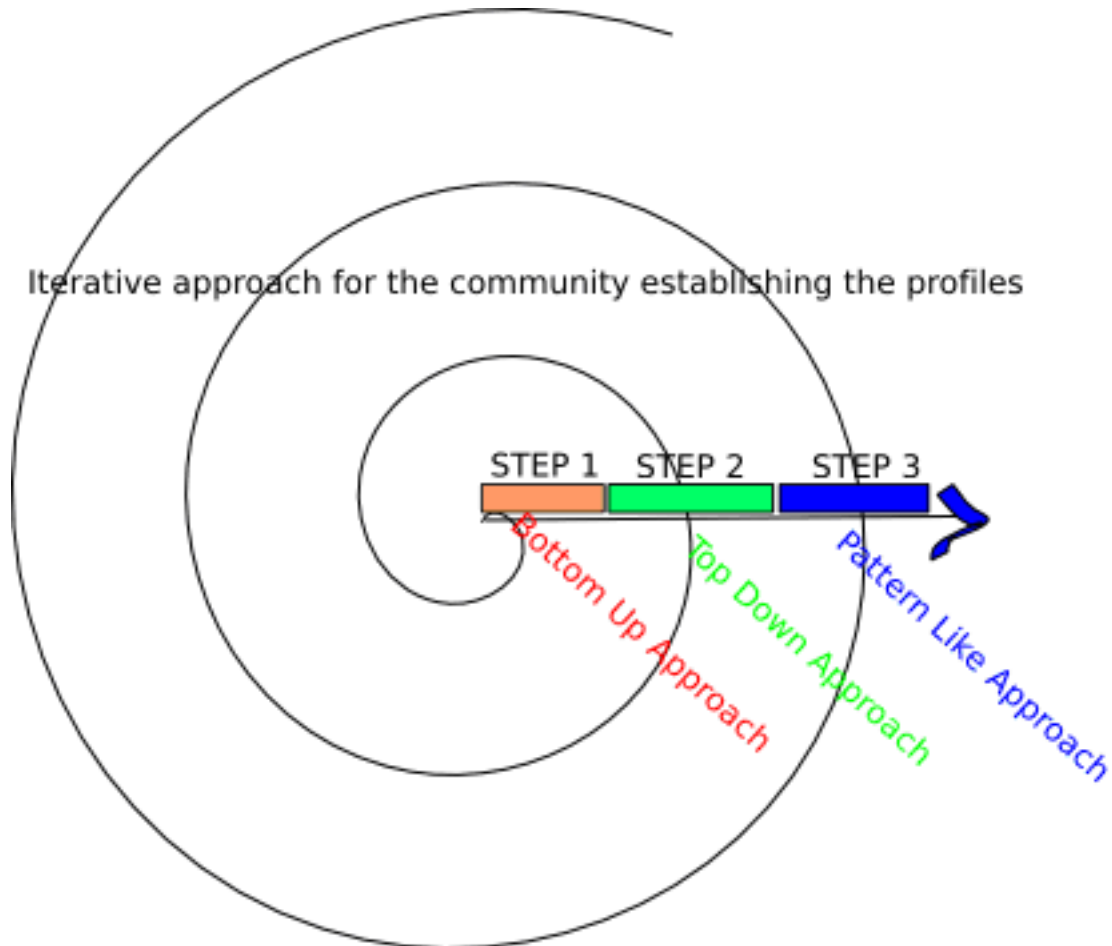




The next figure illustrates the generic evolutionary model use within the program and that is probably followed by any initiative dealing with interoperability of enterprise applications.



The next figure illustrates the profile development steps in a generic way by a community involved in a program implying several iterations. See [ATHENA B4 2004] for more details on programs versus projects and their impact on interoperability.



#### 6.2.1.1. Step 1 (bottom-up approach)

The four initial pilots performed interoperability problem analysis which resulted in business needs and interoperability issues. Since the ATHENA Knowledge Base was not available in the beginning some "ad-hoc" or "rules of thumbs" were used to pick the selected ATHENA generic solutions to use with related specific or ATHENA solutions. These results, given that they are successfully integrated and actually solve the interoperability issues of the pilot, defines an initial ATHENA Interoperability Profile (AIF) for the specific pilot based on:

- A matching sets of generic needs and generic solutions
- Business scenarios sets with similar set of generic needs and for which it is possible to use same set of generic solutions and corresponding concrete and specific solutions

Establishment of such sets is based on similarity/dissimilarity analysis proposed by the ATHENA B4 project.

Such an exercise is valuable only if based on inputs from experts on interoperability. In addition, to take advantage of already existing frameworks dealing with interoperability is particularly important due to in one hand limited resources of the ATHENA program, in the other hand to already existing robust and consensual solution components belonging to different framework for interoperability hosted by different organisation consortia. Some examples to consider are for example:

1. Object Request Broker (ORB) based on OMG's specification [OMG] and implemented through available commodities on the Web (e.g. Java JDK and JRE, open source ORBs like Orbit).
2. LDAP implementing X500 standard [ISO 1993] and with numerous open source commodities (e.g. OpenLDAP).
3. Workflow systems according WfMC standards [WfMC] (e.g. Enhydra Jawe and Shark, Bonita).
4. Application servers based on CORBA Component Model [OMG 2006b] and Java Enterprise Edition [Sun], with existing solutions as well as commercial of the shelves (e.g. IBM WebSphere) and open source industrial platforms (e.g. JBoss).
5. ISO STEP application protocols [ISO 1994] and tools supporting the standard (e.g. Dassault Systèmes' CATIA and EPM Technologies' Enterprise Data Manager).
6. W3C Web services specifications [W3C 2004c] including OASIS's BPEL [OASIS] and available implementations (e.g. ActiveBPEL).

Initialisation of interoperability profiles should be based on those that already exist. Within the ATHENA project, several frameworks of reference were used. For industrial users, OMG and ISO STEP frameworks and the way they are organised were considered as model of reference when establishing the description of work. In particular, the idea of similarity/dissimilarity is coming from the process related to elaborate application protocols, which are in a first step elaborated by a group of expert in a given field, and then mapped with already existing common resources for all the STEP application protocols.

It was possible to establish relationships through some bindings provided by user-oriented frameworks. Within the scope of ATHENA, it is important to point out that it was mandatory to be independent of any existing framework, and to be able to use existing and relevant frameworks simultaneously in a federated way. One important identified gap was existence of overlapping and incompatible standards. In addition, consideration of enterprise modelling, knowledge modelling and semantic mediation in order to address interoperability in a holistic way implied to extend already existing frameworks with new aspects not yet considered.

#### **6.2.1.2. Step 2 (top-down approach)**

---

Using the ATHENA Knowledge Base we can (semi-)automatically perform the requirements to solutions mapping method. This will generate new solutions that are possible candidates to

be added to the initial profile for the pilots. Whether these solutions could be added or not must be validated by the solutions providers in combination with the pilot users.

New pilots can be compared to the previous pilots using the knowledge base. From the knowledge base it is possible to identify generic solutions and corresponding specific or ATHENA solutions that should correspond to the new pilots. If it appears that this is not the case after performing an evaluation, it implies that the sets of generic needs and specific needs and associated relevant contextualisation elements should be extended or improved in order to provide a more appropriate matching (following an iterative approach). This should be continued until a robust framework is obtained that can be applied for numerous different pilots.

### **6.2.1.3. Step 3 (pattern-like approach)**

---

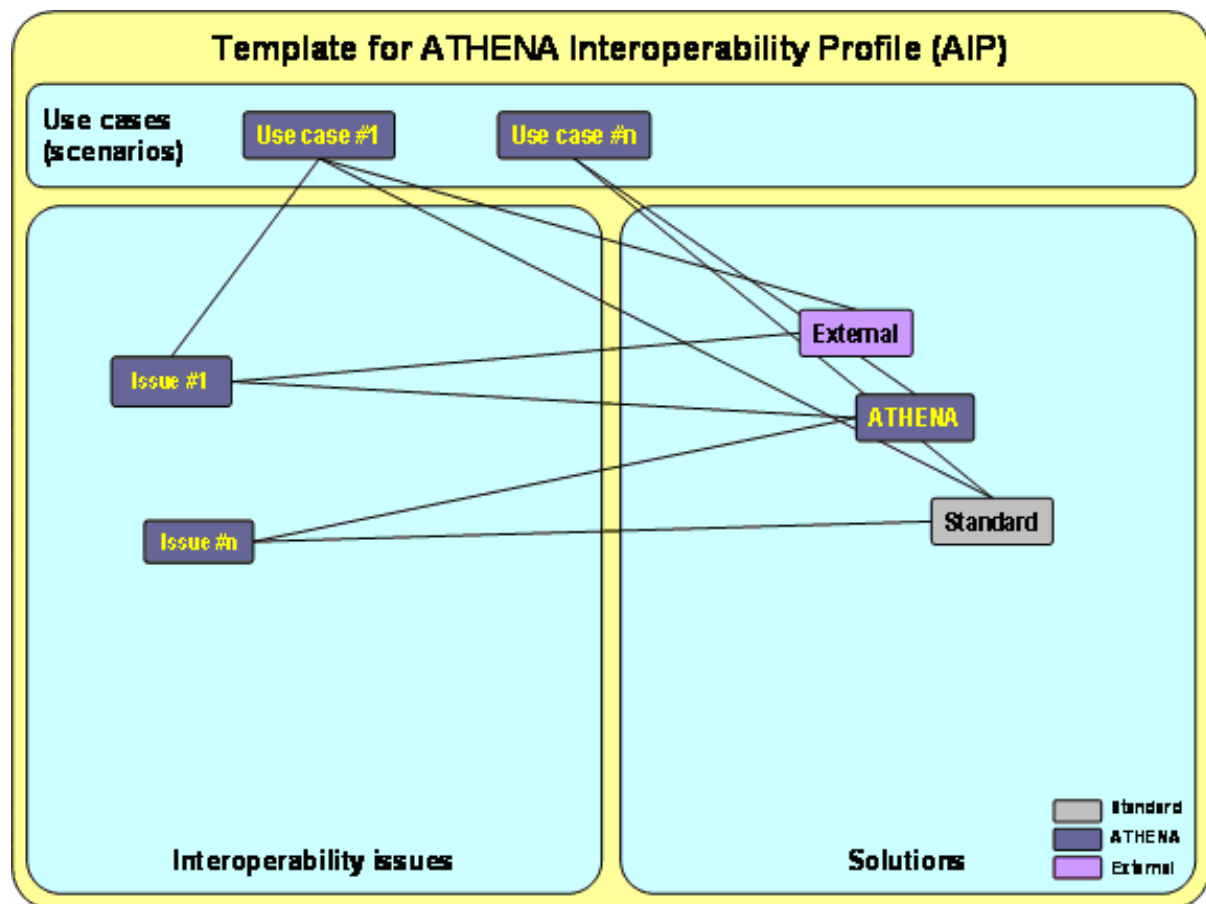
Rather than looking at individual pilot needs we group the issues into meaningful pieces of interoperability problems (that are applicable to different industry sectors and application domains) through the generalisation process defined in Dynamic Requirement Definition Process. Links between specific requirements and ATHENA requirements are tracked using a knowledge base. Then the already existing generic requirements # generic solutions # specific solutions mapping can be reused for identification of generic solution establishment and identification of existing interchangeable concrete and specific solutions.

This will generate "solution patterns" that are usable in different sectors/domains. Of course, as the ATHENA project considered a limited set of business scenarios with limited resources, obtaining robust "solution patterns" imply to continue, on the basis of the method established by ATHENA and starting from the existing knowledge base, to still perform several iterations within communities that will continue the process started within ATHENA. It could be done by the Enterprise Interoperability Centre (EIC) in collaboration with existing networked communities. It is for example targeted within the exploitation plan of some industrial partners (e.g. EADS CCR that will promote the approach and support its continuation within the EADS Group and within projects in the Aerospace and Manufacturing sectors).

### **6.2.2. Templates and supporting tools**

---

During the ATHENA project, a basic template for interoperability profile was used, allowing formalising relationships between use cases and interoperability issues, and then relationships between issues and concrete solutions.



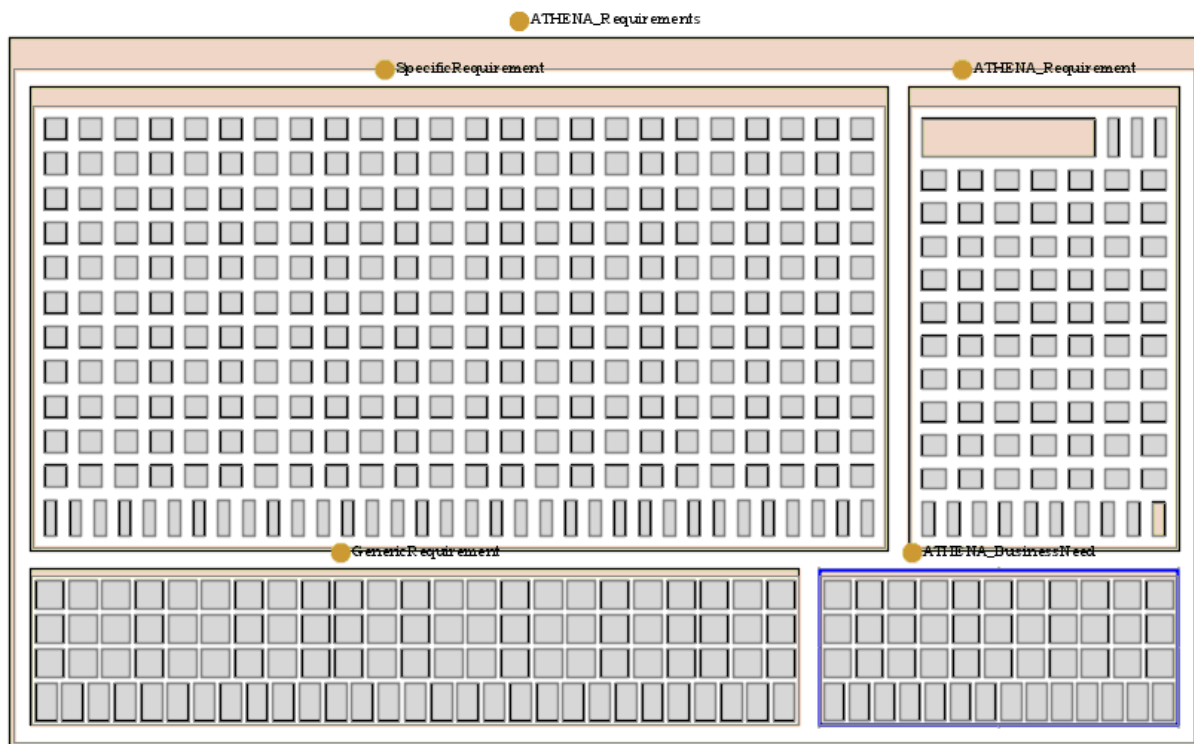
During the final piloting activities that were related to integration, it appeared that such a model should be improved for next iterations, in order to reflect independence of industrial projects and commercial software products development. Other issue was related to composition of solutions where each enterprise solution is a composite solution that aggregates several unitary solutions. An example is the Networked Collaborative Product Development (NCPD) platform that integrates model-generated collaborative workplace, semantic mediation, workflow interconnection, etc. Semantic mediation itself was sub-divided in mapping solution, transformation description solution and transformation execution solution. Finally, as each unitary solution component can be developed independently, without targeting initially any composition, it was important for the ATHENA Interoperability Framework to propose a set of generic solution components easy to integrate within the whole framework, and to select existing interchangeable specific concrete solutions provided by ATHENA or by any solution providers outside the ATHENA project.

This is why the ATHENA Knowledge Base (also referred to as the Harmonisation Model) in Protégé, initially developed within the context of ATHENA piloting activities for Aerospace and then shared with the other partners, proposes more sophisticated profiles based on the Dynamic Requirement Definition (DRD) process and experience gained from pilots within

ATHENA. It provides formalisation of the previously described concepts in order to organize information captured during the project.

The ATHENA Knowledge Base contains the following information related to the problem space:

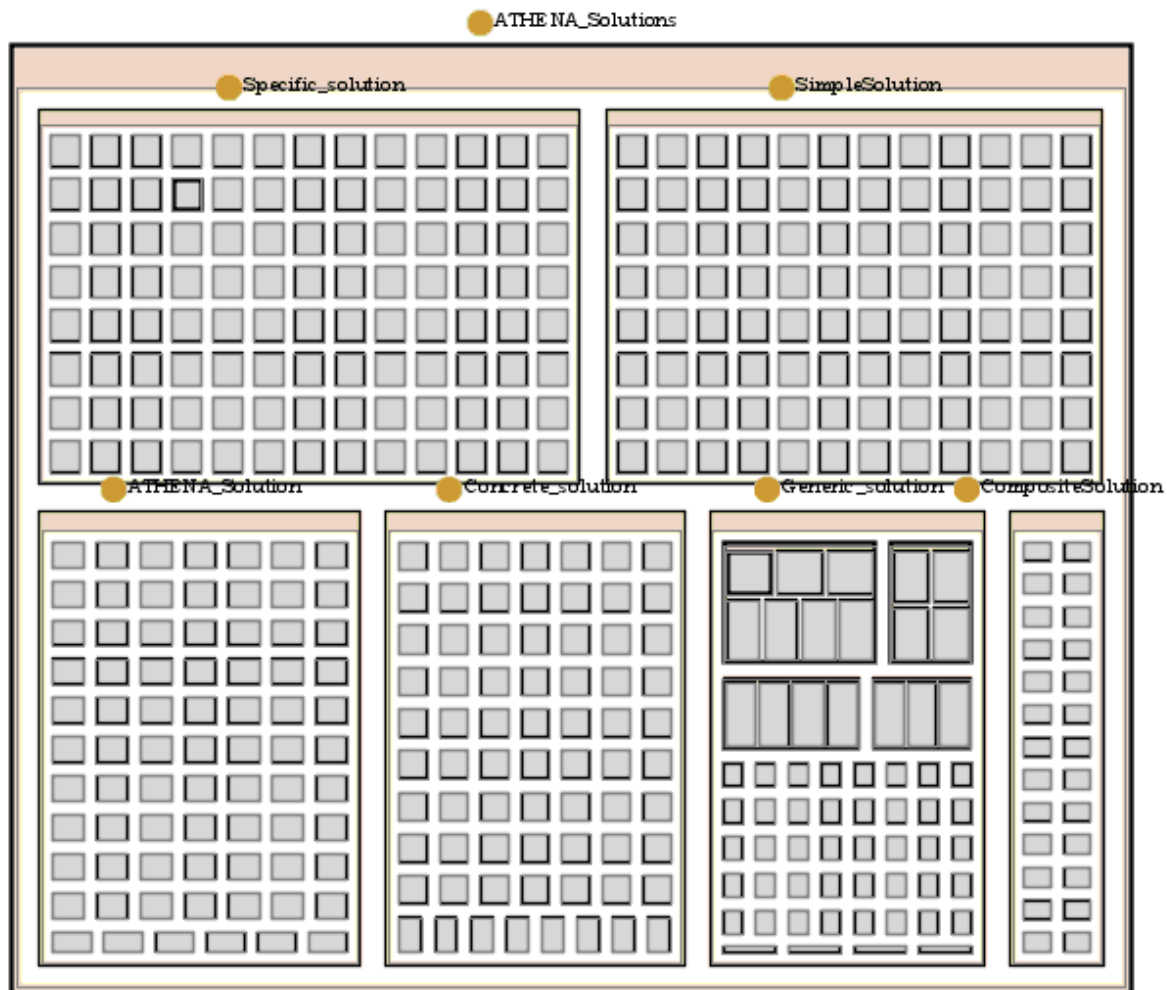
- Specific requirements extracted from business scenarios
- Generic requirements extracted from solution providers clients
- ATHENA requirements extracted from analysis within B4, that aimed to factorize common requirements related to similar interoperability issues
- ATHENA business needs extracted from abstraction of ATHENA requirements for elaboration of the ATHENA Interoperability Framework



The ATHENA Knowledge Base contains the following information related to the solution space:

- ATHENA solutions: Solutions, generic or concrete, simple or composite, and that were developed through ATHENA (i.e. using ATHENA resources).
- Generic solutions: A generic solution is a family of solutions, defined from a functional point of view by the community, and for which several concrete implementation may exist. The functional may be well formalised, for example by means of a standardised specification.
- Concrete solutions: Solutions that are corresponding to a given implementation of a

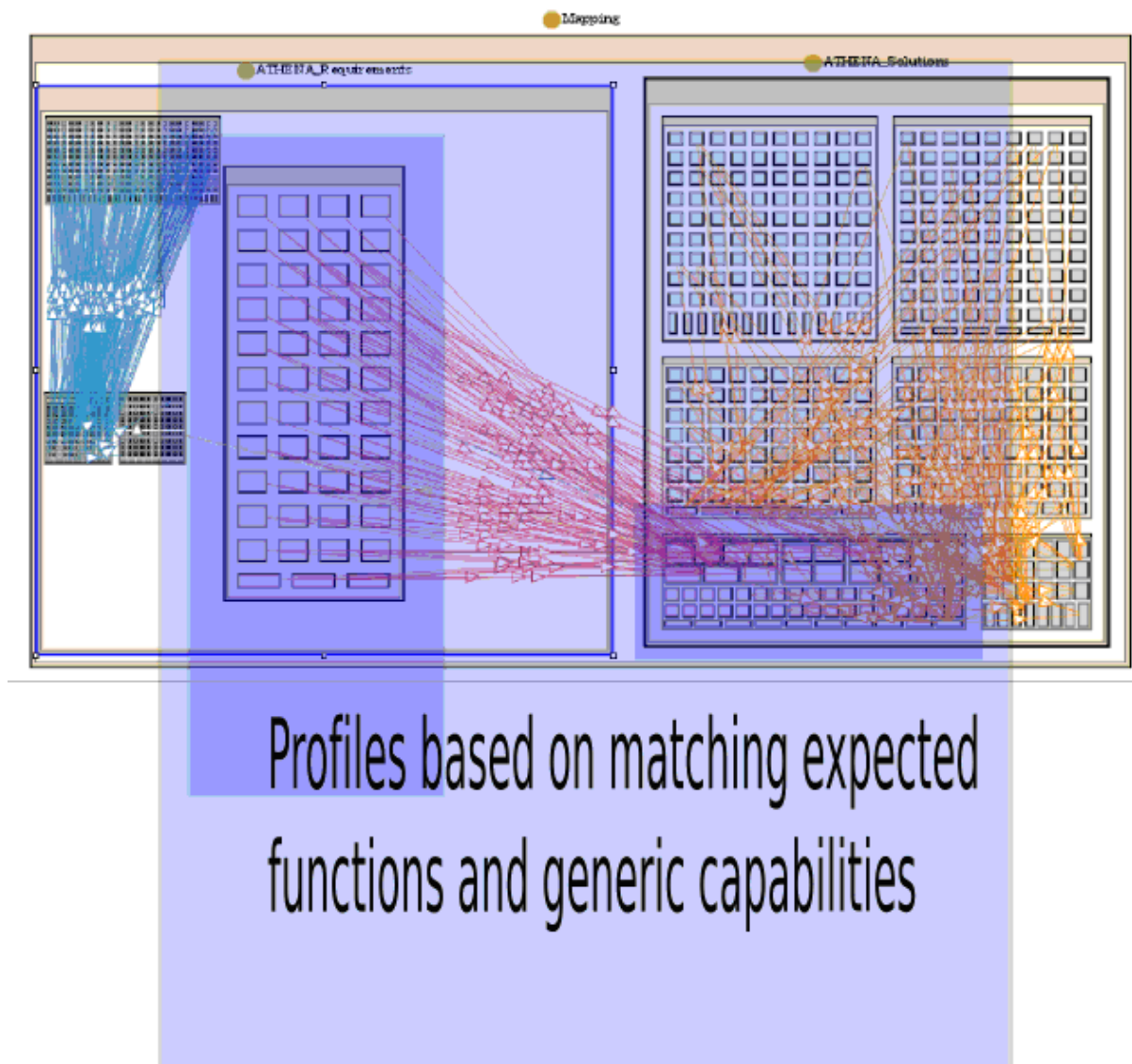
- generic solution, simple or composite.
- Simple solution: A simple solution is a unitary independent solution component that can be packaged, deployed and used alone.
- Composite solution: A composite solution is a solution that is obtained by composing several simple solutions.
- Specific solution: A solution component coming outside of the ATHENA project.



The interoperability profiles can be then established:

- matching expected generic functionalities (ATHENA business requirements) and generic abstract solutions, that will constitute after several iteration robust profiles for needs and solutions
- providing the mechanism for generalisation of specific requirements (abstraction defined in B4)
- providing mappings between generic solutions and concrete specific or ATHENA solutions (performed during integration in B5, with support of A4)





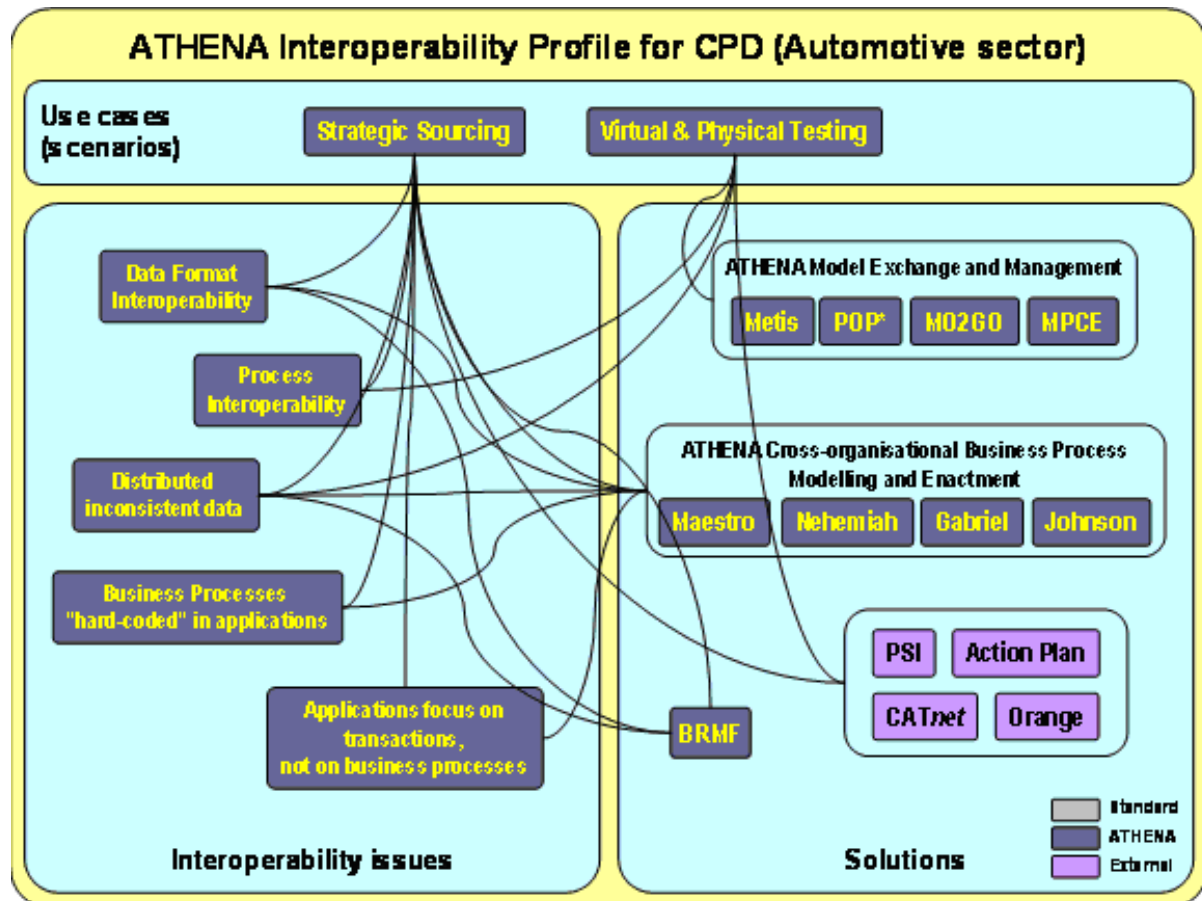
### 6.3. Collaborative product development (CPD) profile

#### 6.3.1. Collaborative product development (CPD) pilot

The Collaborative Product Development is founded mainly on four processes that have been employed separately as use cases for the research and pilot for the assessment of the Athena results. The core process in CPD is the Target Setting, an OEM-internal technical and decisional process that guides all the relationships with the other processes. In chronological order, the pilots that have been developed and executed involve the following sub-processes:



- Target Setting and Virtual & Physical Testing
- Target Setting, Design and Strategic Sourcing
- CPD (overall)



### 6.3.1.1. Virtual and physical testing

The first portion of the Collaborative Product Development that has been analysed and in which a test case for piloting has been identified, is Virtual & Physical Testing, the process with which the OEM tests the prototypes of the car systems and supplier components. The results of these testing activities feed the refinement of the vehicle objectives in the Target setting process as well as in the design operations.

Business processes as well as rough data are exchanged between the OEM and the Test Supplier applications. The ATHENA solutions employed in this pilot are:

- POP\* – Enterprise model exchange specifications
- Metis – Enterprise Model Authoring Tool and its extension for POP\* format
- Mo2Go – Enterprise Model Authoring Tool and its extension for POP\* format
- Grai – Enterprise Model Authoring Tool and its extension for POP\* format

- MPCE – Modelling Platform for Collaborative Enterprises

In addition it has been necessary to integrate these solutions in the OEM environment, in particular with the legacy applications:

- PSI (Piano Sperimentale Integrato) – an application for supporting product performances and test management
- CATnet (Computer Aided Testing on the network) – a Web-based application for test management

### 6.3.1.2. Strategic sourcing

---

The second portion of the Collaborative Product Development considered has been the Strategic Sourcing, i.e. the process selecting the suppliers that will participate in the development and the production of the car.

The Sourcing process under examination is not only a mere supplier choice process, but it is strongly concerned with defining product specifications and product innovation. In fact, this strategic sourcing gravitates around the exchange of a particular document (Request for Quotation) that contains a product description and that is completed and improved during the process by means of the negotiation between the OEM and the first tier suppliers and also between the first and the second tier suppliers. But if we compare the collaboration between the OEM and the first tier supplier with the collaboration between the first tier supplier and the second tier suppliers we notice significant differences in the interoperability requirements. On the OEM side we find a process-driven environment, where the number of participating first tier suppliers is low, and the first tier suppliers rarely change. On the first tier supplier side, we find a number of second tier suppliers. Second and lower tier suppliers join and leave the environment very dynamically. The business processes must continuously adapt to these events, which results in an event driven collaboration paradigm, as opposed to the process driven paradigm on the OEM side. The pilot development has been realised by the collaboration between CR-FIAT and Siemens, acting as OEM and first tier supplier, also outside the ATHENA consortium. The ATHENA solutions used in this first piloting are:

- Mo2Go – Enterprise modelling tool and its extension format for Maestro
- CBP construct – collaborative business process definition
- MAESTRO – Enterprise modelling tool for collaborative business process
- Nehemiah – Process engine and simulator
- Gabriel – Task management tool
- Johnson – SOAP client/server
- BRMF – Business Resources Management Framework, a event and document-driven P2P platform

Besides these solutions, other applications are used in the pilot:

- PSI (Piano Sperimentale Integrato) – for product performances and test management
- ActionPlan – Suppliers data Repository (used by FIAT Purchasing department)

## 6.4. Networked collaborative product development (NCPD) profile

---

### 6.4.1. Networked collaborative product development (NCPD) pilot

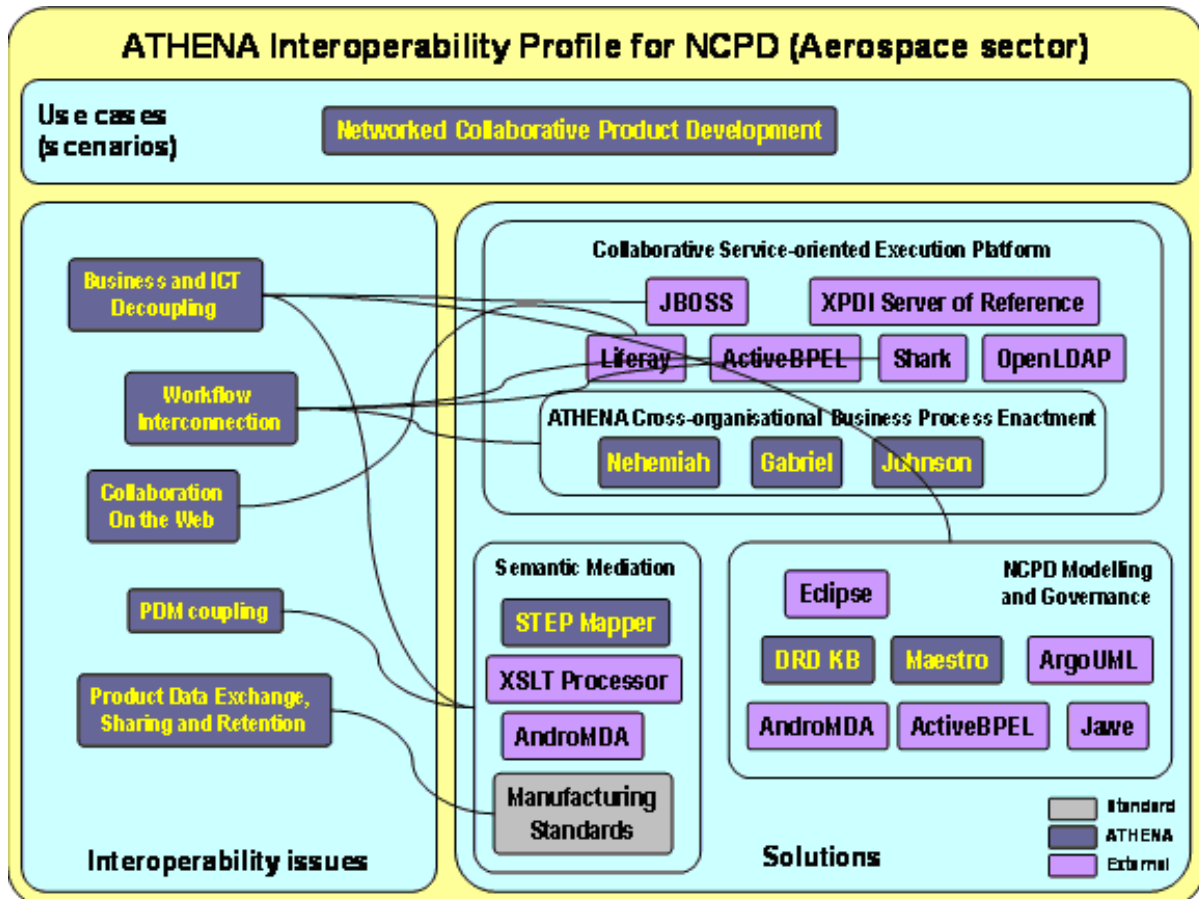
---

In the first year only smaller test pilots for each result were built. In the second year a more integrative pilot focusing working methods for Engineering Change Management (ECM) in a new approach to product design (Networked Collaborative Product Development) was started, utilizing the AKM solution to model-designed and -configured collaboration spaces. In the third year the EADS internal platform using many ATHENA and PLM services are so far implemented. The platform will be further developed, see the <http://nfig.hd.free.fr/ATHENA> Website. These piloted services demonstrate interoperability on most layers of architecture, from the user workplace to data sharing, but is not yet applying AKM beyond setting up the ECM collaboration space. It is due to a choice to start from robust software components based on standards for the execution platform (that is the foundation of any pilot). Several alternative concrete solutions were identified and benchmarked, in order to finally obtain a workable pilot, and in order to be able to illustrate added value of the solutions provided by ATHENA, in particular in term of innovation.

One of the alternative solutions for Collaborative Networked Collaborative Development platform is described in Figure 70 as an interoperability profile. It was designed in order to solve different important interoperability issues as Business/ICT decoupling, workflow interconnection, B2B collaboration on the Web, PDM coupling and finally product data exchange, sharing and long-term retention for all the enterprises belonging to a network. It is the reason why manufacturing standard (ISO STEP, PLM services and CM II) were selected as key components of the solution. In addition, the service-oriented execution platform is built on an application server based on EJB/CCM standard (JBoss), a portal based on JSR168/WSRP, BPEL (ActiveBPEL) and workflow engine compliant with WfMC specification (Shark), a directory for enterprise (OpenLDAP) and finally a cross organisation business process execution engine (Nehemiah + Gabriel + Johnson). A modelling and governance platform was also added in order to be able to model and develop complementary solution components for collaboration or functional extension. This platform was designed in a similar way as the execution platform, i.e. on the basis of free commodities based on ICT and modelling standards. Finally semantic mediation, horizontally or vertically, is ensured by AndroMDA, STEP Mapper and XSLT processor for this particular concrete platform. Some alternative solutions were also studied as the ATHENA Semantic Mediation suite and usage of Metis for active knowledge models for model-generated workplaces. It allowed through the piloting activities to identify added value of the solutions proposed by ATHENA but also to identify remaining gaps or gaps related to proposed solutions (e.g. simultaneous usage of syntactic and semantic Web solutions). From the proposed profile, any organisation willing to set up similar platform is able to take advantage of the profile in order to gain time for elaboration of interoperability issues to address and candidate solutions that are applicable.

The different combination with alternative solution components can't be reflected with the

profile template used here, but are available within the ATHENA Knowledge Base created with Protégé (DRD KB) containing requirements, interoperability issues and solutions with their relationships (DRD KB).



## 6.5. Electronic procurement (e-Procurement) profile

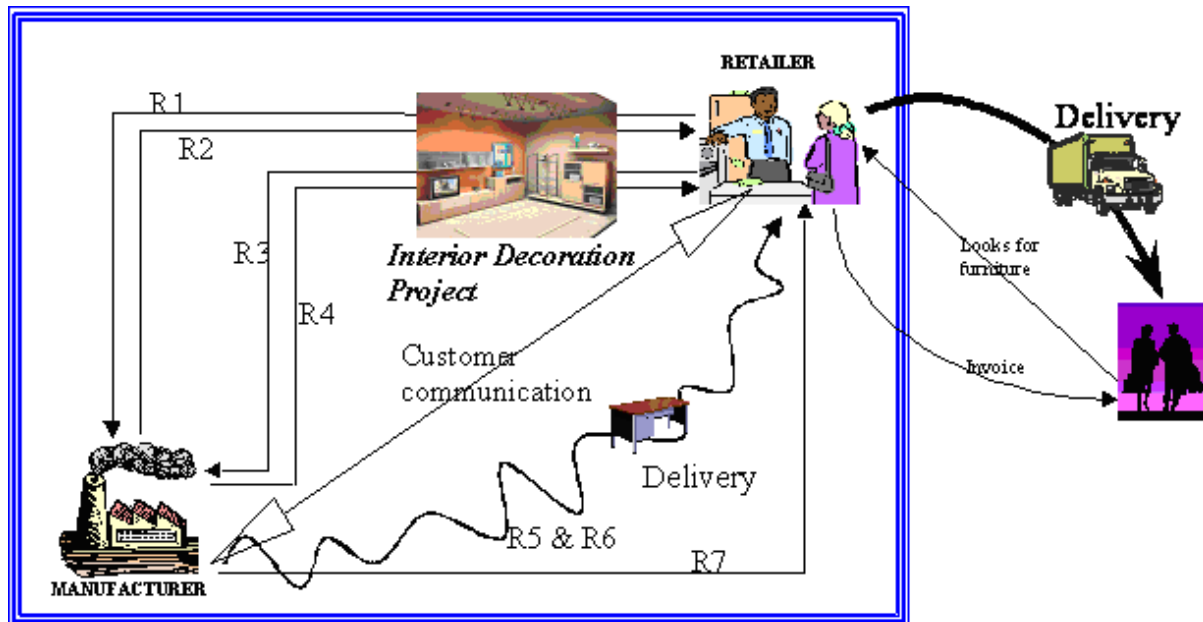
### 6.5.1. Furniture pilot

This is about improving the document flow, messaging and decision-making among core shareholders within the request for quotation (RFQ) and the order flow, including an interior decoration project, integrating Order Management System (OMS), Product Data Management (PDM), and a usual accounting systems for order handling, delivery and invoicing. The Infrastructure architecture is composed of a SAP suite of tools, the ATHENA semantic reconciliation suite and the STEP transformation tools, representing a typical order-supplier CBP with control of business document flows.

The furniture procurement scenario, see figure below, involves four stakeholders: furniture

manufacturer, raw-material supplier, retailer and customer. Although the document flow is started by the retailer, the action starts when a customer looks for furniture at the Retailer's facilities or website. The customer browses the catalogue and asks the salesman to obtain a quotation for the desired furniture products.

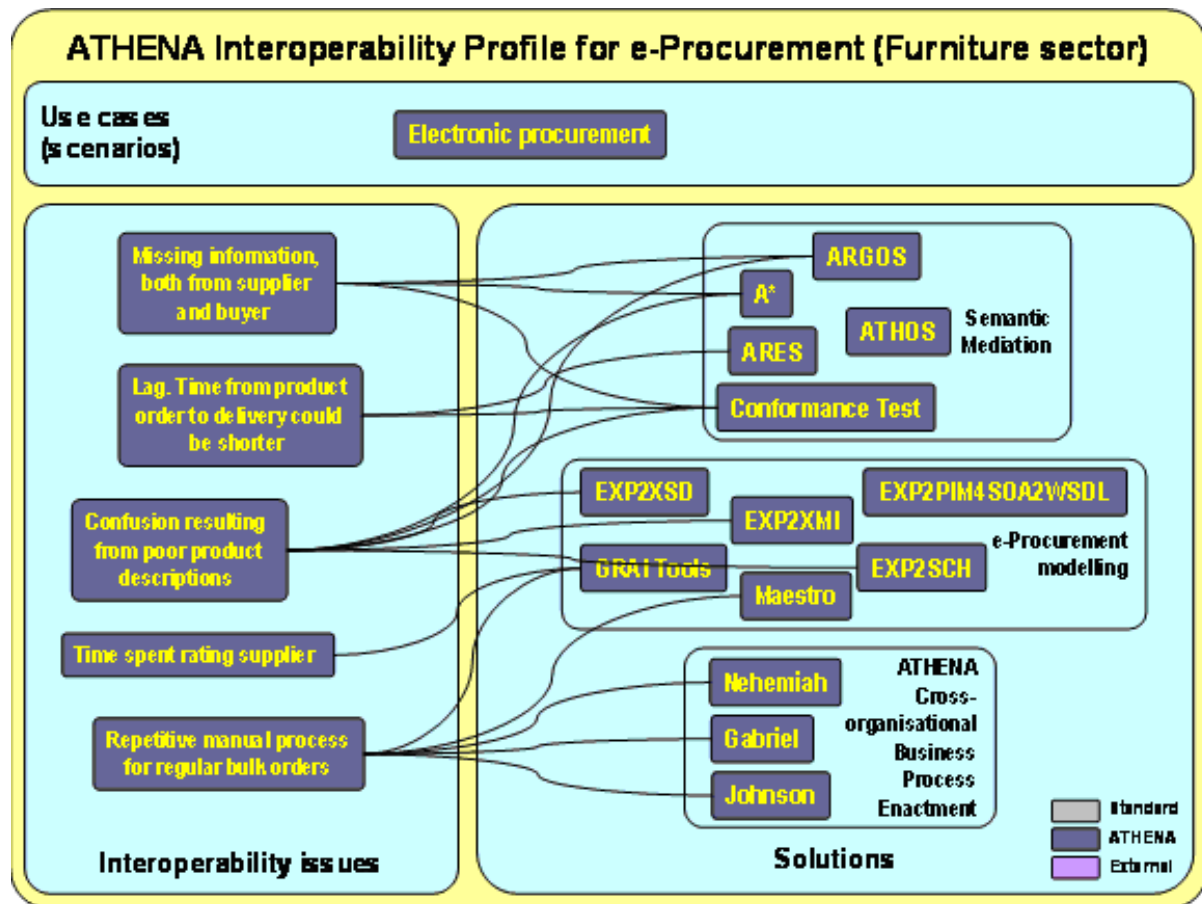
- The Retailer starts the procurement document flow process by sending to the Manufacturer the R1. Request for Quotation document to obtain the current product price list according to the customer desires. The information included in the R1 document is mainly composed by a product list with the corresponding specifications such as finishing, handles, fabrics and special measurements or cuts. Additionally, the Retailer might send the R1 document with an interior Decoration Project attached.
- Once the R1 document is at Manufacturer's side, specifically at the Sales Dept, the internal process regarding to the quotation process.
- To accept the quotation sent by the Manufacturer, the Retailer has to perform the R3. Order Document including a reference to the quotation. The Retailer might include also the decoration project as an attached draft of the ordered products. Once the R3 document arrives to the Sales Dept, the Manufacturer processes the document.
- At this moment, Sales sends to the Production Dept the list of ordered products. This department plans the product manufacturing by calculating the amount of raw material that is necessary to produce the goods. If there is any lack of raw material, Procurement Dept is asked for it.
- Eventually, during the manufacturing process, the Retailer might start a Customer Communication process to know the current status of the order.
- Once the product is manufactured, the different parts of the furniture are packaged. The packages are identified and sent to storage waiting for the shipping and being prepared to be loaded into the transportation.
- While this process is being performed, the Administration Dept prepares the last documents to be sent to the Retailer: R5. Delivery Note and R7. Invoice documents. Additionally, the Production Dept prepares the R6. Packing List which is the product list contained in the R5 broken down in the different identified packages.
- The products are loaded into the transportation and sent to the Retailer accompanied with the R5 and R6 documents. The Retailer signs the R5 and returns it to the Manufacturer who finally sends the invoice (R7).



The different ATHENA solutions used in the implementation of the pilot infrastructure are shown in the figure below are:

- GRAI Tools: Tool to model the internal business processes of the stakeholders involved in the furniture scenario. During ATHENA a link to Maestro was developed which allows GRAI enterprise models to be imported into Maestro.
- Maestro: Tool to model a collaborative environment based on the work done by Enterprise Modelling tools which will help in the development and improvement of the Business Processes of each actor.
- Nehemiah: Tool to simulate the CBP model developed under Maestro.
- Gabriel: Tool to orchestrate the Process and the Messages.
- Johnson: System which will act as a kind of Outlook. This tool is going to act as a front-end of the different actors and will have uploaded the Web Services to receive and send documents.
- A\*: Semantic Annotation Tool used to annotate different instances of the same document to allow the reconciliation and the matching between different documents against the same concepts.
- ARGOS: Rules Generation Tool used to generate the run-time tools for checking and annotating documents.
- ARES: Run time environment. This tool will be uploaded into Johnson as a plug-in, so this will allow the reconciliation and annotation on run-time the documents. Additionally ARES will help in validating the semantic of the documents against the ontology.
- Conformance Test: Tool to perform the validation of the documents both syntactically and basic-semantically against the XML Schema documents. The CT tool, as ARES, will be uploaded into Johnson to allow all the users to check the validity of documents.
- EXP2XSD, EXP2SCH, and EXP2XMI: These transformation tools will be used for

transforming the EXPRESS schemas into a more machine-understandable language, such as XML



Apart from these tools, AIDIMA has used the following ATHENA solutions which are not related to any Interoperability Issue in particular. These tools are:

- ATHOS: ATHENA Ontology Management System to create and manage ontologies. Under the scenario it has been uploaded the furniture ontology developed under the funStep projects.
- EXP2PIM4SOA2WSDL: STEP transformation tool used for transforming the EXPRESS schemas into Web services descriptions.

## 6.6. Product portfolio management (PPM) profile

### 6.6.1. Product portfolio management (PPM) pilot

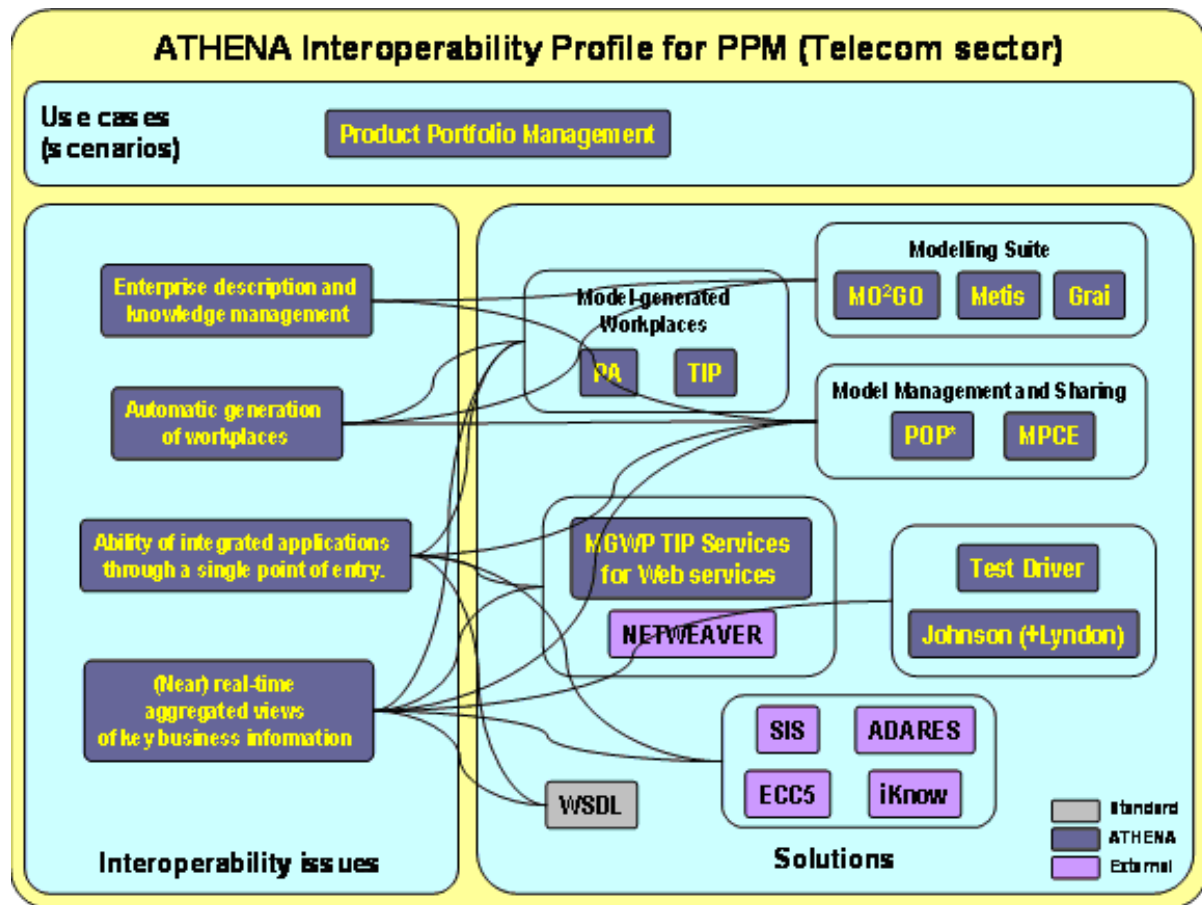
The pilot focuses Product Portfolio Management and product data sharing among key actors inside a telecom company. Charged with the task of selecting the right products and product

variants to produce products for a dynamic market and customer base, the company must find new ways of managing product design and engineering, and supporting customer communications. The pilot is implemented using a Model-configured, User-composed Platform and Services (MUPS) architecture to design a service layer with roles, views and model-generated workplaces and services, focusing the needs of the product manager.

The following ATHENA results and tools were used to deal with the identified interoperability issues and problems as shown in the figure below:

- POP\* for modelling the different aspects of the enterprise and generating the workplace through the models.
- Import/export of POP\* for modelling different aspects of the enterprise.
- MPCE for modelling different aspects of the enterprise.
- Transform ITM and BPM models to MEAF models for modelling different aspects of the enterprise.
- MEAF ATHENA extensions to facilitate Web services, task management, user interface modelling.
- In the generation of MGWP the following tools were used
  - MO2GO for the Process Assistant (PA)
  - Metis for the Troux Internet Portal (TIP)
  - TIP services for Web services for discovery of Web services and linking them to the models.
- Johnson (and Lyndon) for design, testing and deployment of services.
- Test Driver for testing services conformance and integrity.





## 7. ATHENA solutions

### 7.1. ATHENA solutions

#### 7.1.1. Composite solutions

A composite solution is a set of solutions that are grouped and configured together (packaging) to form a more comprehensive approach to solve meaningful set of (related) interoperability issues.

- [ATHENA Interoperability Framework \(AIF\)](#)
- [ATHENA Interoperability Project Support Suite](#)
- [ATHENA Model-Driven Interoperability \(MDI\) Framework](#)
- ATHENA Model Exchange and Management Solution
  - ATHENA Collaboration Space

- ATHENA Model-Generated Workplace Solution
- [ATHENA Modelling Tool Suite for Eclipse/RSM](#)
- ATHENA Process-based Interoperability Infrastructure
- [ATHENA Semantic Reconciliation Suite](#)
- [ATHENA Service-Oriented Interoperability \(SOI\) Framework](#)

### 7.1.2. Individual solutions

An individual is an ICT technology resulting from a specific research and/or development task in ATHENA that can be used to solve a specific interoperability issue. An individual solution can be either a model/language, a software tool or a method/guideline.

- [A\\*](#)
- [Active Knowledge Model \(AKM\) Execution Platform](#)
- [Active object flow \(AOF\) methodology](#)
- [Agent-based Computing Architecture and Design/Runtime Platform](#)
- [Autonomous Computing Engineering Reference Guide](#)
- [ARES](#)
- [ARIS \(Architecture of Integrated Information Systems\)](#)
  - [ARIS EPC CBP extension](#)
  - [ARIS EPC MPCE interface](#)
  - [ARIS EPC to PIM4SOA transformation \(XMI export\)](#)
- [ARGOS](#)
- [ASSERT](#)
- [ATHENA Event and Document Correlation \(AEDoC\)](#)
- [ATHOS](#)
- Baseline methodology for SOA
- [BPEL Metamodel Feature for Eclipse](#)
- [Business Issues Planner \(BIP\)](#)
- [BRMF \(Business Resource Management Framework\)](#)
  - [BRMF plugin for Eclipse \(P2P schema binding editor\)](#)
- [Capability Tables Loader \(CTL\)](#)
- [Conformance testing suite](#)
- [Eclipse](#)
- [EKA Metamodel Feature for Eclipse](#)
- [EKA to PIM4SOA Transformation Feature for Eclipse](#)
- [Enterprise Interoperability Degree Measurement \(EIDM\)](#)
- [Enterprise Interoperability Maturity Model \(EIMM\)](#)
- [Enterprise Interoperability Maturity Model for SMEs \(EIMM-SME\)](#)
- [EXP2PIM4SOA \(Express to PIM4SOA model transformation\)](#)
- [EXP2SCH \(Express to Schematron model transformation\)](#)
- [EXP2UML \(Express to UML model transformation\)](#)
- [EXP2XMI \(Express to XMI model transformation\)](#)
- [EXP2XSD \(Express to XSD model transformation\)](#)

- [Gabriel](#)
- [Gap Table Analyser \(GTA\)](#)
- [GRAI Methodology](#)
- [GraïTools](#)
  - [GraïTools MPCE interface](#)
- [Integrated Enterprise Modelling \(IEM\)](#)
- [JACK](#)
  - [JACK autonomous agents extension](#)
  - [JACK metamodel plugin for Eclipse](#)
- [Johnson](#)
- [Maestro](#)
  - [Maestro PIM4SOA interface](#)
- [Mediated Support Tool \(MST\)](#)
- [Methodology for modelling cross-organisational business processes \(CBPs\)](#)
- [Metis](#)
  - [Metis BPM MPCE interface](#)
  - [Metis BPM template](#)
  - [Metis DIF template](#)
  - [Metis EKA template](#)
  - [Metis ITM template](#)
  - [Metis Enterprise Task Management](#)
  - [Metis UML template](#)
  - [Metis Web service template](#)
- [Model-driven integration of JACK and Web services](#)
- [MOF repository](#)
- [MO2GO \(Method for Object Oriented Business Process Optimization\)](#)
  - [MO2GO IEM/BPDM interface](#)
  - [MO2GO IEM/Maestro interface](#)
  - [MO2GO IEM/MPCE interface](#)
  - [MO2GO IEM/PIM4SOA interface](#)
  - [MO2GO IEM/UML interface](#)
  - [MO2GO process assistant](#)
- [Monitoring Support Tool \(MST\)](#)
- [MPCE \(Modelling Platform for Collaborative Enterprises\)](#)
- [Nehemiah](#)
- [OPAL \(Object, Process, Actor modelling Language\)](#)
- [PIM4SOA](#)
- [PIM4SOA Execution and Simulation Platform](#)
- [PIM4SOA Metamodel Feature for Eclipse](#)
- [PIM4SOA to BPEL Transformation Feature for Eclipse](#)
- [PIM4SOA to JACK model transformation](#)
- [PIM4SOA to OWL-S model transformation](#)

- [PIM4SOA to Web services Model Transformations](#)
- [PIM4SOA to WSDL Transformation Feature for Eclipse](#)
- [PIM4SOA to XSD Transformation Feature for Eclipse](#)
- [POP\\* language interchange format and methodology](#)
- Protege
- [RSM \(Rational Software Modeler\)](#)
  - [RSM PIM4SOA Profiles Feature](#)
  - [RSM PIM4SOA\(UML\) to PIM4SOA\(EMF\) Transformation Feature](#)
  - [RSM POP\\* Profiles Feature](#)
  - [RSM POP\\*\(UML\) to EKA\(EMF\) Transformation Feature](#)
  - [RSM Web Services Profiles Feature](#)
- [Semantic support for service descriptions](#)
- [Semaphore - UML semantic mapping tool](#)
- [SOAP feedback analyser](#)
- STEP mapper
- Test driver
- [THEMIS](#)
- [WSDL Analyser](#)

### 7.1.3. Demonstrators

- [Overview](#)

## 7.2. Composite solutions

### 7.2.1. ATHENA Interoperability Framework (AIF)

#### 7.2.1.1. Datasheet

Solution data	
Name	ATHENA Interoperability Framework (AIF)
Result type	Framework
Description/functionality	<p>The framework aims at providing solution developers and integrators with guidelines on how to use the ATHENA solutions in addressing their business needs and technical requirements for interoperability. The framework is structured into three main parts:</p> <ul style="list-style-type: none"> <li>• Conceptual integration which focuses on concepts, metamodels, languages and model relationships. It provides us with a foundation for systemising various aspects of software model interoperability.</li> </ul>

	<ul style="list-style-type: none"> <li>• Technical integration which focuses on the software development and execution environments. It provides us with development tools for developing software models and execution platforms for executing software models.</li> <li>• Applicative integration which focuses on methodologies, standards and domain models. It provides us with guidelines, principles and patterns that can be used to solve software interoperability issues.</li> </ul>
Benefits to interoperability	A framework for relating solution approaches to problem areas of interoperability.
Supported models/methodologies	-
Supported input interfaces	-
Supported output interfaces	-
Validation/demonstration	Providing a reference framework for the various pilots.
Standards compliance	-
Availability	-
License	-
Status	Prototype
Requirements/dependencies	-
Web references	<ul style="list-style-type: none"> <li>• Website: <a href="http://athena.akmodeling.com/Team/Repository/Projects/Project_207">http://athena.akmodeling.com/Team/Repository/Projects/Project_207</a></li> </ul>
<b>Composed of the following solutions</b>	
Conceptual	List of conceptual solutions
Applicative	List of applicative solutions
Technical	List of technical solutions
<b>ATHENA metadata</b>	
Contact person	Arne-Jørgen Berre, SINTEF
Contributors	SINTEF, TXT, FHG IPK, SAP, TROUX, EADS, CRF, FORMULA, INTRACOM, ESI, SIEMENS, AIDIMA
Provided by project/activity	<ul style="list-style-type: none"> <li>• A4 – Interoperability Framework and Services for Networked Enterprises</li> </ul>

Deliverables representing result	D.A4.2 “Specification of Interoperability Framework and Profiles, Guidelines and Best Practices” (M36)
Contribution to key result	<ul style="list-style-type: none"> <li>6. Reference Architecture</li> </ul>
Used in pilot	Providing a reference framework for the various pilots
Deliverable providing evaluation	-

## 7.2.2. ATHENA Interoperability Project Support Suite

### 7.2.2.1. Datasheet

Solution data	
Name	ATHENA Interoperability Project Support Suite
Result type	Tool suite
Description/functionality	The ATHENA Interoperability Project Support Suite provides a set of life-cycle services.
Benefits to interoperability	-
Supported models/methodologies	-
Supported input interfaces	-
Supported output interfaces	-
Validation/demonstration	-
Standards compliance	-
Availability	-
License	-
Status	-
Requirements/dependencies	-
Web references	-
Composed of the following solutions	
Conceptual	
Applicative	
Technical	<ul style="list-style-type: none"> <li><a href="#">Business Issues Planner (BIP)</a></li> <li><a href="#">Capability Tables Loader (CTL)</a></li> </ul>

	<ul style="list-style-type: none"> <li>• <a href="#">Gap Table Analyser (GTA)</a></li> <li>• <a href="#">Mediated Support Tool (MST)</a></li> <li>• <a href="#">Monitoring Support Tool (MST)</a></li> </ul>
<b>ATHENA metadata</b>	
Contact person	Claudia Guglielmina, TXT
Contributors	TXT
Provided by project/activity	<ul style="list-style-type: none"> <li>• A4 – Interoperability Framework and Services for Networked Enterprises</li> </ul>
Deliverables representing result	-
Contribution to key result	<ul style="list-style-type: none"> <li>• 8. Interoperability Infrastructure</li> </ul>
Used in pilot	-
Deliverable providing evaluation	-

### 7.2.3. ATHENA Model-Driven Interoperability (MDI) Framework

#### 7.2.3.1. Datasheet

<b>Solution data</b>	
Name	ATHENA Model-Driven Interoperability (MDI) Framework
Result type	<ul style="list-style-type: none"> <li>• Framework</li> <li>• Methodology and guidelines</li> </ul>
Description/functionality	Model-driven development (MDD), and in particular OMG's Model Driven Architecture (MDA), is emerging as the state of practice for developing modern enterprise applications and software systems. The MDD paradigm provides us with a better way of addressing and solving interoperability issues compared to earlier non-modelling approaches.
Benefits to interoperability	<p>The <a href="#">ATHENA Model-Driven Interoperability (MDI) Framework</a> provides guidelines for how model-driven development (MDD) approaches can be applied in developing interoperable enterprise software systems.</p> <p>A reference guide that documents the purpose and concepts behind the model-driven interoperability results from ATHENA A5 and A6 in particular, and how to apply them. The reference guide does not</p>

	<p>cover the usage of the ATHENA A5 and A6 tools, but focuses on principles and methods.</p> <p>The MDI framework aims at providing guidelines for the following topics:</p> <ul style="list-style-type: none"> <li>• Model-driven architecture (MDA) and interoperability</li> <li>• Metamodelling</li> <li>• UML profiles and domain-specific languages (DSLs)</li> <li>• Model transformations</li> <li>• Method engineering</li> </ul>
Supported models/methodologies	-
Supported input interfaces	-
Supported output interfaces	-
Validation/demonstration	MDI is used internally within A6 to develop the PIM4SOA models and tools. Results of SOI has been tested in the AIDIMA e-procurement scenario.
Standards compliance	-
Availability	-
License	-
Status	-
Requirements/dependencies	-
Web references	<ul style="list-style-type: none"> <li>• Website: <a href="http://modelbased.net/mdi/">http://modelbased.net/mdi/</a></li> </ul>
<b>Composed of the following solutions</b>	
Conceptual	-
Applicative	-
Technical	-
<b>ATHENA metadata</b>	
Contact person	Brian Elvesæter, SINTEF
Contributors	ESI, SINTEF, IBM, DFKI
Provided by project/activity	<ul style="list-style-type: none"> <li>• A6 – Model-driven and Adaptive Interoperability Architectures</li> </ul>
Deliverables representing result	<ul style="list-style-type: none"> <li>• D.A5.2: Model and Specification of service</li> </ul>



	<p>description and usage as well as advanced concepts (M18)</p> <ul style="list-style-type: none"> <li>• D.A5.4: Execution Framework(s) for Planned and Customisable Service-Oriented Architectures (M21)</li> <li>• D.A6.4 “Model-Driven and Adaptable Interoperability Infrastructure” (M24)</li> <li>• Training material developed in B6, namely the courses AP1, AP2, AP3, AP4, AP5 and AP6.</li> </ul>
Contribution to key result	<ul style="list-style-type: none"> <li>• 7. Guidelines and Best Practices</li> <li>• 13. Model-driven and Adaptable Interoperability Framework and Infrastructure</li> </ul>
Used in pilot	-
Deliverable providing evaluation	<ul style="list-style-type: none"> <li>• D.A5.5 “Validation of Research Results” (M24)</li> <li>• D.A6.4 “Model-Driven and Adaptable Interoperability Infrastructure” (M24)</li> </ul>

## 7.2.4. ATHENA Modelling Tool Suite for Eclipse/RSM

### 7.2.4.1. Datasheet

Solution data	
Name	ATHENA Modelling Tool Suite for Eclipse/RSM
Result type	Tool suite
Description/functionality	<p>The <a href="#">PIM4SOA project</a> aims to develop open-source modelling tools and modelling services in the Eclipse environment to support the design of service-oriented architectures (SOAs) in a platform-independent or technology-neutral manner following the OMG MDA approach. The tools and services are released under the <a href="#">Eclipse Public License (EPL)</a>.</p> <p>PIM4SOA is closely aligned and has been based on the Business Process Definition Metamodel that is in the process of standardisation by OMG. However as the standardization did not completed in the timeframe of ATHENA, the PIM4SOA metamodel was developed as a simplified version. The PIM4SOA metamodel covers four important aspects: service, process, information and quality of service.</p> <ul style="list-style-type: none"> <li>• Information: in the context of virtual enterprises</li> </ul>

	<p>information represents one of the most important elements that need to be described. In fact the other aspects manage or are based on information elements.</p> <ul style="list-style-type: none"> <li>• Service: our main intention is to be able to describe SOA independently from the technology used. Service represents business accessible functionality.</li> <li>• Process: processes describe a set of interactions amongst services in terms of messages exchange.</li> <li>• Quality of service (QoS): a suitable feature is the description and the modelling of non-functional aspects related with the services described.</li> </ul>
Benefits to interoperability	<p>One important result is the PIM4SOA metamodel which defines an abstract language to specify executable business processes that execute within an enterprise and may collaborate between otherwise independent business processes executing in different business units or enterprises. The main objective of the specification is:</p> <ul style="list-style-type: none"> <li>• The ability to exchange business process specifications between modelling tools, and between tools and execution environments.</li> </ul>
Supported models/methodologies	-
Supported input interfaces	-
Supported output interfaces	-
Validation/demonstration	-
Standards compliance	-
Availability	-
License	Eclipse Public License
Status	Prototype
Requirements/dependencies	-
Web references	-
<b>Composed of the following solutions</b>	
Conceptual	<ul style="list-style-type: none"> <li>• <a href="#">PIM4SOA</a></li> </ul>
Applicative	-
Technical	<ul style="list-style-type: none"> <li>• <a href="#">BPEL Metamodel Feature for Eclipse</a></li> <li>• <a href="#">EKA Metamodel Feature for Eclipse</a></li> </ul>

	<ul style="list-style-type: none"> <li>• <a href="#">EKA to PIM4SOA Transformation Feature for Eclipse</a></li> <li>• <a href="#">PIM4SOA Metamodel Feature for Eclipse</a></li> <li>• <a href="#">PIM4SOA to BPEL Transformation Feature for Eclipse</a></li> <li>• <a href="#">PIM4SOA to JACK model transformation</a></li> <li>• <a href="#">PIM4SOA to WSDL Transformation Feature for Eclipse</a></li> <li>• <a href="#">PIM4SOA to XSD Transformation Feature for Eclipse</a></li> <li>• <a href="#">RSM PIM4SOA Profiles Feature</a></li> <li>• <a href="#">RSM PIM4SOA(UML) to PIM4SOA(EMF) Transformation Feature</a></li> <li>• <a href="#">RSM POP* Profiles Feature</a></li> <li>• <a href="#">RSM POP*(UML) to EKA(EMF) Transformation Feature</a></li> <li>• <a href="#">RSM Web Services Profiles Feature</a></li> </ul>
<b>ATHENA metadata</b>	
Contact person	<ul style="list-style-type: none"> <li>• Gorka Benguria, ESI</li> <li>• Brian Elvesæter, SINTEF</li> </ul>
Contributors	ESI, SINTEF, DFKI, IBM
Provided by project/activity	<ul style="list-style-type: none"> <li>• A1 – Enterprise Modelling in the Context of Collaborative Enterprises</li> <li>• A5 – Planned and Customisable Service-Oriented Architectures</li> <li>• A6 – Model-driven and Adaptive Interoperability Architectures</li> </ul>
Deliverables representing result	-
Contribution to key result	<ul style="list-style-type: none"> <li>• 9. Collaborative Enterprise Modelling Platform</li> <li>• 12. Service Composition Framework</li> <li>• 13. Model-driven and Adaptable Interoperability Framework and Infrastructure</li> </ul>
Used in pilot	-
Deliverable providing evaluation	-

## 7.2.5. Semantic reconciliation framework

### 7.2.5.1. Datasheet

<b>Solution data</b>	
Name	ATHENA Collaboration Space

Result type	
Description/functionality	The A3 subproject aims to provide a basic set of semantic tools and services which can be used by other components in order to include semantic support for solving interoperability issues. Some of the solutions, such as the ontology management system, are more general purpose and can be used as basis for achieving different solutions, while other tools are oriented to solve specific semantic goals. In particular the semantic reconciliation allows to reconcile documents and messages, starting from a common ontology and semantic annotations, without the human intervention at run-time.
Benefits to interoperability	-
Supported models/methodologies	-
Supported input interfaces	-
Supported output interfaces	-
Validation/demonstration	-
Standards compliance	-
Availability	-
License	-
Status	-
Requirements/dependencies	-
Web references	-
<b>Composed of the following solutions</b>	
Conceptual	<ul style="list-style-type: none"> <li>• <a href="#">OPAL (Object, Process, Actor modelling Language)</a></li> </ul>
Applicative	<ul style="list-style-type: none"> <li>• <a href="#">OPAL (Object, Process, Actor modelling Language)</a></li> </ul>
Technical	<ul style="list-style-type: none"> <li>• <a href="#">A*</a></li> <li>• <a href="#">ARES</a></li> <li>• <a href="#">ARGOS</a></li> <li>• <a href="#">ATHOS</a></li> <li>• <a href="#">THEMIS</a></li> </ul>
<b>ATHENA metadata</b>	

Contact person	-
Contributors	-
Provided by project/activity	<ul style="list-style-type: none"> <li>A3 – Knowledge Support and Semantic Mediation Solutions</li> </ul>
Deliverables representing result	-
Contribution to key result	-
Used in pilot	-
Deliverable providing evaluation	-

## 7.2.6. ATHENA Service-Oriented Interoperability (SOI) Framework

### 7.2.6.1. Datasheet

Solution data	
Name	ATHENA Service-Oriented Interoperability (SOI) Framework
Result type	<ul style="list-style-type: none"> <li>Framework</li> <li>Methodology and guidelines</li> </ul>
Description/functionality	<p>According to W3C, a service-oriented architecture (SOA) specifies a set of components whose interfaces can be described, published, discovered and invoked over a network. SOA aims to promote software development in a way that leverages the construction of dynamic systems which can easily adapt to volatile environments and be easily maintained as well. The decoupling of system constituent parts enables the re-configuration of system components according to the end-user's needs and the system's environment. Furthermore, the use of widely accepted standards and protocols that are based on XML and operate above internet standards (HTTP, SMTP, etc) enhances interoperability.</p> <p>This result consists of models, metamodels, profiles and methodology for modelling Web Services and autonomous agents. The model-driven development (MDD) framework for Web services described in D.A5.2 provides modelling guidelines for how to specify interoperable Web services in service-oriented architectures (SOAs). The</p>

	<p>framework consists of three main parts:</p> <ul style="list-style-type: none"> <li>• Web service models and metamodels, which describes how Web services should be specified in a service-oriented architecture. The framework covers models for how to specify service descriptions, service compositions, service addressing, service registry, semantic annotation of services, quality of service and e-contracts.</li> <li>• UML profile for Web services, which defines domain-specific language extensions for UML to support the specification of Web service models in UML.</li> <li>• Baseline methodology for SOA, which provides guidelines for developing platform-independent models for SOA and Web service models, and how to map between these.</li> </ul>
Benefits to interoperability	<p>The <a href="#">ATHENA Service-Oriented Interoperability (SOI) Framework</a> provides guidelines for developing and integrating software services in service-oriented architectures (SOAs) using Web services and agent technologies.</p> <p>This result looks into the relationships between Web services and agents. It covers amongst other things the BDI (belief, desire and intention) metamodel for JACK agents and promotes the use of agents to build more dynamic and adaptive service-oriented systems.</p>
Supported models/methodologies	-
Supported input interfaces	-
Supported output interfaces	-
Validation/demonstration	-
Standards compliance	-
Availability	-
License	-
Status	-
Requirements/dependencies	-
Web references	<ul style="list-style-type: none"> <li>• Website: <a href="http://modelbased.net/soi/">http://modelbased.net/soi/</a></li> </ul>
<b>Composed of the following solutions</b>	
Conceptual	<ul style="list-style-type: none"> <li>• <a href="#">PIM4SOA</a></li> </ul>

Applicative	<ul style="list-style-type: none"> <li>• <a href="#">Baseline methodology for SOA</a></li> </ul>
Technical	<ul style="list-style-type: none"> <li>• <a href="#">ATHENA Modelling Tool Suite for Eclipse/RSM</a></li> <li>• <a href="#">Johnson</a></li> <li>• <a href="#">WSDL Analyser</a></li> </ul>
<b>ATHENA metadata</b>	
Contact person	Brian Elvesæter, SINTEF
Contributors	DFKI, ESI, SAP, SINTEF
Provided by project/activity	<ul style="list-style-type: none"> <li>• A5 – Planned and Customisable Service-Oriented Architectures</li> <li>• A6 – Model-driven and Adaptive Interoperability Architectures</li> </ul>
Deliverables representing result	<ul style="list-style-type: none"> <li>• D.A5.2: Model and Specification of service description and usage as well as advanced concepts (M18)</li> <li>• D.A5.4: Execution Framework(s) for Planned and Customisable Service-Oriented Architectures (M21)</li> </ul>
Contribution to key result	<ul style="list-style-type: none"> <li>• 7. Guidelines and Best Practices</li> <li>• 12. Service Composition Framework</li> <li>• 13. Model-driven and Adaptable Interoperability Framework and Infrastructure</li> </ul>
Used in pilot	<ul style="list-style-type: none"> <li>• AIDIMA: eProcurement Pilot</li> <li>• EADS: Change management process</li> <li>• CRF: Automotive Pilot</li> </ul>
Deliverable providing evaluation	D.A5.5 “Validation of Research Results” (M24)

## 7.3. Individual solutions

### 7.3.1. A\*

#### 7.3.1.1. Datasheet

<b>Solution data</b>	
Name	A*
Result type	Semantic annotation tool
Description/functionality	<ul style="list-style-type: none"> <li>• This result consists of a methodology for</li> </ul>

	<p>semantic annotation of digital resources and a tool, A*, that implements a semantic annotation environment based on such a methodology.</p> <ul style="list-style-type: none"> <li>• The semantic annotation method has been conceived for a large variety of digital resources (e.g., business documents, business processes, web services) involved in a typical cooperation between enterprises in an e-Business scenario. The proposed method considers the content and the structure of the documents to be exchanged and contrasts them with the ontology, with the goal of building a mapping to the Reference Ontology. Semantic annotation is a complex task; to make it easier the proposed method is based on an incremental approach that, starting from simple keyword based annotations, evolves towards more advanced and rigorous ones, reaching the last level where the annotation is represented by OWL expressions.</li> <li>• The A* tool supports the user to progressively identify possible mappings between the digital resource and the Reference Ontology specifically selected. To this end, A* is characterised by a friendly user interface and an automatic support capable of proposing matching between elements of the business resource and Reference Ontology elements starting from the analysis of the business resource.</li> </ul>
Benefits to interoperability	This result provides a low entry barrier of semantic annotation by using an incremental approach.
Supported models/methodologies	Internally A* works with RDFS and OWL files. Services provided externally are described through WSDL files. Output is a multi-level annotation: L1,L2,L3 are proprietary representations; L4 is OWL.
Supported input interfaces	<ul style="list-style-type: none"> <li>• Access to Athos for importing the ontology</li> <li>• Access to the RDF visualisation service to import the RDF graphical presentation of the document to be annotated</li> <li>• Import from A6 repository to be still agreed</li> </ul>
Supported output interfaces	<ul style="list-style-type: none"> <li>• A* will provide access to the Annotation Repository via web service</li> </ul>
Validation/demonstration	<ul style="list-style-type: none"> <li>• AIAG: eKanban Pilot</li> <li>• AIDIMA: eProcurement Pilot</li> <li>• EADS: Change management process</li> <li>• CRF: Automotive Pilot</li> </ul>



Standards compliance	-
Availability	-
License	-
Status	-
Requirements/dependencies	-
Web references	-
<b>ATHENA metadata</b>	
Contact person	Francesco Taglino, LEKS
Contributors	LEKS
Provided by project/activity	<ul style="list-style-type: none"> <li>A3 – Knowledge Support and Semantic Mediation Solutions</li> </ul>
Deliverables representing result	D.A3.3: Semantic Annotation language and tool for information and Business Processes (M20)
Contribution to key result	<ul style="list-style-type: none"> <li>11. Ontology-based Semantic Annotation and Reconciliation method/language/tool</li> </ul>
Used in pilot	<ul style="list-style-type: none"> <li>AIAG: eKanban Pilot</li> <li>AIDIMA: eProcurement Pilot</li> <li>EADS: Change management process</li> <li>CRF: Automotive Pilot</li> </ul>
Deliverable providing evaluation	-

### 7.3.2. Active Knowledge Model (AKM) Execution Platform

#### 7.3.2.1. Datasheet

<b>Solution data</b>	
Name	Active Knowledge Model (AKM) Execution Platform
Result type	Modelling and execution platform
Description/functionality	<p>An active model platform has these primary characteristics:</p> <ul style="list-style-type: none"> <li>Integrates modelling and execution, concurrently at runtime, allowing business users to compose and customise simple interoperable solutions without programming,</li> <li>Captures business logic (enterprise models) in</li> </ul>

	<p>executable models, rather than hard-coding it into software, creating an open infrastructure more easy to integrate and extend.</p> <ul style="list-style-type: none"> <li>• Integrates modelling at different meta-levels, concurrently at runtime, replacing both data and the software the manipulates it with reflective models.</li> <li>• Supports knowledge-intensive ad-hoc, emergent and dynamic cross-organisational processes, not just automated routine procedures,</li> <li>• Provides a simple and easily configurable collaboration infrastructure for SMEs and other companies not willing or able to invest in large scale automation,</li> <li>• Enables prototyping and piloting of cross-organisational solutions, letting companies interact and establish mutual trust in the early phases of a collaboration, without a large up-front investment,</li> <li>• Facilitates cross-organisational training, experimentation and testing during the early phases of solution development,</li> <li>• Enables exceptions of predefined procedures to be captured and managed inside the system, for traceability, accountability, and coordination</li> </ul> <p>The Active Model Execution Platform provides the following functions</p> <ul style="list-style-type: none"> <li>• A template and modelling language for configuring portal structures and services in visual models, in the Metis tool</li> <li>• A set of composable and configurable web user interface elements to built solutions from, e.g. lists, trees, tabfolders, navigation and edit controls,</li> <li>• A multi-layered and multi-dimensional configuration framework, where features can be inherited along any relevant relationship, as well as through users' modelled roles,</li> <li>• A modelling language and import facility for web services, as well as a configurable web service invocation component on the AKM server.</li> <li>• A parameterised query interface for model-configuring the navigation structures on top of large models</li> <li>• A generic, interactive framework for extracting parameters for services from the current user, task and collaboration context.</li> </ul>
Benefits to interoperability	What are the benefits of applying the result in terms of improving interoperability?

Supported models/methodologies	-
Supported input interfaces	-
Supported output interfaces	-
Validation/demonstration	<p>The capabilities of the Active Model Execution Platform for setting up generic, model-configured application services will be validated through its use for defining Troux task management solution in A2. The platform's qualities for providing more customised business and user solutions will be prototyped and validated alongside the A1 results in</p> <ul style="list-style-type: none"> <li>• defining and managing the overall collaboration space, as well as targetted collaborative workplaces for specific projects, in the EADS pilot,</li> <li>• establishing, utilising and adapting individual workplaces for the product manager role in the product portfolio management pilot at Intracom.</li> </ul> <p>The system will be demonstrated for the EADS pilot during the M24 review, and then installed on a server hosted by Troux.</p>
Standards compliance	<p>The AKM platform interoperates with external enterprise modeling tools through the POP* (Product, Organization, Process etc.) language and the EKA (Enterprise Knowledge Architecture) XML format defined by A1. This language has been taken into the standardisation process at ISO, and it is also being proposed for the OMG. The platform also offers services for importing data into models through standard formats such as SQL, XML, and tools such as Microsoft Excel and Project. The XML import has been used for capturing WSDL definitions as web service models.</p>
Availability	Hosted service/solution
License	-
Status	Prototype
Requirements/dependencies	<ul style="list-style-type: none"> <li>• <a href="#">MetisMetis</a></li> </ul> <p>The AKM platform is built as a customisable extension to the Metis Enterprise product line. It utilises the Metis client tools for visual modeling, the Troux Information Portal (TIP) for model-configured web workplaces and portals, the Metis Enterprise Repository for storing models, and the Metis Team</p>

	repository for storing document and model view files. The Metis Team repository also implements the MPCE services defined by A1.
Web references	-
<b>ATHENA metadata</b>	
Contact person	Håvard Jørgensen, AKM
Contributors	AKM
Provided by project/activity	<ul style="list-style-type: none"> <li>• A1 – Enterprise Modelling in the Context of Collaborative Enterprises</li> <li>• A6 – Model-driven and Adaptive Interoperability Architectures</li> </ul>
Deliverables representing result	-
Contribution to key result	<ul style="list-style-type: none"> <li>• 9. Collaborative Enterprise Modelling Platform</li> </ul>
Used in pilot	-
Deliverable providing evaluation	-

### 7.3.3. Active object flow (AOF) methodology

#### 7.3.3.1. Datasheet

<b>Solution data</b>	
Name	Active object flow (AOF) methodology
Result type	Methodology/guideline
Description/functionality	The Active Object Flow methodology provides a framework and an algorithm for transforming workflows specified as UML Activity Diagrams into a set of specifications for active objects running on a Linda-like coordination middleware. As a result, a centralized description of the workflow is executed on a decentralised architecture, thereby providing increases fonctionnalités, especially in mobile environments.
Benefits to interoperability	Show that it is possible to go from UML Activity Diagrams to coordination middlewares.
Supported models/methodologies	-
Supported input interfaces	-

Supported output interfaces	-
Validation/demonstration	-
Standards compliance	-
Availability	-
License	-
Status	Concept
Requirements/dependencies	-
Web references	-
<b>ATHENA metadata</b>	
Contact person	Julien Vayssiere, SAP
Contributors	SAP
Provided by project/activity	-
Deliverables representing result	-
Contribution to key result	-
Used in pilot	-
Deliverable providing evaluation	-

### 7.3.4. Agent-based Computing Architecture and Design/Runtime Platform

#### 7.3.4.1. Datasheet

<b>Solution data</b>	
Name	Agent-based Computing Architecture and Design/Runtime Platform
Result type	Platform or platform component
Description/functionality	<ul style="list-style-type: none"> <li>The agent based methodology allows the introduction of self-organisation, autonomy, flexibility, and robustness in service-oriented architectures. Interoperability is supported by explaining how general service-oriented architectures could be extended with these properties.</li> <li>The Jack Intelligent Agent™ Development Environment (JDE) was adopted as a designing and runtime tool for the design and execution of Belief Desire Intention (BDI) agents and teams</li> </ul>

	of such agents. In the context of ATHENA JDE was extended with a library for the integration with service-oriented architectures (Web Services). Additionally model to text transformations were investigated to integrate JDE with the Eclipse modelling framework and with Rational Software Modeller.
Benefits to interoperability	The methodology allows to introduce self-organisation, autonomy, flexibility, and robustness in service-oriented architectures. Interoperability is supported by explaining how general service-oriented architectures could be extended with these properties.
Supported models/methodologies	-
Supported input interfaces	-
Supported output interfaces	-
Validation/demonstration	The design and execution platform is used in the design of the agent-based components of the demonstrators developed in ATHENA.
Standards compliance	-
Availability	Installed service/solution
License	-
Status	Stable production version
Requirements/dependencies	<ul style="list-style-type: none"> <li>• <a href="#">JACKJACK</a></li> </ul>
Web references	-
<b>ATHENA metadata</b>	
Contact person	Klaus Fischer, DFKI
Contributors	DFKI
Provided by project/activity	<ul style="list-style-type: none"> <li>• A5 – Planned and Customisable Service-Oriented Architectures</li> <li>• A6 – Model-driven and Adaptive Interoperability Architectures</li> </ul>
Deliverables representing result	-
Contribution to key result	<ul style="list-style-type: none"> <li>• 13. Model-driven and Adaptable Interoperability Framework and Infrastructure</li> </ul>
Used in pilot	-

Deliverable providing evaluation	-
----------------------------------	---

### 7.3.5. Autonomous Computing Engineering Reference Guide

#### 7.3.5.1. Datasheet

Solution data	
Name	Autonomous Computing Engineering Reference Guide
Result type	Methodology/guideline
Description/functionality	<ul style="list-style-type: none"> <li>A methodology for the design of autonomous systems based on agent technologies is described. The methodology introduces the concept of holons that allow the description of self-organisation in autonomous architectures. This concepts match nicely with the concepts of team-oriented programming in BDI agents, which is adopted in A6 for agent-based model-driven service-composition and choreography. However, the concept of holons is more general.</li> <li>The methodology introduces self-organisation, autonomy, flexibility, and robustness in service-oriented architectures. Interoperability is supported by explaining how general service-oriented architectures could be extended with these properties</li> </ul>
Benefits to interoperability	The methodology allows to introduce self-organisation, autonomy, flexibility, and robustness in service-oriented architectures. Interoperability is supported by explaining how general service-oriented architectures could be extended with these properties.
Supported models/methodologies	-
Supported input interfaces	-
Supported output interfaces	-
Validation/demonstration	Is used as a methodology in the design of the agent-based components of the demonstrators developed in ATHENA.
Standards compliance	-
Availability	-

License	-
Status	Concept
Requirements/dependencies	-
Web references	-
<b>ATHENA metadata</b>	
Contact person	Klaus Fischer, DFKI
Contributors	DFKI
Provided by project/activity	<ul style="list-style-type: none"> <li>• A5 – Planned and Customisable Service-Oriented Architectures</li> <li>• A6 – Model-driven and Adaptive Interoperability Architectures</li> </ul>
Deliverables representing result	-
Contribution to key result	<ul style="list-style-type: none"> <li>• 13. Model-driven and Adaptable Interoperability Framework and Infrastructure</li> </ul>
Used in pilot	-
Deliverable providing evaluation	-

### 7.3.6. ARES

#### 7.3.6.1. Datasheet

<b>Solution data</b>	
Name	ARES
Result type	Semantic reconciliation engine
Description/functionality	<p>ARES is the run-time engine for the execution of the semantic rules. To realize semantic reconciliation at run-time the real message exchange among different partners includes the semantic reconciliation process using the ATHENA Reconciliation Engine (ARES).</p> <p>This tool is integrated in the ATHENA architecture and can also easily be included in external frameworks because is completely built on top of a service-based architecture. ARES uses the semantic rules generated by ARGOS in order to apply reconciliation on messages that carry RDF payloads based on RDF models stored in THEMIS. ARES intercepts a message and extracts the payload. After</p>



	that, ARES retrieves the reconciliation rules associated to that particular model using the service provided by ARGOS. Essentially, in order to perform the reconciliation of a document from a schema A into a Schema B, the reconciliation process will be the result of the composition of the forward rules for Schema A and backward rules for B. In the context of ATHENA the message is provided to the ARES tool by the ATHENA Service Execution Framework (i.e. Johnson).
Benefits to interoperability	The solution provides run-time transformation of messages.
Supported models/methodologies	-
Supported input interfaces	<ul style="list-style-type: none"> <li>• Access to Themis for importing the models of the document to be reconciled</li> <li>• The tool will be used by Johnson as semantic mediator</li> <li>• Import from A6 repository to be still agreed</li> </ul>
Supported output interfaces	<ul style="list-style-type: none"> <li>• ARES will provide a basic set of services to finalize the reconciliation of messages and document</li> <li>• ARES will provide also a set of services for managing the logs information about the real execution of the engine</li> </ul>
Validation/demonstration	<ul style="list-style-type: none"> <li>• AIAG: eKanban Pilot</li> <li>• AIDIMA: eProcurement Pilot</li> <li>• EADS: Change management process</li> <li>• CRF: Automotive Pilot</li> </ul>
Standards compliance	-
Availability	-
License	-
Status	-
Requirements/dependencies	-
Web references	-
<b>ATHENA metadata</b>	
Contact person	Lorenzo Pondrelli, FORMULA
Contributors	-
Provided by project/activity	<ul style="list-style-type: none"> <li>• A3 – Knowledge Support and Semantic</li> </ul>

	Mediation Solutions
Deliverables representing result	<ul style="list-style-type: none"> <li>D.A3.5: A reconciliation and mediation engine, capable to efficiently process semantic mediation and reconciliation rules (M24)</li> </ul>
Contribution to key result	<ul style="list-style-type: none"> <li>11. Ontology-based Semantic Annotation and Reconciliation method/language/tool</li> </ul>
Used in pilot	<ul style="list-style-type: none"> <li>AIAG: eKanban Pilot</li> <li>AIDIMA: eProcurement Pilot</li> <li>EADS: Change management process</li> <li>CRF: Automotive Pilot</li> </ul>
Deliverable providing evaluation	-

### 7.3.7. ARIS (Architecture of Integrated Information Systems)

#### 7.3.7.1. Datasheet

Solution data	
Name	ARIS (Architecture of Integrated Information Systems)
Result type	Modelling tool
Description/functionality	ARIS is an enterprise modelling tool. The Architecture of Integrated Information Systems (ARIS) supports the modelling of Event-driven Process Chains (EPC).
Benefits to interoperability	-
Supported models/methodologies	<ul style="list-style-type: none"> <li>Event-driven Process Chains</li> </ul>
Supported input interfaces	-
Supported output interfaces	-
Validation/demonstration	-
Standards compliance	-
Availability	-
License	Commercial
Status	Commercial

Requirements and dependencies	-
Web references	<ul style="list-style-type: none"> <li>Website: <a href="http://www.ids-scheer.com/international/english/products/53961">http://www.ids-scheer.com/international/english/products/53961</a></li> </ul>
<b>ATHENA metadata</b>	
Contact person	n/a
Contributors	n/a
Provided by project/activity	n/a
Deliverables representing result	n/a
Contribution to key result	n/a
Used in pilot	n/a
Deliverable providing evaluation	n/a

### 7.3.8. ARGOS

#### 7.3.8.1. Datasheet

<b>Solution data</b>	
Name	ARGOS
Result type	Reconciliation rules generation tool
Description/functionality	<p>This result is used to define, create, store and manage the transformation rules used to reconcile heterogeneous documents. ARGOS supports rules that are executable by a machine; semantic driven rules where rule creation is driven by the Reference Ontology and by the semantic annotations associated to the resources to be reconciled; and bi-directional rules that define both forward (to be applied to transform a document instance into an ontology instance) and backward rules (to be applied to transform a ontology instance into an document instance).</p> <p>While annotations represent conceptual correspondences between the business resource and the Reference Ontology, transformation rules represent a procedural way for transforming ground resources (i.e., data) into ontology instances (forward transf. from the resource schema to the ontology) and viceversa (backward transf. from the ontology to the resource schema).</p>

	<p>ARGOS helps the users to create the correct semantic rules, applicable at runtime for the semantic reconciliation of message instances using the ARES result. Those semantic rules are built and stored using ARGOS. The sets of rules are related to the models stored into THEMIS.</p>
Benefits to interoperability	<ul style="list-style-type: none"> <li>• The result is aimed at supporting the typical user (a domain expert, working for an organization, that deeply knows the processes and documents that has to be made interoperable but not necessarily an IT expert) in defining rules that, executed at run-time by the ARES tool, transform document instances into/from a standard, commonly agreed format.</li> <li>• This goal has been reached by developing a graphical high-level environment in which formal models and ontology are represented as graphs and rules are generated by filling in pre-defined templates.</li> <li>• Web-based accessibility: Ares is accessible by a browser, from everywhere within or outside the enterprise.</li> <li>• The executable language for the rules supports the formalisms adopted for the document models and provides reasoning capabilities.</li> <li>• The solution is integrated into the A3 tool chain.</li> </ul>
Supported models/methodologies	<p>Internally ARGOS works with RDFS and OWL files. Output (generated rules) is compliant with the JENA syntax. Services provided externally are described through WSDL files</p>
Supported input interfaces	<ul style="list-style-type: none"> <li>• Access to ATHOS for importing the ontology graphical presentation</li> <li>• Access to the RDF visualisation service to import the RDF graphical presentation</li> <li>• Access to A* to retrieve annotation relevant for an RDF document</li> </ul>
Supported output interfaces	<ul style="list-style-type: none"> <li>• ARGOS provides access to the Reconciliation Rules repository via a web service</li> </ul>
Validation/demonstration	<ul style="list-style-type: none"> <li>• AIAG: eKanban Pilot</li> <li>• AIDIMA: eProcurement Pilot</li> <li>• EADS: Change management process</li> <li>• CRF: Automotive Pilot</li> </ul>
Standards compliance	-
Availability	-

License	-
Status	-
Requirements/dependencies	-
Web references	-
<b>ATHENA metadata</b>	
Contact person	Eva Coscia, TXT
Contributors	-
Provided by project/activity	<ul style="list-style-type: none"> <li>A3 – Knowledge Support and Semantic Mediation Solutions</li> </ul>
Deliverables representing result	D.A3.4 System for reconciliation rules specification, storage and management (M22)
Contribution to key result	<ul style="list-style-type: none"> <li>11. Ontology-based Semantic Annotation and Reconciliation method/language/tool</li> </ul>
Used in pilot	<ul style="list-style-type: none"> <li>AIAG: eKanban Pilot</li> <li>AIDIMA: eProcurement Pilot</li> <li>EADS: Change management process</li> <li>CRF: Automotive Pilot</li> </ul>
Deliverable providing evaluation	-

### 7.3.9. ASSERT

#### 7.3.9.1. Datasheet

<b>Solution data</b>	
Name	ASSERT
Result type	-
Description/functionality	-
Benefits to interoperability	-
Supported models/methodologies	-
Supported input interfaces	-
Supported output interfaces	-
Validation/demonstration	-
Standards compliance	-

Availability	-
License	-
Status	-
Requirements/dependencies	-
Web references	-
<b>ATHENA metadata</b>	
Contact person	-
Contributors	-
Provided by project/activity	-
Deliverables representing result	-
Contribution to key result	-
Used in pilot	-
Deliverable providing evaluation	-

### 7.3.10. ATHENA Event and Document Correlation (AEDoC)

#### 7.3.10.1. Datasheet

<b>Solution data</b>	
Name	ATHENA Event and Document Correlation (AEDoC)
Result type	Tool
Description/functionality	Each process and resource involved in the execution of a Cross-Organisational Business Process (CBP) has to be exclusively identified. This identification is used to link together CBPs, process instances and message payloads. In order to sustain this duty, the ATHENA Event and Document Correlation (AEDoC) provides a set of basic services for matching documents to process instances. The execution of a CBP requires that private documents and events are continuously linked to the correct process instances that can run all the different process engines in the whole architecture of the collaborating parties.
Benefits to interoperability	Correlation of documents and events to process instances independently of the underlying

	infrastructure.
Supported models/methodologies	-
Supported input interfaces	-
Supported output interfaces	-
Validation/demonstration	-
Standards compliance	-
Availability	-
License	-
Status	-
Requirements/dependencies	-
Web references	-
<b>ATHENA metadata</b>	
Contact person	-
Contributors	-
Provided by project/activity	<ul style="list-style-type: none"> <li>A2: Cross-Organisational Business Processes</li> </ul>
Deliverables representing result	<ul style="list-style-type: none"> <li>D.A2.4: Architecture for Enactment and Integration of Cross-Organizational Business Processes (M21)</li> </ul>
Contribution to key result	<ul style="list-style-type: none"> <li>10: Cross-Organisational Business Process Modelling and Enactment</li> </ul>
Used in pilot	-
Deliverable providing evaluation	<ul style="list-style-type: none"> <li>D.A2.5 “Validation of Research Results” (M24)</li> </ul>

### 7.3.11. ATHOS

#### 7.3.11.1. Datasheet

<b>Solution data</b>	
Name	ATHOS
Result type	Ontology authoring and management system
Description/functionality	This result consists of an Ontology Authoring and Management System for Informational

	<p>knowledge and Business Processes represented by the ATHOS tool and the associated OPAL (the Object, Process, Actor modelling Language) methodology. ATHOS allows the definition of both the domain concepts of an ontology and their relations.</p> <p>OPAL has the goals to provide an ontology building method capable of supporting and guiding the ontology modeller; and to provide a number of inherent constraints, associated to the above mentioned categories, used to guarantee a better quality for the built ontology.</p> <p>ATHOS now includes improved of query capabilities, able to answer to queries that involve the relations between concepts; Ontology Diagramming, able to show the ontology in a graphical way; and Import/Export facilities, for sharing ontologies with other tools and better integrating with the entire A3 semantic suite.</p>
Benefits to interoperability	ATHOS supports the OPAL methodology guiding the ontology developer resulting in improved ontology quality. Improved capabilities of ATHOS improve its usability for the pilots.
Supported models/methodologies	An OPAL ontology stored in the Athos DB. To be exported into proprietary XML, XMI-light, OWL
Supported input interfaces	-
Supported output interfaces	Export of ontologies via web service to be mainly used by A* and Argos
Validation/demonstration	<ul style="list-style-type: none"> <li>• AIAG: eKanban Pilot</li> <li>• AIDIMA: eProcurement Pilot</li> <li>• EADS: Change management process</li> <li>• CRF: Automotive Pilot</li> </ul>
Standards compliance	-
Availability	-
License	-
Status	-
Requirements/dependencies	-
Web references	-
<b>ATHENA metadata</b>	



Contact person	Francesco Taglino, LEKS
Contributors	LEKS
Provided by project/activity	<ul style="list-style-type: none"> <li>A3 – Knowledge Support and Semantic Mediation Solutions</li> </ul>
Deliverables representing result	D.A3.2: Ontology Authoring and Management System for informational knowledge and Business Processes (M20)
Contribution to key result	<ul style="list-style-type: none"> <li>11. Ontology-based Semantic Annotation and Reconciliation method/language/tool</li> </ul>
Used in pilot	<ul style="list-style-type: none"> <li>AIAG: eKanban Pilot</li> <li>AIDIMA: eProcurement Pilot</li> <li>EADS: Change management process</li> <li>CRF: Automotive Pilot</li> </ul>
Deliverable providing evaluation	-

### 7.3.12. BPEL Metamodel Feature for Eclipse

#### 7.3.12.1. Datasheet

Solution data	
Name	BPEL Metamodel Feature for Eclipse
Result type	Modelling tool.
Description/functionality	Provides EMF based model for BPEL, including serialization to BPEL text. Model is based on WS-BPEL 1.1 schema (XSD). Presented as Eclipse plugin.
Benefits to interoperability	Avoids re-implementing BPEL serialization - model can be created from XSD using the EMF tooling. Allows for possibility of manipulation at model level.
Supported models/methodologies	BPEL
Supported input interfaces	-
Supported output interfaces	-
Validation/demonstration	Used as integral part of PIM4SOA to BPEL transformation work
Standards compliance	<ul style="list-style-type: none"> <li><a href="http://www-128.ibm.com/developerworks/webservices/library/specif">http://www-128.ibm.com/developerworks/webservices/library/specif</a></li> <li><a href="http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=ws">http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=ws</a></li> </ul>

Availability	-
License	-
Status	Concept
Requirements/dependencies	EMF framework for Eclipse (also included in RSM or RSA)
Web references	-
<b>ATHENA metadata</b>	
Contact person	Anthony Beardsmore, IBM
Contributors	IBM
Provided by project/activity	<ul style="list-style-type: none"> <li>• A2 – Cross-Organisational Business Processes</li> <li>• A5 – Planned and Customisable Service-Oriented Architectures</li> <li>• A6 – Model-driven and Adaptive Interoperability Architectures</li> </ul>
Deliverables representing result	-
Contribution to key result	<ul style="list-style-type: none"> <li>• 12. Service Composition Framework</li> <li>• 13. Model-driven and Adaptable Interoperability Framework and Infrastructure</li> </ul>
Used in pilot	-
Deliverable providing evaluation	-

### 7.3.13. Business Issues Planner (BIP)

#### 7.3.13.1. Datasheet

<b>Solution data</b>	
Name	Business Issues Planner (BIP)
Result type	Tool
Description/functionality	<p>The Business Issues Planner (BIP) is a web (middleware) solution that supports the enterprises to collaborate into an interoperability project.</p> <p>BIP structures the identified set of interoperability gaps collected in the Gap Table in a project oriented mode to be planned and characterized as work items with deadlines and priorities. The list of work items resulting from the comparison phase could be</p>

	complemented by the work items that the mediator of the interoperability project introduces as necessary.
Benefits to interoperability	<ul style="list-style-type: none"> <li>• Implementation of a SOA platform for BI management.</li> <li>• Give access to Mediated Collaboration Tool.</li> </ul>
Supported models/methodologies	-
Supported input interfaces	-
Supported output interfaces	-
Validation/demonstration	-
Standards compliance	Use of web service standards (XML, WSDL).
Availability	-
License	-
Status	Prototype
Requirements/dependencies	SVG plugin is needed. Installation requirements: Java2
Web references	-
<b>ATHENA metadata</b>	
Contact person	maurizio.megliola@txt.it
Contributors	TXT e-solutions
Provided by project/activity	<ul style="list-style-type: none"> <li>• A4 – Interoperability Framework and Services for Networked Enterprises</li> </ul>
Deliverables representing result	Deliverable number and name
Contribution to key result	<ul style="list-style-type: none"> <li>• 8. Interoperability Infrastructure</li> </ul>
Used in pilot	-
Deliverable providing evaluation	-

### 7.3.14. BRMF (Business Resource Management Framework)

#### 7.3.14.1. Datasheet

<b>Solution data</b>	
Name	BRMF (Business Resource Management

	Framework)
Result type	Platform or platform component
Description/functionality	<p>The BRMF is a middleware solution that supports the creation of distributed business applications on top of a general purpose, industry grade P2P infrastructure. It enables decentralized management of business resources, such as business documents, business services, or product models, that are developed and controlled by different, loosely coupled partners. For these applications, BRMF provides a virtualization layer that enables controlled publishing, sharing, and synchronization of resources in a distributed network – without necessarily implying the existence of a central control instance or repository. This makes the BRMF approach particularly suitable for use in open, dynamic business applications as they occur in the context of Virtual Enterprises, outsourcing, and re-organizations caused by mergers and acquisitions activities.</p> <p>BRMF is not a stand-alone interoperability solution. It offers a flexible and distributed/decentralized information and collaboration space infrastructure which can be combined with technologies and solutions such as model-driven development, ontologies, and agent technology to leverage the scope of today’s interoperability solutions towards a higher degree of openness, scalability, flexibility, and adaptability.</p>
Benefits to interoperability	<ul style="list-style-type: none"> <li>• Basic execution environment for document-centric, event-driven business processes: easily develop new business applications with focus on rapidly changing event driven processes requiring collaborative management/tracking of changes to shared business documents or business objects</li> <li>• Environment for decentralized management of business documents, services, and different types of models (e.g. product models): basic interoperability infrastructure, on top of which different applications can be networked in order to share resources of common interest, and make them available to developers or other business communities in an open and dynamic way.</li> <li>• Run-time adaptability and communication interoperability for web services and BP execution engines: basic communication</li> </ul>

	functionality allowing web service communication across network barriers such as NAT routers and firewalls with only minimal configuration overhead required.
Supported models/methodologies	-
Supported input interfaces	-
Supported output interfaces	-
Validation/demonstration	Validation through two demonstrators, a proof-of-concept distributed workflow demonstrator (travel expense claim) showing robust BP execution, and a collaborative document revision demonstrator using the Automotive collaborative product development scenario (sourcing phase). Further validation in B5 testing and piloting activities in Year 3 of ATHENA.
Standards compliance	Use of web service standards (XML, WSDL); architectural supports for integration of application level standards
Availability	-
License	-
Status	Prototype
Requirements/dependencies	<ul style="list-style-type: none"> <li>Requires Java 5, network connection.</li> <li>Uses the Siemens Resource Management Framework, which is not publicly available outside of ATHENA.</li> </ul>
Web references	
<b>ATHENA metadata</b>	
Contact person	Jörg Müller, SIEMENS
Contributors	SIEMENS
Provided by project/activity	<ul style="list-style-type: none"> <li>A6 – Model-driven and Adaptive Interoperability Architectures</li> </ul>
Deliverables representing result	-
Contribution to key result	<ul style="list-style-type: none"> <li>13. Model-driven and Adaptable Interoperability Framework and Infrastructure</li> </ul>
Used in pilot	-

Deliverable providing evaluation	-
----------------------------------	---

### 7.3.15. Capability Tables Loader (CTL)

#### 7.3.15.1. Datasheet

Name	Capability Tables Loader (CTL)
Result type	Tool
Description/functionality	<p>The Capability Tables Loader (CTL) is a middleware solution that supports the enterprises that want to start an interoperability project.</p> <p>The enterprises involved in the collaboration evaluation, use the ATHENA services and tools in order to capture and classify the available existing models, metamodels, schemas of documents and reports, and any other artefact that the enterprise can or is willing to expose. The result of this phase represents the enterprise Capability Table.</p>
Benefits to interoperability	<ul style="list-style-type: none"> <li>• Implementation of a SOA platform for Capability Tables management.</li> <li>• Easy integration with File Analyzers (for example BPEL and WSDL) for subsequent evaluation and analysis of models and business documents in view of identifying and resolving interoperability mismatches.</li> </ul>
Supported models/methodologies	-
Supported input interfaces	-
Supported output interfaces	-
Validation/demonstration	-
Standards compliance	Use of web service standards (XML, WSDL).
Availability	-
License	-
Status	Prototype
Requirements/dependencies	<ul style="list-style-type: none"> <li>• Requires Java 2, network connection.</li> <li>• Uses the BPEL Analyzer and the WSDL Analyzer, which are not publicly available outside of ATHENA.</li> </ul>
Web references	-

ATHENA metadata	
Contact person	maurizio.megliola@txt.it
Contributors	TXT e-solutions
Provided by project/activity	<ul style="list-style-type: none"> <li>A4 – Interoperability Framework and Services for Networked Enterprises</li> </ul>
Deliverables representing result	-
Contribution to key result	<ul style="list-style-type: none"> <li>8. Interoperability Infrastructure</li> </ul>
Used in pilot	-
Deliverable providing evaluation	-

### 7.3.16. Conformance testing suite

#### 7.3.16.1. Datasheet

Solution data	
Name	Conformance testing suite
Result type	Tool
Description/functionality	<ul style="list-style-type: none"> <li>This results covers two subresults in the context of conformance testing. The first result addresses the conformacne of EXPRESS messages The second result looks at the interactions of organisations and tests conformance of the interaction itself.</li> <li>Conformance and interoperability testing are procedures that should be performed to validate and assure the quality of the global integrated system.</li> <li>To assure the quality of a system it is necessary to verify the conformity of its data with the conceptual model. In the proposed approach, the data is exchanged in XML and the reference model originally defined in EXPRESS.</li> <li>Usually, this kind of validation is designated by conformance testing and is performed in two different stages: Firstly, model validation, and secondly rules/constraints validation. The conformance-test Johnson plugin uses the output of this transformation tool, since the validation of the XML data with his EXPRESS model will be executed using the model converted represented XSD and the rules/constraints in Schematron format</li> </ul>

	<ul style="list-style-type: none"> <li>• The ATHENA B2B Conformance Testing Systems allows a user to check conformance of an interaction with a business partners. For that, the system can be configured to behave like the business partner. The basic assumption is that a B2B interaction is defined by the content and structure of the messages exchanged as well as the sequence in which the messages are exchanged. The system allows testing all three mentioned aspects.</li> <li>• Furthermore, the system is configurable and provides detailed reports.</li> <li>• The system is complemented with a set of integrated tools that support the definition of test suites based on existing instances of messages.</li> <li>• The system not only provides conformance testing capabilities but can also be used for simulation of (business) functionality. For example the system was used in the EADS pilot to simulate the PDM system.</li> </ul>
Benefits to interoperability	<ul style="list-style-type: none"> <li>• Assure Conformance to a given message exchange pattern</li> <li>• Allow business partners to validate conformance before doing online test</li> <li>• Supports all XML based Industry Standards</li> </ul>
Supported models/methodologies	-
Supported input interfaces	-
Supported output interfaces	-
Validation/demonstration	<ul style="list-style-type: none"> <li>• INTRACOM: Product and Portfolio Management</li> <li>• CRF: Automotive Pilot</li> <li>• AIDIMA: eProcurement Pilot</li> </ul>
Standards compliance	-
Availability	-
License	-
Status	-
Requirements/dependencies	-
Web references	-
<b>ATHENA metadata</b>	
Contact person	-



Contributors	-
Provided by project/activity	<ul style="list-style-type: none"> <li>A5: Planned and Customisable Service-Oriented Architectures</li> <li>B5: Piloting including Technology Testing Coordination and Pilot Infrastructure</li> </ul>
Deliverables representing result	WD.B5.4: Reference Manual to ATHENA Prototyping Services (M24)
Contribution to key result	<ul style="list-style-type: none"> <li>8: Interoperability Infrastructure</li> </ul>
Used in pilot	<ul style="list-style-type: none"> <li>INTRACOM: Product and Portfolio Management</li> <li>CRF: Automotive Pilot</li> <li>AIDIMA: eProcurement Pilot</li> </ul>
Deliverable providing evaluation	D.A5.5 "Validation of Research Results" (M24)

### 7.3.17. EKA Metamodel Feature for Eclipse

#### 7.3.17.1.

##### Datasheet

Solution data	
Name	EKA Metamodel Feature for Eclipse
Result type	Tool (for model creation and management)
Description/functionality	The Eclipse plugin for the EKA metamodel is an EMF (Eclipse Modeling Framework) based metamodel for persistent EKA model management inside de Eclipse IDE. This tool allows user to manage EKA models in a persistent way using the Eclipse environment.
Benefits to interoperability	<ul style="list-style-type: none"> <li>EKA has been defined as a core technology for enterprise model interchange in Athena project.</li> <li>EKA is the container technology for POP* interchange models</li> </ul>
Supported models/methodologies	POP*
Supported input interfaces	-
Supported output interfaces	-
Validation/demonstration	ESI has used this tool as intermediate result for an entire MDA transformation chain

Standards compliance	-
Availability	<ul style="list-style-type: none"> <li>Source code</li> </ul>
License	<a href="#">Eclipse Public License</a>
Status	Prototype
Requirements/dependencies	Eclipse
Web references	<ul style="list-style-type: none"> <li>Website: <a href="http://pim4soa.sourceforge.net/">http://pim4soa.sourceforge.net/</a></li> <li>Source code: <a href="http://sourceforge.net/projects/pim4soa/">http://sourceforge.net/projects/pim4soa/</a></li> </ul>
<b>ATHENA metadata</b>	
Contact person	Iñaki Peña, European Software Institute (ESI)
Contributors	-
Provided by project/activity	<ul style="list-style-type: none"> <li>A1 – Enterprise Modelling in the Context of Collaborative Enterprises</li> <li>A6 – Model-driven and Adaptive Interoperability Architectures</li> </ul>
Deliverables representing result	-
Contribution to key result	<ul style="list-style-type: none"> <li>13. Model-driven and Adaptable Interoperability Framework and Infrastructure</li> </ul>
Used in pilot	-
Deliverable providing evaluation	-

### 7.3.18. EKA to PIM4SOA Transformation Feature for Eclipse

#### 7.3.18.1. Datasheet

<b>Solution data</b>	
Name	EKA to PIM4SOA Transformation Feature for Eclipse
Result type	Model transformation.
Description/functionality	<p>Transformation from EKA metamodel (ecore) to PIM4SOA (ecore) metamodel using MTF. There is ongoing work on this issue.</p> <p>In this project we have adopted a MDA vision to enterprise architectures as well as systems architectures to build model based systems to avoid</p>

	<p>or reduce the loss of information between the enterprise models and the ICT systems focussed on SOA. Within A6, the POP* to PIM4SOA transformation work has been focused on defining transformations of the POP* process models. Preliminary consideration has been given to service and information model transformations; their completion is subject to future work.</p>
Benefits to interoperability	<p>Bridging this gap using model based transformations techniques we ensure the separation of concerns providing flexibility and traceability. This transformation does not produce a complete PIM4SOA model. This is mainly caused by two reasons. The first one is due to the fact that the current state of the POP* profile does not includes all aspects (process, organisation, product, etc.). And finally we cannot generate all PIM4SOA elements. Therefore this transformation generates in terms of PIM4SOA elements a skeleton with the following elements: documents (sketched), collaborations, collaborationUses, roles and their relationships.</p>
Supported models/methodologies	-
Supported input interfaces	-
Supported output interfaces	-
Validation/demonstration	INTRACOM scenario.
Standards compliance	-
Availability	<ul style="list-style-type: none"> <li>• Source code</li> </ul>
License	<ul style="list-style-type: none"> <li>• <a href="#">Eclipse Public License</a></li> </ul>
Status	Prototype
Requirements/dependencies	<ul style="list-style-type: none"> <li>• <a href="#">EclipseEclipse</a></li> </ul>
Web references	<ul style="list-style-type: none"> <li>• Website: <a href="http://pim4soa.sourceforge.net/">http://pim4soa.sourceforge.net/</a></li> <li>• Source code: <a href="http://sourceforge.net/projects/pim4soa/">http://sourceforge.net/projects/pim4soa/</a></li> </ul>
<b>ATHENA metadata</b>	
Contact person	Gorka Benguria, European Software Institute (ESI)

Contributors	-
Provided by project/activity	<ul style="list-style-type: none"> <li>• A1 – Enterprise Modelling in the Context of Collaborative Enterprises</li> <li>• A5 – Planned and Customisable Service-Oriented Architectures</li> <li>• A6 – Model-driven and Adaptive Interoperability Architectures</li> </ul>
Deliverables representing result	-
Contribution to key result	<ul style="list-style-type: none"> <li>• 13. Model-driven and Adaptable Interoperability Framework and Infrastructure</li> </ul>
Used in pilot	-
Deliverable providing evaluation	-

### 7.3.19. Enterprise Interoperability Degree Measurement (EIDM)

#### 7.3.19.1. Datasheet

Solution data	
Name	Enterprise Interoperability Degree Measurement (EIDM)
Result type	<ul style="list-style-type: none"> <li>• Model</li> <li>• Methodology, guidelines</li> </ul>
Description/functionality	The Enterprise Interoperability Degree Measurement (EIDM), is performed in an inter-enterprise context (or between two heterogeneous systems). The measure of interoperability degree is decomposed into two sub-measures defined as compatibility and operational performances.
Benefits to interoperability	This model provides a means to identify interoperability barriers faced by organizations that want to interoperate.
Supported models/methodologies	
Supported input interfaces	
Supported output interfaces	
Validation/demonstration	The EIDM has been validated in the scenario described in the A8 sub-project.
Standards compliance	

Availability	<ul style="list-style-type: none"> <li>Documentation about the EIMM-SME can be found in the DA8.2 deliverable and its appendixes. The model is also supported by a web based tool.</li> </ul>
License	
Status	Prototype
Requirements/dependencies	
Web references	
<b>ATHENA metadata</b>	
Contact person	David Chen, UB1
Contributors	UB1, ESI
Provided by project/activity	<ul style="list-style-type: none"> <li>A8 – SME Interoperability in Practice</li> </ul>
Deliverables representing result	D.A8.2 - Guidelines and Best Practices for Applying the ATHENA Interoperability Framework to Support SME Participation in Digital Ecosystems
Contribution to key result	<ul style="list-style-type: none"> <li>7. Guidelines and Best Practices</li> </ul>
Used in pilot	
Deliverable providing evaluation	

### 7.3.20. Enterprise Interoperability Maturity Model (EIMM)

#### 7.3.20.1. Datasheet

<b>Solution data</b>	
Name	Enterprise Interoperability Maturity Model (EIMM)
Result type	<ul style="list-style-type: none"> <li>Model</li> <li>Methodology/guidelines</li> </ul>
Description/functionality	The Enterprise Interoperability Maturity Model (EIMM) is a staged model, based on the structure of the Capability Maturity Model™ (CMM™) which is being successfully applied to assess and improve processes in organizations. The EIMM helps to assess an organization's maturity level concerning the use of enterprise

	models as well as the capability of these models to enable the company to be part of a collaboration. Based on an EIMM assessment, companies will be guided to choose the right concepts for improving their capabilities, by taking into account actual market and enterprise challenges.
Benefits to interoperability	The approach will also be used for planning and implementing new enterprise concepts in short and mid term perspectives. Here the integration of today missing aspects like organisational capabilities and skills will allow an easier and more sustainable application of EM.
Supported models/methodologies	-
Supported input interfaces	-
Supported output interfaces	-
Validation/demonstration	The EIMM and accompanying questionnaire has been validated and improved in two subsequent project internal trials
Standards compliance	n/a
Availability	-
License	-
Status	Prototype
Requirements/dependencies	-
Web references	-
<b>ATHENA metadata</b>	
Contact person	<ul style="list-style-type: none"> <li>• Stefan Schuster, ESI</li> <li>• Thomas Knothe, DFKI</li> </ul>
Contributors	ESI, IPK, SINTEF
Provided by project/activity	<ul style="list-style-type: none"> <li>• A1 – Enterprise Modelling in the Context of Collaborative Enterprises</li> </ul>
Deliverables representing result	part of D.A.1.4.1 <a href="http://www.athena-ip.org/index.php?option=com_docman&amp;task=do">http://www.athena-ip.org/index.php?option=com_docman&amp;task=do</a>
Contribution to key result	<ul style="list-style-type: none"> <li>• 3. Interoperability Impact Analysis Model</li> </ul>
Used in pilot	-

Deliverable providing evaluation	D.A1.6.1 "Benefit-Assessment" (M24)
----------------------------------	-------------------------------------

### 7.3.21. Enterprise Interoperability Maturity Model for SMEs (EIMM-SME)

#### 7.3.21.1. Datasheet

Solution data	
Name	Enterprise Interoperability Maturity Model for SMEs (EIMM-SME)
Result type	<ul style="list-style-type: none"> <li>• Model</li> <li>• Methodology, guidelines</li> </ul>
Description/functionality	The Enterprise Interoperability Maturity Model for SMEs (EIMM-SME) is a staged model, based on the structure of the EIMM developed within ATHENA. The new model has been adapted for SME environments. Based on an EIMM-SME assessment, SMEs will be guided to choose the right concepts for improving their capabilities, by taking into account actual market and enterprise challenges.
Benefits to interoperability	This approach will help SMEs assess their interoperability maturity in order to be able to participate in digital ecosystems. The model provides a roadmap for establishing core capabilities required for participating in digital ecosystems.
Supported models/methodologies	
Supported input interfaces	
Supported output interfaces	
Validation/demonstration	The EIMM which this model is based on has been validated through internal trials.
Standards compliance	
Availability	Documentation about the EIMM-SME can be found in the DA8.2 deliverable and its appendixes. The model is also supported by an Excel sheet.
License	
Status	Prototype
Requirements/dependencies	

Web references	
<b>ATHENA metadata</b>	
Contact person	Igor Santos, ESI
Contributors	ESI, IPK, DFKI, UB1
Provided by project/activity	<ul style="list-style-type: none"> <li>A1 – Enterprise Modelling in the Context of Collaborative Enterprises</li> <li>A8 – SME Interoperability in Practice</li> </ul>
Deliverables representing result	D.A8.2 - Guidelines and Best Practices for Applying the ATHENA Interoperability Framework to Support SME Participation in Digital Ecosystems
Contribution to key result	<ul style="list-style-type: none"> <li>7. Guidelines and Best Practices</li> </ul>
Used in pilot	
Deliverable providing evaluation	

### 7.3.22. EXP2PIM4SOA (Express to PIM4SOA model transformation)

#### 7.3.22.1. Datasheet

<b>Solution data</b>	
Name	EXP2PIM4SOA (Express to PIM4SOA model transformation)
Result type	Model transformation
Description/functionality	Harmonization of PIM4SOA with the XMI produced by EXP2XMI.
Benefits to interoperability	This transformation will be one step further in the link between R&D and Standardization and it will also be beneficial in the use of the PIM4SOA in industrial environments that already are relying on ISO standards like STEP.
Supported models/methodologies	-
Supported input interfaces	-
Supported output interfaces	-
Validation/demonstration	-
Standards compliance	-



Availability	-
License	-
Status	-
Requirements/dependencies	-
Web references	-
<b>ATHENA metadata</b>	
Contact person	Hugo Vieira, UNINOVA
Contributors	UNINOVA
Provided by project/activity	<ul style="list-style-type: none"> <li>A6 – Model-driven and Adaptive Interoperability Architectures</li> </ul>
Deliverables representing result	D.A6.4 “Model-Driven and Adaptable Interoperability Infrastructure” (M24)
Contribution to key result	<ul style="list-style-type: none"> <li>13. Model-driven and Adaptable Interoperability Framework and Infrastructure</li> </ul>
Used in pilot	-
Deliverable providing evaluation	-

### 7.3.23. EXP2SCH (Express to Schematron model transformation)

#### 7.3.23.1. Datasheet

<b>Solution data</b>	
Name	EXP2SCH (Express to Schematron model transformation)
Result type	Model transformation
Description/functionality	Is a Tool that makes the mapping of the behavioral part (rules) of a model written in STEP-EXPRESS (ISO 10303-11) to the Schematron language (ISO/IEC 19757). The output provided will be used for the conformance testing.
Benefits to interoperability	-
Supported models/methodologies	-
Supported input interfaces	EXPRESS

Supported output interfaces	Schematron 1.5
Validation/demonstration	-
Standards compliance	-
Availability	-
License	-
Status	-
Requirements/dependencies	-
Web references	-
<b>ATHENA metadata</b>	
Contact person	Hugo Vieira, UNINOVA
Contributors	UNINOVA
Provided by project/activity	<ul style="list-style-type: none"> <li>A6 – Model-driven and Adaptive Interoperability Architectures</li> </ul>
Deliverables representing result	-
Contribution to key result	-
Used in pilot	e-Procurement, Aerospace CPD?
Deliverable providing evaluation	-

### 7.3.24. EXP2UML (Express to UML model transformation)

#### 7.3.24.1. Datasheet

<b>Solution data</b>	
Name	EXP2UML (Express to UML model transformation)
Result type	Model transformation
Description/functionality	EXP2UML is a tool for the transformation of (STEP) EXPRESS schemas to UML. The transformation is implemented using ATL, which conforms to the QVT standard.
Benefits to interoperability	-
Supported models/methodologies	-
Supported input interfaces	-

Supported output interfaces	-
Validation/demonstration	-
Standards compliance	-
Availability	-
License	-
Status	-
Requirements/dependencies	-
Web references	-
<b>ATHENA metadata</b>	
Contact person	Uwe Kaufmann, IPK
Contributors	IPK
Provided by project/activity	-
Deliverables representing result	-
Contribution to key result	-
Used in pilot	e-Procurement, Aerospace CPD
Deliverable providing evaluation	-

### 7.3.25. EXP2XMI (Express to XMI model transformation)

#### 7.3.25.1. Datasheet

<b>Solution data</b>	
Name	EXP2XMI (Express to XMI model transformation)
Result type	Model transformation
Description/functionality	The EXP2XMI model transformation tool, is a tool that makes the mapping of the structural part of a model described in STEP-EXPRESS (ISO 10303-25) to XMI (XML Metadata Interchange) according to the recommendations of Part25 (ISO 10303-25).
Benefits to interoperability	<ul style="list-style-type: none"> <li>OMG is using UML along with its Model Driven Architectures (MDA) to enable the integration of different applications by explicitly relating their models. This enables interoperability and</li> </ul>

	<p>supports systems evolution (deployment choices) as platform technologies change.</p> <ul style="list-style-type: none"> <li>• Compared to STEP, UML related tools abound, which facilitates the task of many organizations that want to use it. It is also major problem to visualize the relationships between STEP constructs and the other information types that were used in their development process. UML, through its class diagrams and through its profile extensibility mechanism, provides much simpler mechanisms to perform that task [2].</li> <li>• If the tool is feed with an EXPRESS service metamodel, the output will provide means for the integration with the Athena PIM4SOA format.</li> </ul>
Supported models/methodologies	-
Supported input interfaces	EXPRESS
Supported output interfaces	XMI version 1.1
Validation/demonstration	Validation and demonstration in the B5 testing and piloting activities, particularly for AIDIMA and EADS scenarios.
Standards compliance	<ul style="list-style-type: none"> <li>• EXPRESS: ISO10303-11 (<a href="http://www.tc184-sc4.org">www.tc184-sc4.org</a>)</li> <li>• Part 25 : ISO 10303-25 (<a href="http://www.tc184-sc4.org">www.tc184-sc4.org</a>)</li> <li>• XMI: <a href="http://www.omg.org">www.omg.org</a></li> </ul>
Availability	<ul style="list-style-type: none"> <li>• Installed service / solution</li> <li>• Binary download</li> </ul>
License	-
Status	Prototype
Requirements/dependencies	<ul style="list-style-type: none"> <li>• Requires JAVA 1.4 or above.</li> <li>• Depends on the Part 25 evolution, which specifies a standard way to map from the EXPRESS standard to the XMI standard</li> </ul>
Web references	-
<b>ATHENA metadata</b>	
Contact person	Hugo Vieira, UNINOVA
Contributors	UNINOVA
Provided by project/activity	<ul style="list-style-type: none"> <li>• A6 – Model-driven and Adaptive Interoperability Architectures</li> </ul>
Deliverables representing result	-

Contribution to key result	<ul style="list-style-type: none"> <li>13. Model-driven and Adaptable Interoperability Framework and Infrastructure</li> </ul>
Used in pilot	-
Deliverable providing evaluation	-

### 7.3.26. EXP2XSD (Express to XSD model transformation)

#### 7.3.26.1. Datasheet

Solution data	
Name	EXP2XSD (Express to XSD model transformation)
Result type	Model transformation
Description/functionality	The EXP2XSD model transformation tool, is a tool that makes the mapping of the structural part of a model described in STEP-EXPRESS (ISO 10303-28) to XML Schemas (XSD) according to the recommendations of Part28 (ISO 10303-28).
Benefits to interoperability	<ul style="list-style-type: none"> <li>The STEP standard currently defines three neutral data exchange formats – ASCII text file (STEP Part 21), programming language APIs (STEP Part 22-27, 29), and XML (STEP Part 28).</li> <li>Only a few relatively specialized applications, such as STEP translators and STEP repositories, use the API programming language, called Standard Data Access Interface (SDAI) [5,9]. Unlike the STEP Part 21 syntax, XML data is easily extensible and is supported by numerous inexpensive and widely used software tools. Thus, from the perspective of a typical programmer, it is easier to render XML data into forms that are suitable for human perusal [2].</li> <li>The output of this transformation tool is the representation of the model in XML that will be used for implementation purposes, and also in the conformance testing process.</li> </ul>
Supported models/methodologies	-
Supported input interfaces	EXPRESS
Supported output interfaces	XML Schemas
Validation/demonstration	Validation and demonstration in the B5 testing

	and piloting activities, particularly for AIDIMA and EADS scenarios.
Standards compliance	<ul style="list-style-type: none"> <li>EXPRESS: ISO10303-11 (<a href="http://www.tc184-sc4.org">www.tc184-sc4.org</a>)</li> <li>Part 28 : ISO 10303-28 (<a href="http://www.tc184-sc4.org">www.tc184-sc4.org</a>)</li> <li>XSD: <a href="http://www.w3c.org">www.w3c.org</a></li> </ul>
Availability	Installed service/solution
License	-
Status	Prototype
Requirements/dependencies	<ul style="list-style-type: none"> <li>Requires JAVA 1.4 or above.</li> <li>Depends on the Part 28 evolution, which specifies a standard way to map from the EXPRESS standard to the XSD standard</li> </ul>
Web references	
<b>ATHENA metadata</b>	
Contact person	Hugo Vieira, UNINOVA
Contributors	UNINOVA
Provided by project/activity	<ul style="list-style-type: none"> <li>A6 – Model-driven and Adaptive Interoperability Architectures</li> </ul>
Deliverables representing result	-
Contribution to key result	<ul style="list-style-type: none"> <li>13. Model-driven and Adaptable Interoperability Framework and Infrastructure</li> </ul>
Used in pilot	-
Deliverable providing evaluation	-

### 7.3.27. Gabriel

#### 7.3.27.1. Datasheet

<b>Solution data</b>	
Name	Gabriel
Result type	Tool
Description/functionality	Gabriel is a tool that service enables the cross organisational business processes. Using Gabriel a complete tool chain ranging from Enterprise Modelling tools like MO <sup>2</sup> GO, via

	<p>Maestro to Johnson is created.</p> <p>The modelling part of Gabriel allows linking the cross-organisational business processes modelled in Maestro with Web services. Web services can either be used to execute tasks in private processes or to provide the interface over which messages are sent to the partners. The runtime part of Gabriel links the task execution in the process to service calls.</p>
Benefits to interoperability	Seamless integration of business processes and a service-oriented architecture for deployment.
Supported models/methodologies	-
Supported input interfaces	-
Supported output interfaces	-
Validation/demonstration	EADS: Change management process
Standards compliance	-
Availability	-
License	-
Status	-
Requirements/dependencies	-
Web references	-
<b>ATHENA metadata</b>	
Contact person	Ulrike Greiner, SAP
Contributors	SAP
Provided by project/activity	<ul style="list-style-type: none"> <li>A4 – Interoperability Framework and Services for Networked Enterprises</li> </ul>
Deliverables representing result	WD.A4.3 “Process-based Interoperability Infrastructure 1st issue” (M24)
Contribution to key result	<ul style="list-style-type: none"> <li>8. Interoperability Infrastructure</li> <li>10. Cross-Organisational Business Process Modelling and Enactment</li> <li>12. Service Composition Framework</li> </ul>
Used in pilot	EADS: Change management process
Deliverable providing evaluation	D.A2.5 “Validation of Research Results” (M24)

### 7.3.28. Gap Table Analyser (GTA)

#### 7.3.28.1. Datasheet

Solution data	
Name	Gap Table Analyser (GTA)
Result type	Tool
Description/functionality	The Gap Table Analyser (GTA) is a web (middleware) solution that supports the enterprises willing to collaborate in the context of an interoperability project. For these enterprises, GTA provides a SOA platform with services performing the comparison of the enterprises exposed Capability Tables. This comparison is performed with the support of appropriate comparison agents, and the result is captured in the Gap Table.
Benefits to interoperability	<ul style="list-style-type: none"> <li>• Implementation of a SOA platform for comparing Capability Tables.</li> <li>• Identify interoperability mismatches and collect them in Gap Table cells.</li> <li>• Generation of report with details about the identified interoperability mismatches as derived from models and artefacts.</li> </ul>
Supported models/methodologies	-
Supported input interfaces	-
Supported output interfaces	-
Validation/demonstration	-
Standards compliance	Use of web service standards (XML, WSDL)
Availability	-
License	-
Status	Prototype
Requirements/dependencies	<ul style="list-style-type: none"> <li>• Requires Java 2, network connection.</li> <li>• Uses the BPEL Analyzer and the WSDL Analyzer, which are not publicly available outside of ATHENA.</li> </ul>
Web references	-



<b>ATHENA metadata</b>	
Contact person	maurizio.megliola@txt.it
Contributors	TXT e-solutions
Provided by project/activity	<ul style="list-style-type: none"> <li>A4 – Interoperability Framework and Services for Networked Enterprises</li> </ul>
Deliverables representing result	-
Contribution to key result	<ul style="list-style-type: none"> <li>8. Interoperability Infrastructure</li> </ul>
Used in pilot	-
Deliverable providing evaluation	-

### 7.3.29. GRAI Methodology

#### 7.3.29.1. Datasheet

<b>Solution data</b>	
Name	GRAI Methodology
Result type	Methodology
Description/functionality	The GRAI Methodology is a set of methodological modules which contributes to the improvement of companies' performances through BPM techniques.
Benefits to interoperability	-
Supported models/methodologies	<p>The GRAI Methodology covers several application domains:</p> <ul style="list-style-type: none"> <li>Enterprise Re-engineering</li> <li>Selection and Implementation of Solutions in :               <ul style="list-style-type: none"> <li>Information Technology</li> <li>Technology</li> <li>Organisation</li> </ul> </li> <li>Performance Indicators</li> <li>Industrial Strategy</li> <li>Support the implementation of Quality approach</li> <li>Knowledge Management</li> <li>...</li> </ul>
Supported input interfaces	-
Supported output interfaces	-

Validation/demonstration	-
Standards compliance	-
Availability	-
Software license	-
Status	-
Requirements/dependencies	-
Web references	<ul style="list-style-type: none"> <li>Website: <a href="http://www.graisoft.com/home/index.php?option=content&amp;task=view">http://www.graisoft.com/home/index.php?option=content&amp;task=view</a></li> </ul>
<b>ATHENA metadata</b>	
Contact person	-
Contributors	-
Provided by project/activity	-
Deliverables representing result	-
Contribution to key result	-
Used in pilot	-
Deliverable providing evaluation	-

### 7.3.30. GraiTools

#### 7.3.30.1. Datasheet

<b>Solution data</b>	
Name	GraiTools
Result type	Modelling tool
Description/functionality	GraiTools proposes a complete environment of BPM, enterprise modelling and project management while being based on tested techniques and concepts (model GRAI and Project management). The whole of its tools are integrated in only one platform, facilitating the implementation of the projects and the maintenance of the enterprise reference frames; they accompany the permanent effort of adaptation to the market trends.
Benefits to interoperability	-

Supported models/methodologies	GRAI Methodology
Supported input interfaces	-
Supported output interfaces	-
Validation/demonstration	-
Standards compliance	-
Availability	-
License	Commercial
Status	Commercial
Requirements/dependencies	-
Web references	<ul style="list-style-type: none"> <li>Website: <a href="http://www.graisoft.com/home/index.php?option=content&amp;task=view">http://www.graisoft.com/home/index.php?option=content&amp;task=view</a></li> </ul>
<b>ATHENA metadata</b>	
Contact person	-
Contributors	-
Provided by project/activity	-
Deliverables representing result	-
Contribution to key result	-
Used in pilot	-
Deliverable providing evaluation	-

### 7.3.31. Solution template

#### 7.3.31.1. Datasheet

<b>Solution data</b>	
Name	JACK
Result type	Development environments for intelligent agents.
Description/functionality	JACK is an environment for building, running and integrating commercial-grade multi-agent systems using a component-based approach.
Benefits to interoperability	-

Supported models/methodologies	The JACK Agent Language is a programming language that extends Java with agent-oriented concepts, such as: <ul style="list-style-type: none"> <li>• Agents</li> <li>• Capabilities</li> <li>• Events</li> <li>• Plans</li> <li>• Agent Knowledge Bases (Databases)</li> <li>• Resource and Concurrency Management</li> </ul>
Supported input interfaces	-
Supported output interfaces	-
Validation/demonstration	-
Standards compliance	-
Availability	-
License	-
Status	-
Requirements/dependencies	-
Web references	<ul style="list-style-type: none"> <li>• Website: <a href="http://www.agent-software.com/shared/products/index.html">http://www.agent-software.com/shared/products/index.html</a></li> </ul>
<b>ATHENA metadata</b>	
Contact person	n/a
Contributors	n/a
Provided by project/activity	n/a
Deliverables representing result	n/a
Contribution to key result	n/a
Used in pilot	n/a
Deliverable providing evaluation	n/a

### 7.3.32. Johnson

#### 7.3.32.1. Datasheet

<b>Solution data</b>	
Name	Johnson

Result type	<ul style="list-style-type: none"> <li>• Software tool, infrastructure</li> </ul>
Description/functionality	<p>This result represents a runtime tool for enacting Service-Oriented Architectures. Johnson enables users to enact most of the roles typically found in an SOA, thereby enacting complex SOA scenarios by sending real SOAP messages between Web services without having to write a single line of code.</p> <p>Johnson features a Web-based user interface designed to closely resemble Web-based email applications, with the only difference that SOAP messages and Web services endpoints are used in place of email messages and email addresses. The user can see incoming SOAP messages in the Inbox and create outgoing SOAP messages in the Outbox that will be sent to external Web services. A powerful user-interface generator relieves the user from having to deal with XML documents by generating forms for displaying and editing any XML-based data type.</p> <p>A processing module was also developed for keeping an audit trail of messages, which forms the basis for troubleshooting and performance measurement. The headers of SOAP messages are turned into RDF and stored in an RDF store.</p>
Benefits to interoperability	The solution provides easy access to SOA's for industrial users. It offers a low entry barrier to web services and is easily extensible with additional modules using its plug-in interface.
Supported models/methodologies	-
Supported input interfaces	Web-based + WSDL
Supported output interfaces	SOAP messages
Validation/demonstration	<ul style="list-style-type: none"> <li>• AIAG: eKanban Pilot</li> <li>• AIDIMA: eProcurement Pilot</li> <li>• EADS: Change management process</li> <li>• CRF: Automotive Pilot</li> <li>• CAS: Car Configuration</li> <li>• INTRACOM: Product Portfolio Management</li> </ul>
Standards compliance	-
Availability	-
License	-
Status	-

Requirements/dependencies	-
Web references	-
<b>ATHENA metadata</b>	
Contact person	Julien Vayssiere, SAP
Contributors	SAP
Provided by project/activity	<ul style="list-style-type: none"> <li>A5 – Planned and Customisable Service-Oriented Architectures</li> </ul>
Deliverables representing result	<ul style="list-style-type: none"> <li>D.A5.3: Architecture of SOA platforms (M21)</li> <li>D.A5.4: Execution Framework(s) for Planned and Customisable Service-Oriented Architectures (M21)</li> </ul>
Contribution to key result	<ul style="list-style-type: none"> <li>8: Interoperability Infrastructure</li> <li>12. Service Composition Framework</li> </ul>
Used in pilot	<ul style="list-style-type: none"> <li>AIAG: eKanban Pilot</li> <li>AIDIMA: eProcurement Pilot</li> <li>EADS: Change management process</li> <li>CRF: Automotive Pilot</li> <li>CAS: Car Configuration</li> <li>INTRACOM: Product Portfolio Management</li> </ul>
Deliverable providing evaluation	D.A5.5 “Validation of Research Results” (M24)

### 7.3.33. Maestro

#### 7.3.33.1. Datasheet

<b>Solution data</b>	
Name	Maestro
Result type	Modelling tool
Description/functionality	<p>Maestro is a business process modelling tool on a technical level that implements the methodology for modelling cross-organisational business processes (CBPs) developed in ATHENA A2. To model CBPs each partner starts from a private process describing the steps executed in its organisation. Then a view process is created that provides a process-oriented interface to the partners whilst at the same time hiding internal process steps that should not be published. The CBP then</p>

	<p>links the view processes of all partners and defines at which steps data and messages area exchanged between partners.</p> <p>With Maestro it is possible to model private processes, view processes and CBPs and their interlinkage in a semi-automated way. When defining view processes out of private processes, links are kept automatically and for CBPs, sender and receiver nodes are automatically inserted. Processes can be saved into the repository of the runtime execution engine Nehemiah.</p>
Benefits to interoperability	<ul style="list-style-type: none"> <li>• Implementation of view approach into modelling tool</li> <li>• Support of required CBP mechanism that selectively hides details of private processes, whilst providing a process-oriented interface to the outside world, facilitating interweaving into partner processes</li> <li>• Joint modelling of CBPs</li> </ul>
Supported models/methodologies	-
Supported input interfaces	v1.2.03: Proprietary Maestro Interface
Supported output interfaces	v1.2.03: Proprietary Maestro/Nehemiah interface
Validation/demonstration	<p>Validation through demonstrator.</p> <ul style="list-style-type: none"> <li>• modelling of CRF sourcing scenario</li> <li>• modelling of eProcurement CBP, demonstration of supplier sub-scenario</li> <li>• modelling of change management process of EADS scenario</li> </ul>
Standards compliance	No direct links.
Availability	-
License	-
Status	Prototype
Requirements/dependencies	No dependencies. Installation requirements: Java 1.4
Web references	<ul style="list-style-type: none"> <li>• Web: <a href="#">User guide</a></li> </ul>
<b>ATHENA metadata</b>	
Contact person	sonia.lippe@sap.com urike.greiner@sap.com
Contributors	SAP

Provided by project/activity	<ul style="list-style-type: none"> <li>A2 – Cross-Organisational Business Processes</li> </ul>
Deliverables representing result	<ul style="list-style-type: none"> <li>D.A2.2: Specification of a Cross-Organisational Business Process Model (M15)</li> <li>D.A2.4: Enactment of Cross-Organisational Business Processes (M21)</li> </ul>
Contribution to key result	<ul style="list-style-type: none"> <li>10. Cross-Organisational Business Process Modelling and Enactment</li> </ul>
Used in pilot	<ul style="list-style-type: none"> <li>AIDIMA: eProcurement Pilot</li> <li>EADS: Change management process</li> <li>CRF: Automotive Pilot</li> <li>CAS: Car Configuration</li> </ul>
Deliverable providing evaluation	D.A2.5 “Validation of Research Results” (M24)

### 7.3.34. Mediated Support Tool (MST)

#### 7.3.34.1. Datasheet

Solution data	
Name	Mediated Support Tool (MST)
Result type	Tool
Description/functionality	<p>The Mediated Support Tool (MST) is a web (middleware) solution that supports the enterprises to collaborate into an interoperability project.</p> <p>The support is provided in terms of a collaboration tool for mediated collaborations.</p> <p>The architecture is designed around the concept of the Moderator or Mediator. This type of collaboration is based on a central module that manages the interactions between experts and the project mediator, tracking the involved actions, in the process of convergence towards agreed interfaces and protocols.</p>
Benefits to interoperability	<ul style="list-style-type: none"> <li>The support tool contains a module for document handling and a module for recording all executed transactions, dealing with associated repositories.</li> <li>The tool is a web-based one, with instant messaging, session and document handling facilities for sharing documents in interactive sessions.</li> <li>Experts contribute to the solution of the problem,</li> </ul>



	so the module supports the supply of documents and textual support information associated to the generated solution
Supported models/methodologies	-
Supported input interfaces	-
Supported output interfaces	-
Validation/demonstration	-
Standards compliance	Use of web service standards (XML, WSDL)
Availability	-
License	-
Status	Prototype
Requirements/dependencies	No dependencies. Installation requirements: Java 2 and .NET Framework.
Web references	-
<b>ATHENA metadata</b>	
Contact person	maurizio.megliola@txt.it
Contributors	TXT e-solutions
Provided by project/activity	<ul style="list-style-type: none"> <li>A4 – Interoperability Framework and Services for Networked Enterprises</li> </ul>
Deliverables representing result	-
Contribution to key result	<ul style="list-style-type: none"> <li>8. Interoperability Infrastructure</li> </ul>
Used in pilot	-
Deliverable providing evaluation	-

### 7.3.35. Metis

#### 7.3.35.1. Datasheet

<b>Solution data</b>	
Name	Metis
Result type	Modelling tool
Description/functionality	<ul style="list-style-type: none"> <li>Advanced modeling, analysis, and reporting</li> </ul>

	<ul style="list-style-type: none"> <li>using the first enterprise-class EA repository</li> <li>Automated data collection, aggregation and analysis for timely and relevant decision-making information</li> </ul>
Benefits to interoperability	-
Supported models/methodologies	-
Supported input interfaces	-
Supported output interfaces	-
Validation/demonstration	-
Standards compliance	-
Availability	<ul style="list-style-type: none"> <li>Binary download</li> </ul>
License	Commercial
Status	Commercial
Requirements/dependencies	-
Web references	<ul style="list-style-type: none"> <li>Website: <a href="http://www.troux.com/metis/">http://www.troux.com/metis/</a></li> </ul>
<b>ATHENA metadata</b>	
Contact person	Håvard D. Jørgensen, AKM
Contributors	-
Provided by project/activity	-
Deliverables representing result	-
Contribution to key result	-
Used in pilot	-
Deliverable providing evaluation	-

### 7.3.36. Model-driven integration of JACK and Web Services

#### 7.3.36.1. Datasheet

<b>Solution data</b>	
Name	Model-driven integration of JACK and Web services
Result type	Methodology/guideline

Description/functionality	The model-driven integration of the Jack Agent platform into a Web service environment extends the PIM4SOA to JACK Agent platform model transformation. With the help of the WSDL Analyzer, a generated instance of the Jack meta-model is adapted to a specific scenario with possibly changing partners.
Benefits to interoperability	The model-driven integration of the Jack agent platform into a Web service environment contributes to the integration of agent technologies into the ATHENA interoperability framework.
Supported models/methodologies	-
Supported input interfaces	-
Supported output interfaces	-
Validation/demonstration	Will be done with respect to the different scenarios in the ATHENA project.
Standards compliance	-
Availability	-
License	-
Status	Concept
Requirements/dependencies	<ul style="list-style-type: none"> <li>• Instance of Jack meta-model</li> <li>• WSDL Analyzer</li> </ul>
Web references	-
<b>ATHENA metadata</b>	
Contact person	Klaus Fischer, DFKI
Contributors	DFKI
Provided by project/activity	<ul style="list-style-type: none"> <li>• A5 – Planned and Customisable Service-Oriented Architectures</li> <li>• A6 – Model-driven and Adaptive Interoperability Architectures</li> </ul>
Deliverables representing result	-
Contribution to key result	<ul style="list-style-type: none"> <li>• 12. Service Composition Framework</li> <li>• 13. Model-driven and Adaptable Interoperability Framework and Infrastructure</li> </ul>
Used in pilot	-

Deliverable providing evaluation	-
----------------------------------	---

### 7.3.37. MOF repository

#### 7.3.37.1. Datasheet

Solution data	
Name	MOF repository
Result type	Model repository
Description/functionality	<p>The Enhanced Model Repository is where both service providers and consumers can store models that describe the services in their environment: including information models that describe the messages consumed and produced by these services and the public business processes and choreographies that these services may participate in.</p> <p>The repository developed in A6 provides a public model repository which allows for the collaborative annotation of models and also for richer query and search than is currently provided by MOF-based repositories.</p> <p>After uploading a metamodel, defined with the MOF 1.4, instances of that metamodel (i.e. models) can be stored and uploaded using a standard XMI serialisation mechanism.</p> <p>When a transformation has been defined between the MOF model and an RDF representation, models may then be annotated with RDF statements. This then allows models to be queried using the RDQL language.</p>
Benefits to interoperability	It provides a public model repository which allows for the collaborative annotation of models and also for richer query and search than is currently provided by MOF-based repositories.
Supported models/methodologies	MOF 1.4
Supported input interfaces	RDQL
Supported output interfaces	MOF 1.4
Validation/demonstration	-
Standards compliance	The repository supports storing models that are defined according to the MOF version 1.4. It can

	support serialisation, deserialisation of models and metamodels according to the XMI version 1.1. It can also support annotation of models with RDF statements and querying of models using the RDQL.
Availability	<ul style="list-style-type: none"> <li>Hosted service</li> </ul>
License	-
Status	Prototype
Requirements/dependencies	Does the tool rely on any commercial or licensed software? If so, on which?
Web references	-
<b>ATHENA metadata</b>	
Contact person	Murray Spork, SAP
Contributors	SAP
Provided by project/activity	<ul style="list-style-type: none"> <li>A6 – Model-driven and Adaptive Interoperability Architectures</li> </ul>
Deliverables representing result	-
Contribution to key result	<ul style="list-style-type: none"> <li>13. Model-driven and Adaptable Interoperability Framework and Infrastructure</li> </ul>
Used in pilot	-
Deliverable providing evaluation	-

### 7.3.38. MO<sup>2</sup>GO (Method for Object Oriented Business Process Optimization)

#### 7.3.38.1. Datasheet

<b>Solution data</b>	
Name	MO <sup>2</sup> GO (Method for Object Oriented Business Process Optimization)
Result type	Modelling tool
Description/functionality	MO <sup>2</sup> GO is an enterprise modelling tool. MO <sup>2</sup> GO supports the integrated enterprise modelling (IEM). MO <sup>2</sup> GO NG has as well been extended to support modelling of CBPs on the business level. It also provides export functionality to transform process models from the business

	level to the technical level. This supports re-use of process models so that users do not have to completely re-model processes when enriching them with information relevant for execution.
Benefits to interoperability	<ul style="list-style-type: none"> <li>• Implementation of view approach into business level modelling tools</li> <li>• Support of required CBP mechanism that selectively hides details of private processes, whilst providing a process-oriented interface to the outside world</li> </ul>
Supported models/methodologies	<ul style="list-style-type: none"> <li>• Integrated Enterprise Modelling (IEM)</li> </ul>
Supported input interfaces	<ul style="list-style-type: none"> <li>• <a href="#">MO2GO IEM/MPCE interface</a></li> </ul>
Supported output interfaces	<ul style="list-style-type: none"> <li>• <a href="#">MO2GO IEM/BPDM interface</a></li> <li>• <a href="#">MO2GO IEM/Maestro interface</a></li> <li>• <a href="#">MO2GO IEM/MPCE interface</a></li> <li>• <a href="#">MO2GO IEM/PIM4SOA interface</a></li> <li>• <a href="#">MO2GO IEM/UML interface</a></li> </ul>
Validation/demonstration	<ul style="list-style-type: none"> <li>• AIDIMA: eProcurement Pilot</li> <li>• EADS: Change management process</li> <li>• CRF: Automotive Pilot</li> </ul>
Standards compliance	-
Availability	-
License	Commercial
Status	Commercial
Requirements/dependencies	-
Web references	<ul style="list-style-type: none"> <li>• Website: <a href="http://www.moogo.de/overview/index.html">http://www.moogo.de/overview/index.html</a></li> </ul>
<b>ATHENA metadata</b>	
Contact person	Frank-Walter Jaekel, IPK
Contributors	-
Provided by project/activity	-
Deliverables representing result	-
Contribution to key result	-
Used in pilot	<ul style="list-style-type: none"> <li>• AIDIMA: eProcurement Pilot</li> <li>• EADS: Change management process</li> </ul>

	<ul style="list-style-type: none"> <li>• CRF: Automotive Pilot</li> </ul>
Deliverable providing evaluation	-

### 7.3.39. Monitoring Support Tool (MST)

#### 7.3.39.1. Datasheet

Solution data	
Name	Monitoring Support Tool (MST)
Result type	Tool
Description/functionality	<p>The Monitoring Support Tool (MST) is a web (middleware) solution that supports the enterprises willing to collaborate into an interoperability project.</p> <p>For these enterprises, MST provides a SOA platform featuring services to collect information about the execution of the runtime tools for the collaboration enactments at all levels when available. This information is collected in a database and exposed for monitoring purposes.</p> <p>The tool also supplies reasoning services to help determine the distance between the expectations for the tools and what the tools actually deliver.</p> <p>This reasoning is obtained by letting the Moderators upload and review log files from ATHENA tools, track them (depending on date, priority, type, module) and associate a warning for each log file.</p>
Benefits to interoperability	<ul style="list-style-type: none"> <li>• Implementation of a SOA platform for log files management.</li> <li>• Extract log information from the different Action Line A tools in order to report discrepancies in Operations.</li> <li>• Give hints about possible causes of error.</li> </ul>
Supported models/methodologies	-
Supported input interfaces	-
Supported output interfaces	-
Validation/demonstration	-
Standards compliance	Use of web service standards (XML, WSDL).
Availability	-

License	-
Status	Prototype
Requirements/dependencies	No dependencies. Installation requirements: Java 2
Web references	-
<b>ATHENA metadata</b>	
Contact person	maurizio.megliola@txt.it
Contributors	TXT e-solutions
Provided by project/activity	<ul style="list-style-type: none"> <li>A4 – Interoperability Framework and Services for Networked Enterprises</li> </ul>
Deliverables representing result	-
Contribution to key result	<ul style="list-style-type: none"> <li>8. Interoperability Infrastructure</li> </ul>
Used in pilot	-
Deliverable providing evaluation	-

### 7.3.40. MPCE (Modelling Platform for Collaborative Enterprises)

#### 7.3.40.1. Datasheet

<b>Solution data</b>	
Solution name	MPCE (Modelling Platform for Collaborative Enterprises)
Result type	Modelling platform
Description/functionality	The Modelling Platform for collaborative enterprises (MPCE) supports the POP* language and provides model management and model exchange services. The MPCE can be used as a web-service hosted somewhere or can be locally installed.
Benefits to interoperability	<p>Across this infrastructure organizations are able to exchange and share process models, translating between the proprietary languages of the different tools. The key parts of this infrastructure are</p> <ul style="list-style-type: none"> <li>Enterprise Knowledge Architecture (EKA), a modeling framework and XML schema for representing any kind of enterprise models and enterprise modeling languages,</li> </ul>



	<ul style="list-style-type: none"> <li>MPCE web services that tools use to access models in the repository.</li> </ul>
Supported models/methodologies	-
Supported input interfaces	-
Supported output interfaces	-
Validation/demonstration	The INTRACOM PPM pilot.
Standards compliance	-
Availability	-
License	-
Status	-
Requirements/dependencies	-
Web references	-
<b>ATHENA metadata</b>	
Contact person	Håvard D. Jørgensen, AKM
Contributors	Other partners contributing to the result
Provided by project/activity	<ul style="list-style-type: none"> <li>A1 – Enterprise Modelling in the Context of Collaborative Enterprises</li> </ul>
Deliverables representing result	D.A1.5.2 “Collaborative Modeling Platform – 1st Prototype”
Contribution to key result	<ul style="list-style-type: none"> <li>9. Collaborative Enterprise Modelling Platform</li> </ul>
Used in pilot	INTRACOM: Product/project portfolio management
Deliverable providing evaluation	D.A1.6.1 “Benefit-Assessment” (M24)

### 7.3.41. Nehemiah

#### 7.3.41.1. Datasheet

<b>Solution data</b>	
Name	Nehemiah
Result type	Tool (enactment engine)

Description/functionality	<ul style="list-style-type: none"> <li>• Nehemiah is a process execution engine that is able to execute cross-organisational business processes (CBPs) modelled with the methodology developed in ATHENA A2. Nehemiah directly depends on the Maestro modelling tool, i.e. CBPs modelled in Maestro are deployed directly to the Nehemiah repository.</li> <li>• To execute a CBP each partner has to deploy his private process, the view processes from all partners participating in the CBP, and the CBP. CBPs are executed without a central process engine. Therefore the process enactment engines resp. infrastructures of the different partners have to talk to each other. Nehemiah uses Gabriel and Johnson to send /receive messages to / from other partners. Therefore it can talk to any process engine that provides a Web service based interface.</li> <li>• Nehemiah also provides simulation functionality that can be used to simulate the CBP modelled in Maestro before actually deploying it for execution.</li> </ul>
Benefits to interoperability	Implementation of view approach into modelling tool: meets requirement of selected visibility, privacy and flexible
Supported models/methodologies	-
Supported input interfaces	-
Supported output interfaces	-
Validation/demonstration	<p>Validation through demonstrator.</p> <ul style="list-style-type: none"> <li>• modelling of CRF sourcing scenario</li> <li>• modelling of eProcurement CBP, demonstration of supplier sub-scenario</li> <li>• modelling of change management process of EADS scenario</li> </ul>
Standards compliance	No direct links.
Availability	Install service / solution available
License	-
Status	Prototype
Requirements/dependencies	No dependencies. Installation requirements: Java 1.4
Web references	<a href="http://athena.troux.com/AKMii/Default.aspx?SystemID=4&amp;FolderID">http://athena.troux.com/AKMii/Default.aspx?SystemID=4&amp;FolderID</a>

<b>ATHENA metadata</b>	
Contact person	<ul style="list-style-type: none"> <li>• Sonia Lippe, SAP</li> <li>• Ulrike Greiner, SAP</li> </ul>
Contributors	SAP
Provided by project/activity	<ul style="list-style-type: none"> <li>• A2 – Cross-Organisational Business Processes</li> </ul>
Deliverables representing result	D.A2.4: Architecture for Enactment and Integration of Cross-Organizational Business Processes (M21)
Contribution to key result	<ul style="list-style-type: none"> <li>• 10. Cross-Organisational Business Process Modelling and Enactment</li> </ul>
Used in pilot	<ul style="list-style-type: none"> <li>• AIDIMA: eProcurement Pilot</li> <li>• EADS: Change management process</li> <li>• CRF: Automotive Pilot</li> <li>• CAS: Car Configuration</li> </ul>
Deliverable providing evaluation	D.A2.5 “Validation of Research Results” (M24)

### 7.3.42. OPAL (Object, Process, Actor modelling Language)

#### 7.3.42.1. Datasheet

<b>Solution data</b>	
Name	OPAL (Object, Process, Actor modelling Language)
Result type	Ontology methodology
Description/functionality	The OPAL (Object, Process, Actor modelling Language) methodology has the goals to provide an ontology building method capable of supporting and guiding the ontology modeller; and to provide a number of inherent constraints, associated to the above mentioned categories, used to guarantee a better quality for the built ontology.
Benefits to interoperability	-
Supported models/methodologies	-
Supported input interfaces	-
Supported output interfaces	-
Validation/demonstration	-

Standards compliance	-
Availability	-
License	-
Status	-
Requirements/dependencies	-
Web references	-
<b>ATHENA metadata</b>	
Contact person	LEKS
Contributors	LEKS
Provided by project/activity	<ul style="list-style-type: none"> <li>A3 – Knowledge Support and Semantic Mediation Solutions</li> </ul>
Deliverables representing result	D.A3.2: Ontology Authoring and Management System for informational knowledge and Business Processes (M20)
Contribution to key result	<ul style="list-style-type: none"> <li>11. Ontology-based Semantic Annotation and Reconciliation method/language/tool</li> </ul>
Used in pilot	<ul style="list-style-type: none"> <li>AIAG: eKanban Pilot</li> <li>AIDIMA: eProcurement Pilot</li> <li>EADS: Change management process</li> <li>CRF: Automotive Pilot</li> </ul>
Deliverable providing evaluation	-

### 7.3.43. PIM4SOA (Platform-independent model for service-oriented architecture)

#### 7.3.43.1. Datasheet

<b>Solution data</b>	
Name	PIM4SOA (Platform-independent model for service-oriented architecture)
Result type	Metamodel
Description/functionality	ATHENA is addressing business and IT needs with specialized and appropriate methods and tools. To bridge the gap between business (comparable to the CIM level in MDA) and IT (comparable to the PSM level in MDA) ATHENA defined an intermediate technical level which is

	<p>comparable to the PIM level in MDA. To ensure consistence across all levels the transformation of models between the technical (~PIM) and IT level (~PSM) level is crucial. ATHENA provided multiple transformations in this context addressing different metamodels.</p> <p>The PIM4SOA metamodel defines an abstract language to specify executable business processes that execute within an enterprise and may collaborate between otherwise independent business processes executing in different business units or enterprises.</p> <p>The main objective of the specification is:</p> <ul style="list-style-type: none"> <li>• The ability to exchange business process specifications between modelling tools, and between tools and execution environments.</li> </ul> <p>PIM4SOA is closely aligned and has been based on the Business Process Definition Metamodel that is in the process of standardization by OMG. However as the standardization did not completed in the timeframe of Athena the PIM4SOA metamodel was developed as a simplified version.</p> <p>In order to reduce the gap between enterprise models and the service oriented implementations, we have applied a model driven architecture approach to enterprise architectures in the implementation of the PIM for SOA (PIM4SOA).</p> <p>The PIM4SOA identifies four aspects where specific concerns can be addressed:</p> <ul style="list-style-type: none"> <li>• Information: in the context of virtual enterprises information represents one of the most important elements and other aspects are based on it.</li> <li>• Service: our main intention is to be able to describe SOA indepently from the technologies used. Service represents business accesible functionality</li> <li>• Process: Processes describe a set of interactions amongst services in terms of messages exchanged</li> <li>• Quality of service: Based on the current proposal, we have integrated the main elements to describe quality of services.</li> </ul>
<p>Benefits to interoperability</p>	<ul style="list-style-type: none"> <li>• Basically the PIM4SOA allows the definition of SOA models independently from the technology used.</li> <li>• In addition it allows to share SOA models and to bridge the gap between enterprise models and ICT implementations.</li> </ul>

Supported models/methodologies	-
Supported input interfaces	-
Supported output interfaces	-
Validation/demonstration	The eprocurement scenario has been used to validate this approach.
Standards compliance	This metamodel is based on: <ul style="list-style-type: none"> <li>EMOF: Essential Meta Object Facility (<a href="http://www.omg.org/docs/ptc/03-10-04.pdf">http://www.omg.org/docs/ptc/03-10-04.pdf</a>)<a href="http://www.omg.org/docs/ptc/03-10-04.pdf">http://www.omg.org/docs/ptc/03-10-04.pdf</a></li> <li>UML Profile for modeling quality of service and fault tolerance characteristics and mechanisms. Object Management Group (<a href="http://www.omg.org/docs/ptc/04-09-01.pdf">http://www.omg.org/docs/ptc/04-09-01.pdf</a>)<a href="http://www.omg.org/docs/ptc/04-09-01.pdf">http://www.omg.org/docs/ptc/04-09-01.pdf</a></li> </ul>
Availability	<ul style="list-style-type: none"> <li>Documentation</li> <li>Ecore metamodel format</li> </ul>
License	<a href="#">Eclipse Public License</a>
Status	Prototype
Requirements/dependencies	n/a
Web references	<ul style="list-style-type: none"> <li>Website: <a href="http://pim4soa.sourceforge.net/">http://pim4soa.sourceforge.net/</a></li> <li>Source code: <a href="http://sourceforge.net/projects/pim4soa/">http://sourceforge.net/projects/pim4soa/</a></li> </ul>
<b>ATHENA metadata</b>	
Contact person	Xabier Larrucea, ESI
Contributors	ESI, IBM, SINTEF
Provided by project/activity	<ul style="list-style-type: none"> <li>A6 – Model-driven and Adaptive Interoperability Architectures</li> </ul>
Deliverables representing result	D.A6.4 “Model-Driven and Adaptable Interoperability Infrastructure” (M24)
Contribution to key result	<ul style="list-style-type: none"> <li>13. Model-driven and Adaptable Interoperability Framework and Infrastructure</li> </ul>
Used in pilot	<ul style="list-style-type: none"> <li>AIDIMA: eProcurement pilot</li> </ul>
Deliverable providing evaluation	D.A6.4 “Model-Driven and Adaptable Interoperability Infrastructure” (M24)

### 7.3.44. PIM4SOA Execution and Simulation Platform

### 7.3.44.1. Datasheet

<b>Solution data</b>	
Name	PIM4SOA Execution and Simulation Platform
Result type	Model Execution Platform
Description/functionality	<ul style="list-style-type: none"> <li>The model execution platform is a runtime environment for UML models. Since UML 2.0 provides rich behavioural semantics, the resulting models can be executed – just as any application, created with the help of a common programming language. The model execution platform provides a generic extensible model execution engine. The engine provides mechanisms for the realization of behavioural semantics and the execution and observation of model behaviour.</li> <li>The runtime environment permits creation of a set of design time support tools – debuggers, test generators, etc. all of which use the model execution engine to predict model behaviour.</li> <li>The platform is being developed as part of the MODELWARE project. ATHENA contribution contains implementation of interoperability related features and PIM4SOA support.</li> <li>Execution of models, described with the help of the UML profile for PIM4SOA.</li> </ul>
Benefits to interoperability	Execution of PIM level models, permits vizualization of the execution of the interoperating solution at the abstract level
Supported models/methodologies	-
Supported input interfaces	-
Supported output interfaces	-
Validation/demonstration	To be used on the aerospace scenario.
Standards compliance	-
Availability	Installed service/solution
License	Awaiting licence confirmation
Status	Prototype
Requirements/dependencies	Rational Software Modeler or Software Architect, UML profile for PIM for SOA, Model Execution Engine plugin for Eclipse

Web references	-
<b>ATHENA metadata</b>	
Contact person	Sergey Olovsky, IBM
Contributors	IBM
Provided by project/activity	<ul style="list-style-type: none"> <li>A6 – Model-driven and Adaptive Interoperability Architectures</li> </ul>
Deliverables representing result	-
Contribution to key result	<ul style="list-style-type: none"> <li>13. Model-driven and Adaptable Interoperability Framework and Infrastructure</li> </ul>
Used in pilot	-
Deliverable providing evaluation	-

### 7.3.45. PIM4SOA Metamodel Feature for Eclipse

#### 7.3.45.1. Datasheet

<b>Solution data</b>	
Name	PIM4SOA Metamodel Feature for Eclipse
Result type	Tool (for basic edition, and management of PIM4SOA models).
Description/functionality	This is a Eclipse plugging that allows to edit PIM4SOA models. Besides it provides the necessary interfaces for the later transformation of the models to higher and lower abstraction levels
Benefits to interoperability	PIM4SOA establish and intermediente layer between the enterprise layer and the platform layer. This allow to address in a first stage the interoperability of the business logic independently form the especific platform details.
Supported models/methodologies	PIM4SOA
Supported input interfaces	-
Supported output interfaces	-
Validation/demonstration	Tested in demonstration environments
Standards compliance	-



Availability	<ul style="list-style-type: none"> <li>• Binary download</li> <li>• Source code</li> </ul>
License	<ul style="list-style-type: none"> <li>• Eclipse Public License</li> </ul>
Status	Prototype
Requirements/dependencies	<ul style="list-style-type: none"> <li>• Eclipse</li> </ul>
Web references	<ul style="list-style-type: none"> <li>• Website: <a href="http://pim4soa.sourceforge.net/">http://pim4soa.sourceforge.net/</a></li> <li>• User guide: <a href="http://pim4soa.sourceforge.net/metamodels/pim4soa_html.html">http://pim4soa.sourceforge.net/metamodels/pim4soa_html.html</a></li> </ul>
<b>ATHENA metadata</b>	
Contact person	Xabier Larrucea, ESI
Contributors	ESI, IBM, SINTEF
Provided by project/activity	<ul style="list-style-type: none"> <li>• A6 – Model-driven and Adaptive Interoperability Architectures</li> </ul>
Deliverables representing result	-
Contribution to key result	<ul style="list-style-type: none"> <li>• 13. Model-driven and Adaptable Interoperability Framework and Infrastructure</li> </ul>
Used in pilot	-
Deliverable providing evaluation	-

### 7.3.46. PIM4SOA to BPEL Transformation Feature for Eclipse

#### 7.3.46.1. Datasheet

<b>Solution data</b>	
Name	PIM4SOA to BPEL Transformation Feature for Eclipse
Result type	Model transformation
Description/functionality	<ul style="list-style-type: none"> <li>• Allows users to take a PIM4SOA model (e.g. generated from higher level tooling) and convert to an execution platform (BPEL).</li> <li>• Rather than a direct model to text transformation, the web service layer PSM transformations make use of platform specific models. For the BPEL transform, an Ecore / EMF model of BPEL has been created to manipulate the transformed process.</li> </ul>

	<ul style="list-style-type: none"> <li>The EMF implementation of BPEL is generated directly from the XSD (BPEL schema). Tools within the Eclipse Modelling framework create the ecore xmi schema and Java classes for the implementation. These are packaged as a separate Eclipse plugin (required to use the transformation). A useful effect of this approach is that the models directly serialise to a BPEL conformant document.</li> </ul>
Benefits to interoperability	<ul style="list-style-type: none"> <li>Allows user to take a Platform independent business model (e.g. generated from higher level tooling) and convert to an execution platform (BPEL).</li> <li>Conversion will require some level of human interaction as by nature a platform independent model cannot contain all platform specific information.</li> </ul>
Supported models/methodologies	-
Supported input interfaces	-
Supported output interfaces	-
Validation/demonstration	<ul style="list-style-type: none"> <li>Reviewer appraisal (Athens meeting)</li> <li>Presentation and evaluation by users (Munich)</li> </ul>
Standards compliance	-
Availability	Installed service / solution
License	Awaiting licence confirmation
Status	Prototype
Requirements/dependencies	Rational Software Modeler or Software Architect, PIM4SOA EMF Plugins
Web references	-
<b>ATHENA metadata</b>	
Contact person	Anthony Beardsmore, IBM
Contributors	IBM
Provided by project/activity	<ul style="list-style-type: none"> <li>A2 – Cross-Organisational Business Processes</li> <li>A6 – Model-driven and Adaptive Interoperability Architectures</li> </ul>
Deliverables representing result	D.A6.4 “Model-Driven and Adaptable Interoperability Infrastructure” (M24)

Contribution to key result	<ul style="list-style-type: none"> <li>13. Model-driven and Adaptable Interoperability Framework and Infrastructure</li> </ul>
Used in pilot	<ul style="list-style-type: none"> <li>AIDIMA: eProcurement pilot</li> </ul>
Deliverable providing evaluation	D.A6.4 “Model-Driven and Adaptable Interoperability Infrastructure” (M24)

### 7.3.47. PIM4SOA to JACK model transformation

#### 7.3.47.1. Datasheet

Solution data	
Name	PIM4SOA to JACK model transformation
Result type	Model transformation
Description/functionality	<ul style="list-style-type: none"> <li>A model mapping between the meta-model for PIM4SOA and the meta-model for Jack is specified on a conceptual level. Current work is the investigation in how far the model mapping can be automated using the Eclipse MTF model transformation framework.</li> <li>The model transformation contributes to the integration of agent technologies into the ATHENA interoperability framework. Even if the mapping between the meta-models is specified on a conceptual level only, this already contributes to the interation. The conceptual mapping relates concepts in agent design to concepts of the model-driven approach to service-oriented architectures that is in the focus of ATHENA. If an automated model transformation is achieved, the agent platform provides an alternative excution environment that is likely to extend the flexibility and power of the PIM4SOA.</li> </ul>
Benefits to interoperability	The model transformation contributes to the integration of agent technologies into the ATHENA interoperability framework. Even if the mapping between the meta-models is specified on a conceptual level only, this already contributes to the interation. The conceptual mapping relates contepts in agent design to concepts of the model-driven approach to service-oriented architectures that is in the focus of ATHENA. If an automated model transformation is achieved, the agent platform provides an alternative excution environment

	that is likely to extend the flexibility and power of the PIM4SOA as it is currently proposed.
Supported models/methodologies	-
Supported input interfaces	-
Supported output interfaces	-
Validation/demonstration	Mapping of concrete models that are provided at the PIM4SOA level for different demonstrators will be investigated.
Standards compliance	<ul style="list-style-type: none"> <li>• <a href="http://www.fipa.org/">http://www.fipa.org/</a></li> <li>• <a href="http://www.agentlink.org/">http://www.agentlink.org/</a></li> </ul>
Availability	-
License	-
Status	Concept
Requirements/dependencies	-
Web references	-
<b>ATHENA metadata</b>	
Contact person	Klaus Fischer, DFKI
Contributors	DFKI
Provided by project/activity	<ul style="list-style-type: none"> <li>• A6 – Model-driven and Adaptive Interoperability Architectures</li> </ul>
Deliverables representing result	-
Contribution to key result	<ul style="list-style-type: none"> <li>• 13. Model-driven and Adaptable Interoperability Framework and Infrastructure</li> </ul>
Used in pilot	-
Deliverable providing evaluation	-

### 7.3.48. PIM4SOA to OWL-S model transformation

#### 7.3.48.1. Datasheet

<b>Solution data</b>	
Name	PIM4SOA to OWL-S model transformation
Result type	Model transformation

Description/functionality	<p>This result is a visual modelling tool for semantic enrichment of Web Service descriptions.</p> <p>UMT2OWLS is a visual tool for modelling the semantic enrichment of Web service interfaces description. It is based on the existing UMT-QVT tool (an open source project). This extension allows transformation of UML-based models for describing Web services directly to their WSDL and OWL-S representations. In this manner a real model-driven development is achieved. The resulted WSDL and OWL-S documents can be used in a Web services registry for the description of the service interface including all the necessary business and technical information</p>
Benefits to interoperability	The result allows for a richer description of Web Services using semantics.
Supported models/methodologies	-
Supported input interfaces	-
Supported output interfaces	-
Validation/demonstration	-
Standards compliance	-
Availability	-
License	-
Status	-
Requirements/dependencies	-
Web references	-
<b>ATHENA metadata</b>	
Contact person	-
Contributors	-
Provided by project/activity	<ul style="list-style-type: none"> <li>• A5: Planned and Customisable Service-Oriented Architectures</li> </ul>
Deliverables representing result	<ul style="list-style-type: none"> <li>• D.A5.2 Model and Specification of service description and usage as well as advanced concepts (M18)</li> <li>• D.A5.4: Execution Framework(s) for Planned and Customisable Service-Oriented Architectures (M21)</li> </ul>

Contribution to key result	<ul style="list-style-type: none"> <li>• 13: Model-driven and Adaptable Interoperability Framework and Infrastructure</li> <li>• 12: Service Composition Framework</li> </ul>
Used in pilot	-
Deliverable providing evaluation	D.A5.5 "Validation of Research Results" (M24)

### 7.3.49. PIM4SOA to Web services model transformations

#### 7.3.49.1. Datasheet

Solution data	
Name	PIM4SOA to Web Services Transformations
Result type	Model transformations
Description/functionality	<p>A collection (feature) of plugins (tools) supporting transformations between PIM4SOA models (Eclipse Modelling Framework) and Web service models (EMF). The following two-way transformations are supported:</p> <ul style="list-style-type: none"> <li>• PIM4SOA &lt;-&gt; XSD</li> <li>• PIM4SOA &lt;-&gt; WSDL</li> <li>• PIM4SOA &lt;-&gt; BPEL</li> </ul>
Benefits to interoperability	<ul style="list-style-type: none"> <li>• The "integrated" collection of these transformation tools will ensure that we are able to develop new Web services (top-down approach) and integrating existing Web services (bottom-up approach) into an interoperable solution at the platform independent level (PIM4SOA). There exists a number of different and competing Web service standards. The model transformations should help us to develop Web service applications according to "interoperability" best practices in existence today. One such guideline is the WS-I profile for Web services.</li> <li>• The PIM4SOA also aims to support two-way transformations for P2P and Agent execution platforms so that technical interoperability between heterogeneous systems consisting of Web services, P2P and Agents can be managed.</li> </ul>
Supported models/methodologies	-
Supported input interfaces	-
Supported output interfaces	-

Validation/demonstration	See plans for individual transformations.
Standards compliance	<ul style="list-style-type: none"> <li>Web Services Interoperability (WS-I) Organization, <a href="http://www.ws-i.org/">http://www.ws-i.org/</a></li> </ul>
Availability	<ul style="list-style-type: none"> <li>Installed service / solution</li> <li>Object files available</li> </ul>
License	<ul style="list-style-type: none"> <li>Licensing issues as the different plugins (tools) that make up this collection (feature) probably will be licensed under different terms.</li> </ul>
Status	Prototype
Requirements/dependencies	<p>Requires the following plugins (tools) developed in ATHENA A6:</p> <ul style="list-style-type: none"> <li>PIM4SOA metamodel</li> <li>PIM4SOA &lt;-&gt; XSD transformation</li> <li>PIM4SOA &lt;-&gt; WSDL transformation</li> <li>PIM4SOA &lt;-&gt; BPEL transformation</li> </ul>
Web references	-
<b>ATHENA metadata</b>	
Contact person	<ul style="list-style-type: none"> <li>Gorka Benguria, ESI</li> <li>Tor Neple, SINTEF</li> <li>Anthony Beardsmore, IBM</li> </ul>
Contributors	ESI, SINTEF, IBM
Provided by project/activity	<ul style="list-style-type: none"> <li>A6 – Model-driven and Adaptive Interoperability Architectures</li> </ul>
Deliverables representing result	-
Contribution to key result	<ul style="list-style-type: none"> <li>13. Model-driven and Adaptable Interoperability Framework and Infrastructure</li> </ul>
Used in pilot	-
Deliverable providing evaluation	-

### 7.3.50. PIM4SOA to WSDL Transformation Feature for Eclipse

#### 7.3.50.1. Datasheet

<b>Solution data</b>	
Name	PIM4SOA to WSDL Transformation Feature for

	Eclipse
Result type	Model transformation
Description/functionality	This Eclipse plugin takes a PIM4SOA model instance and transforms it into a description of a Web Service in the Web Service Description Language (WSDL). The generated WSDL contains an XSD schema representing the information elements for the Web Service. The information for the WSDL is taken from the Services and Information segments of the PIM4SOA metamodel.
Benefits to interoperability	This result connects the PIM4SOA metamodel to a widely used SOA platform, Web Services. The benefit to interoperability is there when more similar transformations are written to support other SOA platforms.
Supported models/methodologies	
Supported input interfaces	
Supported output interfaces	
Validation/demonstration	The transformations have been used on the Athena e-procurement scenario.
Standards compliance	<ul style="list-style-type: none"> <li>• <a href="http://www.w3.org/TR/ws-arch/">http://www.w3.org/TR/ws-arch/</a></li> </ul>
Availability	
License	<ul style="list-style-type: none"> <li>• <a href="#">Eclipse Public License</a></li> </ul>
Status	
Requirements/dependencies	The plugin runs under Eclipse and requires that the PIM4SOA plugin and the Eclipse Web Tools plugins (with prerequisites) are installed
Web references	<ul style="list-style-type: none"> <li>• Website: <a href="http://pim4soa.sourceforge.net/">http://pim4soa.sourceforge.net/</a></li> <li>• User guide: <a href="http://pim4soa.sourceforge.net/transformations/pim4soa2wsdl_html.html">http://pim4soa.sourceforge.net/transformations/pim4soa2wsdl_html.html</a></li> </ul>
<b>ATHENA metadata</b>	
Contact person	Tor Neple, SINTEF
Contributors	SINTEF
Provided by project/activity	<ul style="list-style-type: none"> <li>• A6 – Model-driven and Adaptive Interoperability Architectures</li> </ul>



Deliverables representing result	D.A6.4 “Model-Driven and Adaptable Interoperability Infrastructure” (M24)
Contribution to key result	<ul style="list-style-type: none"> <li>13. Model-driven and Adaptable Interoperability Framework and Infrastructure</li> </ul>
Used in pilot	<ul style="list-style-type: none"> <li>AIDIMA: eProcurement pilot</li> </ul>
Deliverable providing evaluation	D.A6.4 “Model-Driven and Adaptable Interoperability Infrastructure” (M24)

### 7.3.51. PIM4SOA to XSD Transformation Feature for Eclipse

#### 7.3.51.1. Datasheet

Solution data	
Name	PIM4SOA to XSD Transformation Feature for Eclipse
Result type	Model transformation
Description/functionality	Starting form a PIM4SOA file it generates a XML Schema file
Benefits to interoperability	It generates a platform resource form the business logic abstracting from the platform details. If the platform changes, it is only needed to change the transformation and generate again the platform resource.
Supported models/methodologies	-
Supported input interfaces	-
Supported output interfaces	-
Validation/demonstration	This has been tested in the demonstrators
Standards compliance	-
Availability	<ul style="list-style-type: none"> <li>Source code</li> </ul>
License	<ul style="list-style-type: none"> <li><a href="#">Eclipse Public License</a></li> </ul>
Status	Prototype
Requirements/dependencies	<ul style="list-style-type: none"> <li><a href="#">Eclipse</a></li> </ul>
Web references	<ul style="list-style-type: none"> <li>Website: <a href="http://pim4soa.sourceforge.net/">http://pim4soa.sourceforge.net/</a></li> <li>Source code:</li> </ul>

	<a href="http://sourceforge.net/projects/pim4soa/">http://sourceforge.net/projects/pim4soa/</a>
<b>ATHENA metadata</b>	
Contact person	Gorka Benguria, ESI
Contributors	ESI
Provided by project/activity	<ul style="list-style-type: none"> <li>• A5 – Planned and Customisable Service-Oriented Architectures</li> <li>• A6 – Model-driven and Adaptive Interoperability Architectures</li> </ul>
Deliverables representing result	D.A6.4 “Model-Driven and Adaptable Interoperability Infrastructure” (M24)
Contribution to key result	<ul style="list-style-type: none"> <li>• 13. Model-driven and Adaptable Interoperability Framework and Infrastructure</li> </ul>
Used in pilot	<ul style="list-style-type: none"> <li>• AIDIMA: eProcurement pilot</li> </ul>
Deliverable providing evaluation	D.A6.4 “Model-Driven and Adaptable Interoperability Infrastructure” (M24)

### 7.3.52. POP\* language interchange format and methodology

#### 7.3.52.1. Datasheet

<b>Solution data</b>	
Name	POP*
Result type	<ul style="list-style-type: none"> <li>• Metamodel</li> <li>• Exchange format</li> <li>• Methodology</li> </ul>
Description/functionality	The POP* language (stands for Process, Organisation, Products and other enterprise dimensions like Systems) defines a core set of enterprise issues to be defined in an enterprise model as a flexible intermediate language to facilitate model exchange between different enterprise modelling tools. The guideline for applying POP* enables companies to share knowledge in a structured way.
Benefits to interoperability	The major advantage of the POP* concept and the MPCE is the capability to keep models consistent even by using different modelling tools. So modelling elements which do not exist

	in one tool will be not destroyed to be used in a different tool. POP* had already influenced the work on ISO 19440.
Supported models/methodologies	-
Supported input interfaces	-
Supported output interfaces	-
Validation/demonstration	The INTRACOM PPM pilot.
Standards compliance	-
Availability	-
License	-
Status	POP* metamodel has been annexed to an ISO standard submission
Requirements/dependencies	-
Web references	-
<b>ATHENA metadata</b>	
Contact person	-
Contributors	-
Provided by project/activity	<ul style="list-style-type: none"> <li>A1 – Enterprise Modelling in the Context of Collaborative Enterprises</li> </ul>
Deliverables representing result	DA1.3.1 “Report on Methodology description and guidelines definition”
Contribution to key result	<ul style="list-style-type: none"> <li>9. Collaborative Enterprise Modelling Platform</li> </ul>
Used in pilot	INTRACOM: Product/project portfolio management
Deliverable providing evaluation	D.A1.6.1 “Benefit-Assessment” (M24)

### 7.3.53. RSM (Rational Software Modeler)

#### 7.3.53.1. Datasheet

<b>Solution data</b>	
Name	RSM (Rational Software Modeler)
Result type	Modelling tool

Description/functionality	Enables architects, systems analysts, designers and others to specify and communicate development project information from several perspectives and to various stakeholders.
Benefits to interoperability	-
Supported models/methodologies	UML 2.0
Supported input interfaces	XMI 2.0
Supported output interfaces	XMI 2.0
Validation/demonstration	-
Standards compliance	UML 2.0, XMI 2.0
Availability	Binary download
License	Commercial
Status	Commercial product
Requirements/dependencies	-
Web references	<ul style="list-style-type: none"> <li>Website: <a href="http://www-306.ibm.com/software/awdtools/modeler/swmodeler/index.html">http://www-306.ibm.com/software/awdtools/modeler/swmodeler/index.html</a></li> </ul>
<b>ATHENA metadata</b>	
Contact person	n/a
Contributors	n/a
Provided by project/activity	n/a
Deliverables representing result	n/a
Contribution to key result	n/a
Used in pilot	n/a
Deliverable providing evaluation	n/a

### 7.3.54. THEMIS

#### 7.3.54.1. Datasheet

<b>Solution data</b>	
Name	Semantic support for service descriptions
Result type	<ul style="list-style-type: none"> <li>Modelling tool</li> </ul>

Description/functionality	<p>The A5 project has addressed the development of extensions to existing standards for describing Web services. Such extensions included ACE-GIS at the beginning and PIM4SOA when it became available. A modelling tool based on the UMT software for supporting the modelling annotation of OWL-S information was implemented.</p> <p>Integration of the run-time A5 services platform with the semantic reconciliation infrastructure developed in A3 in order to provide a complete ATHENA run-time semantic mediation solution. This task has included also the investigation of existing semantic solutions the reconciliation support.</p>
Benefits to interoperability	<ul style="list-style-type: none"> <li>• Approach to richer modelling of services provided based on existing approaches</li> <li>• Solution provides integration with A3 and A5 results.</li> </ul>
Supported models/methodologies	-
Supported input interfaces	-
Supported output interfaces	-
Validation/demonstration	-
Standards compliance	-
Availability	-
License	-
Status	-
Requirements/dependencies	-
Web references	-
<b>ATHENA metadata</b>	
Contact person	Lorenzo Pondrelli, FORMULA
Contributors	FORMULA
Provided by project/activity	<ul style="list-style-type: none"> <li>• A5 – Planned and Customisable Service-Oriented Architectures</li> </ul>
Deliverables representing result	<ul style="list-style-type: none"> <li>• D.A5.2</li> <li>• D.A5.4</li> </ul>
Contribution to key result	<ul style="list-style-type: none"> <li>• 8: Interoperability Infrastructure</li> </ul>

	<ul style="list-style-type: none"> <li>13: Model-driven and Adaptable Interoperability Framework and Infrastructure</li> </ul>
Used in pilot	-
Deliverable providing evaluation	D.A5.5 "Validation of Research Results" (M24)

### 7.3.55. Semaphore - UML semantic mapping tool

#### 7.3.55.1. Datasheet

Solution data	
Name	Semaphore - UML semantic mapping tool
Result type	Mapping tool
Description/functionality	<ul style="list-style-type: none"> <li>Sempahore is a tool that aids in creating mappings or transformations between two different structural data formats at the Platform Independent level. The definitions of the actual data formats are reverse engineered into UML PIMs. The tool represents these models as two class diagrams on the screen. The user can then define mappings between the input and the output model grapically. Different types of mappings are supported such as copy, concatenate, split etc. The tool also has support for automatic matching between models, currently using name matching through string comparison.</li> <li>When completed the mapping defintion it self is viewed as a PIM. The mapping PIM is then transformed to code that performs the actual data transformation. For instance; if the input and output platforms are XML the generated data transformation code that is generated is XSLT.</li> <li>In summary this tool provides mapping capabilities using standard MDA technologies.</li> <li>Using Sempahore an architect or developer can create mappings between data structures at the Platform Independent level, and have the technical implementation of the actual data transformation code generated to suit the needed platform.</li> </ul>
Benefits to interoperability	Using Sempahore an architect or developer can create mappings between data structures at the Platform Independent level, and have the technical implementation of the actual data transformation code generated to suit the

	needed platform.
Supported models/methodologies	<ul style="list-style-type: none"> <li>• UML</li> <li>• XSD</li> </ul>
Supported input interfaces	<ul style="list-style-type: none"> <li>• UL</li> <li>• XSD</li> </ul>
Supported output interfaces	<ul style="list-style-type: none"> <li>• XSLT</li> </ul>
Validation/demonstration	Sempahore has been used to develop a mapping between a subset of the AIDIMA order format and the UBL order format.
Standards compliance	-
Availability	<ul style="list-style-type: none"> <li>• Binary download</li> </ul>
License	Eclipse Public License
Status	Prototype
Requirements/dependencies	Semaphore is developed as an Eclipse plugin, and needs Eclipse 3.1 to run. Detailed requirements for other plugins is provided in the installation guide.
Web references	<ul style="list-style-type: none"> <li>• Website: <a href="http://modelbased.net/semaphore/">http://modelbased.net/semaphore/</a></li> <li>• Eclipse update site: <a href="http://www.modelbased.net/semaphore/update/">http://www.modelbased.net/semaphore/update/</a></li> <li>• Installation guide: <a href="http://www.modelbased.net/semaphore/SemaphoreInstallGuide.pdf">http://www.modelbased.net/semaphore/SemaphoreInstallGuide.pdf</a></li> <li>• User guide: <a href="http://www.modelbased.net/semaphore/SemaphoreUserGuide.pdf">http://www.modelbased.net/semaphore/SemaphoreUserGuide.pdf</a></li> </ul>
<b>ATHENA metadata</b>	
Contact person	Andreas Limyr, SINTEF
Contributors	SINTEF
Provided by project/activity	<ul style="list-style-type: none"> <li>• A6 – Model-driven and Adaptive Interoperability Architectures</li> </ul>
Deliverables representing result	-
Contribution to key result	<ul style="list-style-type: none"> <li>• 13. Model-driven and Adaptable Interoperability Framework and Infrastructure</li> </ul>
Used in pilot	-
Deliverable providing evaluation	-

### 7.3.56. SOAP feedback analyser

#### 7.3.56.1. Datasheet

<b>Solution data</b>	
Name	SOAP feedback analyser
Result type	-
Description/functionality	-
Benefits to interoperability	-
Supported models/methodologies	-
Supported input interfaces	SOAP + WSDL in RDF-XML
Supported output interfaces	Web Interface
Validation/demonstration	-
Standards compliance	-
Availability	-
License	-
Status	-
Requirements/dependencies	-
Web references	-
<b>ATHENA metadata</b>	
Contact person	Julien Vayssiere, SAP
Contributors	-
Provided by project/activity	<ul style="list-style-type: none"> <li>A5 – Planned and Customisable Service-Oriented Architectures</li> </ul>
Deliverables representing result	-
Contribution to key result	<ul style="list-style-type: none"> <li>12. Service Composition Framework</li> </ul>
Used in pilot	-
Deliverable providing evaluation	-

### 7.3.57. THEMIS



### 7.3.57.1. Datasheet

<b>Solution data</b>	
Name	THEMIS
Result type	<ul style="list-style-type: none"> <li>• Software tool</li> </ul>
Description/functionality	<p>This result describes a repository and service for storing, managing and retrieving RDF schemas.</p> <p>A3 has chosen to use RDF and RDFS as main format for the resources that have to be reconciled. As the full A3 tool chain consisting of multiple tools (including ATHOS, A*, ARGOS, ARES) is involved in the reconciliation process and needs access to the RDF schemas A3 has chosen also to build its own repository for sharing RDF schemas. Each model is related to annotations, semantic rules and so on.</p> <p>Using a single repository enables reusing these relations between models and other resources, maintaining external references that can be used in the integration of the different software components. THEMIS not only has the role of a repository but it also provides a set of services used for the integration of the other A3 tools..</p>
Benefits to interoperability	The solution provides a common and shareable repository for the models used in A3; uniform access to RDF schema and a base for integration of the A3 results.
Supported models/methodologies	-
Supported input interfaces	Set of services for managing RDF and RDFS files
Supported output interfaces	Set of services for managing RDF and RDFS files
Validation/demonstration	<ul style="list-style-type: none"> <li>• AIAG: eKanban Pilot</li> <li>• AIDIMA: eProcurement Pilot</li> <li>• EADS: Change management process</li> <li>• CRF: Automotive Pilot</li> </ul>
Standards compliance	-
Availability	-
License	-
Status	-

Requirements/dependencies	-
Web references	-
<b>ATHENA metadata</b>	
Contact person	Lorenzo Pondrelli, FORMULA
Contributors	-
Provided by project/activity	<ul style="list-style-type: none"> <li>A3 – Knowledge Support and Semantic Mediation Solutions</li> </ul>
Deliverables representing result	D.A3.5: A reconciliation and mediation engine, capable to efficiently process semantic mediation and reconciliation rules (M24)
Contribution to key result	<ul style="list-style-type: none"> <li>11. Ontology-based Semantic Annotation and Reconciliation method/language/tool</li> </ul>
Used in pilot	<ul style="list-style-type: none"> <li>AIAG: eKanban Pilot</li> <li>AIDIMA: eProcurement Pilot</li> <li>EADS: Change management process</li> <li>CRF: Automotive Pilot</li> </ul>
Deliverable providing evaluation	-

### 7.3.58. WSDL Analyzer

#### 7.3.58.1. Datasheet

<b>Solution data</b>	
Name	WSDL Analyzer
Result type	Tool
Description/functionality	<p>A tool for detecting syntactical similarities between Web service descriptions.</p> <p>The WSDL Analyzer is a tool for detecting similarities between Web service descriptions (WSDL files). The tool can be used to find a list of similar services and produces a mapping between messages, thereby enabling brokering and mediation of services.</p> <p>A possible scenario for using the WSDL Analyzer is that the user already knows a service which provides the correct format. The WSDL of this service can be used as requirement for a similarity search. The WSDL Analyzer allows browsing the original WSDL and the candidate files.</p>

	<p>The algorithm detects common structures in port types, operations, messages and data type definitions. WordNet is integrated to improve the matching result. Mappings are assessed with a score which is used to establish a ranking between candidate service descriptions. Based on the similarities, a mapping is generated between two WSDL descriptions which can be used to transform SOAP messages exchanged between similar services at runtime. The result is a ranking of the candidates according to their matching score.</p> <p>The translation can be done automatically, if there is a one-to-one correspondence between elements. However, if several possible corresponding elements exist, translation requires intervention from a user in order to unambiguously transform parameters. The latter case shows the limitation of the structural approach. There are possible mismatches which can be detected with the help of the WSDL Analyzer, but not automatically corrected.</p>
Benefits to interoperability	<ul style="list-style-type: none"> <li>• The result supports the detection of inconsistencies between Web Services. If a one-to-one mapping between elements exists, the result supports automatic correction of the mismatch.</li> <li>• The result supports the IT Architect with resolving service mismatches in the case of system evolution.</li> <li>• The algorithm of the WSDL Analyzer improves over an existing algorithm for finding structural similarities taking into account additional features of the WSDL structure. More specifically, we make use of the tree-edit distance measure and the concept of a weak subsumption relation.</li> </ul>
Supported models/methodologies	WSDL
Supported input interfaces	-
Supported output interfaces	-
Validation/demonstration	<ul style="list-style-type: none"> <li>• EADS: Change management process</li> <li>• CRF: Automotive Pilot</li> </ul>
Standards compliance	<ul style="list-style-type: none"> <li>• <a href="http://www.w3.org/TR/wsdl">http://www.w3.org/TR/wsdl</a></li> </ul>
Availability	-
License	-

Status	-
Requirements/dependencies	-
Web references	<ul style="list-style-type: none"> <li>Installation and user guide:  <a href="http://athena.troux.com/Team/Repository/Projects/Project_223/Uploads/Installation_and_user_guide.pdf">http://athena.troux.com/Team/Repository/Projects/Project_223/Uploads/Installation_and_user_guide.pdf</a> </li> </ul>
<b>ATHENA metadata</b>	
Contact person	Klaus Fischer, DFKI
Contributors	DFKI
Provided by project/activity	<ul style="list-style-type: none"> <li>A5 – Planned and Customisable Service-Oriented Architectures</li> </ul>
Deliverables representing result	<ul style="list-style-type: none"> <li>D.A5.3: Architecture of SOA platforms (M21)</li> <li>D.A5.4: Execution Framework(s) for Planned and Customisable Service-Oriented Architectures (M21)</li> </ul>
Contribution to key result	<ul style="list-style-type: none"> <li>8: Interoperability Infrastructure</li> <li>12. Service Composition Framework</li> </ul>
Used in pilot	<ul style="list-style-type: none"> <li>EADS: Change management process</li> <li>CRF: Automotive Pilot</li> </ul>
Deliverable providing evaluation	D.A5.5 “Validation of Research Results” (M24)

## 7.4. Demonstrators

### 7.4.1. Overview

#### 7.4.1.1. Overview of demonstrators

##### A5 demonstrators

Demonstrator	Used tools	Integration with project	Description of integration
Discovery of services based on similarity of service description and subsequent runtime adaption of messages	WSDL Analyzer		
Automatic compliance check with Web Service specifications	Lyndon/Johnson		

Creation of logical services for use in Business Processes	Lyndon/Johnson	A2	The logical services are used by the Gabriel tool from A2
Automatic UI generation for Web Services	Lyndon/Johnson		
Transformation from PIM4SOA into PSM models for JACK	JACK	A6	The tool uses PIM4SOA
Transformation from ACE-GIS and PIM4SOA to OWL-S	UMT2OWLS	A6	The tool uses PIM4SOA
Johnson as semantic mediator	ARES and Johnson	A3	Johnson leverages ARES to provide semantic mediation

**A6 demonstrators**

Demonstrator	Leading partner	Used tools	Integration with project	Description of integration
Model-driven business process integration with PIM4SOA based on EADS scenario	IBM	PIM4SOA Tools (including Metamodels, Plugins, Web Service transformations, and Jack Transformation	<ul style="list-style-type: none"> <li>• A1</li> <li>• A2</li> <li>• A5</li> </ul>	<ul style="list-style-type: none"> <li>• mapping from POP* to PIM4SOA</li> <li>• mapping from MAESTRO and ARIS EPC to PIM4SOA</li> <li>• mapping to PSM webservice models</li> </ul>
P2P Business Resource Management Framework (CPD demonstrator)	SIEMENS	BRMF	B5	Preliminary version of CPD pilot
Jack EMF Model (preferably in conjunction with PIM4SOA)	DFKI	Jack Model and Transformations		
Semantic UML Mapping and Mediation (extended version of Athens demo)	SINTEF	Semaphore	A3	Could read PIM4SOA instances

MOF-based repository	SAP	MOF repository		
MDI Online help system	SINTEF	in conjunction with PIM4SOA tools	A4	
Portal: enterprise models for Collaboration space	Troux	POP*	A1 + A2	Use enterprise modeling to describe ATHENA solutions and how they interwork;