



SIMDAT

Data Grids for Process and Product Development using Numerical Simulation
and Knowledge Discovery

Project no.: 511438

Grid-based Systems for solving complex problems – IST Call 2
Integrated project



Deliverable

D.8.1.3 First Prototype of Knowledge Services Tools

Start date of project: 1 September 2004

Duration: 48 months

Due date of deliverable: 1 March 2006

Actual submission date: 13 April 2006

Lead contractor for this deliverable: FhG-AIS

Revision: 1.0

Project co-funded by the European Commission within the Sixth Framework Programme (2002-2006)		
Dissemination level		
PU	Public	X
PP	Restricted to other programme participant (including the Commission Services)	
RE	Restricted to a group specified by the consortium (including the Commission Services)	
CO	Confidential, only for members of the consortium (including the Commission Services)	

Copyright

Copyright © Fraunhofer-Gesellschaft (Institute AiS) and other members of the SIMDAT consortium, www.simdat.org, 2006

Table of contents

1	Introduction	5
2	State of the Art	7
2.1	Grid-Enabled Knowledge Services	7
2.1.1	Semantic Description and Discovery of Resources	7
2.1.2	Workflows	8
2.1.3	Distributed Data Mining	8
2.2	Relation SIMDAT – DataMiningGrid	10
2.2.1	Objectives	10
2.2.2	Synergies and Original SIMDAT Contribution	10
3	Work Provided to the Application Activities	11
3.1	Grid-Enabled Data Mining	12
3.1.1	Knowledge Services PM12 Prototype (grid-enabled Weka)	12
3.1.2	Automotive: Auto-1 PM24 Prototype	18
3.1.3	Aerospace: Cost-Modelling PM24 Prototype	24
3.1.4	Pharma	27
3.2	Knowledge-Based Directory/Registry Service	28
3.2.1	Pharma: Semantic Service Broker	28
3.2.2	Pharma: Searchable Registry of Bioinformatics Sequences Analysis Tool.....	29
3.2.3	Automotive: SimManager interface for semantic queries.....	30
3.2.4	Aerospace: interface for semantic queries	30
3.3	Response Surface Modelling.....	31
3.4	Semantic Annotation Tools	31
3.4.1	TUAM Annotation Tool.....	31
3.5	Workflow Construction Advisor	31
4	Implementation Plan	32
5	Future Work: Outline of the PM30 Prototypes	35
5.1	Workflow	35
5.2	Aerospace	35
5.3	Automotive	35
5.4	Pharma	35
6	Conclusion.....	35

Executive Summary

The purpose of this document is to describe the work done in the technology area of Knowledge Services in SIMDAT. The reporting period is PM13-PM18, but the document also contains a revised version of deliverable D8.1.1 (covering the period PM1-PM12), which was rejected at the PM12 review.

The document gives an overview of the state-of-the-art of Knowledge Services in a grid environment. The main focus is on the description of the key and 12 Knowledge Services prototype and its extension and integration into of the application areas:

- Distributed clustering in the Automotive Auto-1 prototype of PM24 and the resulting requirements. This section contains the revised version of the Automotive use case already described in deliverable D8.1.1.
- Cost-modelling in the Aerospace prototype of PM24 and the resulting requirements.
- Workflow mining based on graph-mining algorithms in the PM24 workflow construction advisor of Inforsense.
- Preliminary Pharma requirements to be fulfilled after PM24.

Knowledge Services in SIMDAT are of course more general than distributed data mining. We therefore also describe contributions by other partners in shorter sections. In particular, we mentioned here work done at the technology areas of Ontologies and Workflow. These sections are complemented by an outline of future work in the Knowledge Services area and an implementation plan.

1 Introduction

The work on Knowledge Services in SIMDAT addresses general issues of knowledge discovery in a grid-computing environment. This clearly exceeds the provision of grid-enabled or web-service enabled implementations of particular data mining algorithms. The goal of SIMDAT rather is to investigate how data mining can be used to gain new knowledge in the environment, where:

1. Many data mining tools (or algorithms) are available to perform different tasks
2. Different computational resources exist and can run these different algorithms

In this setting it is possible to select a data mining tool (algorithm) and submit it for execution on a remote computational source. However, the selection of the right tool/resource depends on

1. The application requirements.
2. The data being analysed.

The key to addressing these problems is based on the ability to develop methods, which exploit descriptions of the tools, resources, data, and applications effectively. The contribution of this work package within SIMDAT is

1. The development of grid-enabled data mining tools for distributed knowledge discovery
2. The development of meta data models for the description of data mining tools/algorithms, which are available
3. The development and use of workflow manipulation methods and workflow models, which allow the description of the application requirements (in collaboration with the workflow work package)
4. The use of ontologies (developed within the Ontologies work package for describing the data sources and data sets, which are available), for the effective access and integration of the required data. Ontologies are also used to obtain the match between the data and the right algorithms for analysis
5. The use of resource descriptions (developed within the grid infrastructure work package for describing and accessing resources) for the selection of computational resources, which match the algorithms and data.

The Knowledge Services work package focuses on the discovery of services through semantic descriptions, i.e. using ontologies for describing and finding data, algorithms, tools and application requirements. The Knowledge Services work package is evolving along the following action lines:

- Meta data: Current data mining applications vary greatly in their complexity, number and meaning of parameters, resource requirements, dependencies on third party products, and scope.
For these reasons, we strongly believe that developing specialised solutions for individual data mining applications is not the right approach. Instead we favour generic solutions, which are based on meta-data descriptions of the actual applications and corresponding client and server-side components to process them.
We are developing a generic meta data schema, describing resource requirements, types of algorithm parameters, and data features. A meta data schema allows to

develop generic components of data mining for SIMDAT applications. Furthermore, the meta data schema allows to locate the data mining components via a central registry.

- Workflow: currently workflows are used for coordinating different resources, i.e. accessing algorithms and submit them for execution. Hardly any of the ongoing grid-projects related to data mining and knowledge discovery are using a workflow editor to coordinate data mining tasks and to hide the inherent complexity of grid technology from end-users, which are often data miners with only little knowledge about grid technology, if any. With the help from the workflow technology area Knowledge Services integrate the generic data mining and knowledge management solutions into an easy-to-use workflow editor that can be operated with little or even without any knowledge about grid technology. Later workflow mining will be used to assess application requirements.
- Resources: in SIMDAT, most applications use GRIA to control resources and algorithm execution. We extended GRIA to allow code mobility, i.e. code of data mining components is sent to the resource, which does not need to have the component pre-installed for execution.
- Data: the application areas of SIMDAT need to access simulation databases, etc to perform knowledge discovery. The key element in this scenario is the development of a data semantics schema. We will investigate how ontologies can be used to describe the input/output requirements for the different algorithms and data sets.

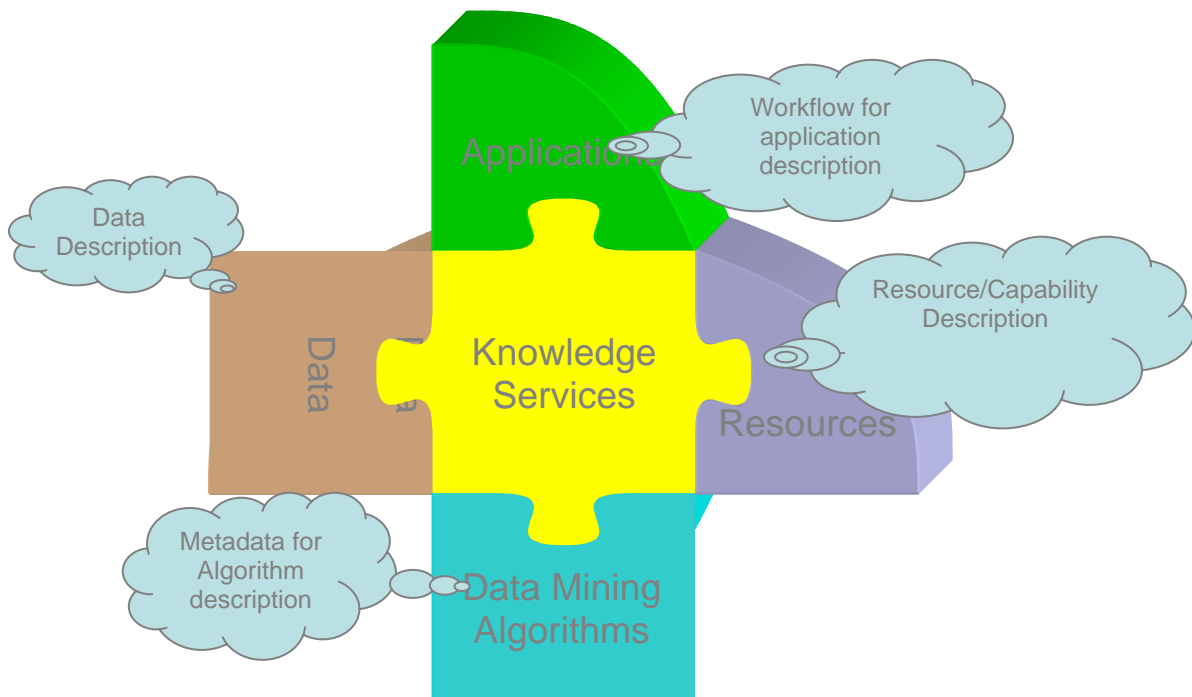


Figure 1: Knowledge Services in Simdat

This deliverable is the third document describing progress in work package 8.1. Its title has been devised before it was agreed to have operable versions of the SIMDAT prototypes already at PM12. The title therefore is somewhat misleading, because now the first Knowledge Services prototype is described already in D8.1.2. The present focus of D8.1.3 is on evaluation, updated requirements, and the extended prototype design.

The PM12 review rejected deliverable D8.1.1 because it did not clearly identify and comment on the potential overlaps between SIMDAT and DataMiningGrid, the two projects in which the FhG-AIS is developing different versions of a grid-enabled Weka toolbox. Moreover, a revision of the Pharma use case was requested by the reviewers.

From our point of view, there are strong reasons to include the revised version of D8.1.1 into the present deliverable D8.1.3. A major topic of D8.1.3 is also the description of requirements from the application areas. For reasons explained earlier in the deliverables and progress reports, it was not possible to collect detailed requirements from the application areas for a deliverable D8.1.1. In particular the Pharma use case only was a proposal created by FhG-AIS. Now we are in a better situation, where requirements have been collected from all three application areas. It is therefore expedient not to cite the former Pharma requirements again, but to replace them by the new requirements. As for the Automotive requirements, they have also been made considerably more precise at this point.

In particular parts from deliverable D8.1.1 have been integrated into this deliverable in the following sections:

- Description of the Automotive Auto-1 prototype (section 3.1.2).
- Description of the Pharma requirements (sections 3.1.4.2.1 and 3.2.2.1.1).

The reviewers requested also an explanation of potential overlaps and synergies between the two projects Simdat and DataMiningGrid with respect to the development of grid-enabled Weka tools. This explanation is given in section 2.2

2 State of the Art

2.1 Grid-Enabled Knowledge Services

In the context of grid services, knowledge-based information plays a central role in the following areas.

2.1.1 Semantic Description and Discovery of Resources

It is not a trivial to locate resources on the grid. What is needed to find resources is an exhaustive description of everything the user needs to execute his application. In particular, properties of the data, the algorithms (data mining applications, knowledge-based analysis applications, etc), and the required computing resources, must be described. Then the user requirements have to be matched against resources which are available. In general the matching will be performed in a central registry. Ontologies and formal means of matching requirements play a central role here. Several FP6 projects are actively developing solutions in this area.

- **Ontogrid**¹ develops a prototype of the next generation semantic grid for rapid prototyping and knowledge intensive distributed open services. Ontologies are a central building block here.
- **Akogrimo**² develops mobile grids. A central topic is the dynamic creation of virtual organisations, which is only possible through the evaluation of elaborated resource descriptions.
- **Inteligrid**³ develops the grid based integration and interoperability infrastructure. This project applies intelligent ontologies services to bridge the gap between technical grid

¹ <http://www.ontogrid.net/>

² <http://www.mobilegrids.org/>

³ <http://www.inteligrid.com/>

concepts, and engineering domain concepts. The technology used is based on the Ontogrid recommendations for semantic grid services.

- **Provenance**⁴ deals with the design and implementation of an open Provenance architecture for grid systems. We will possibly use parts of this architecture to enhance our own data and algorithm descriptions.
- **DataMiningGrid**⁵ is the second project, in which the FhG-AIS participates. It also use as a central registry for the detection of resources. Details are described in the next sections.

2.1.2 Workflows

Workflows represent the user's knowledge about data processing, and knowledge discovery. They are therefore a central element in grid computing. It is worthwhile to treat workflows as knowledge representations, which can be stored in the same way as other grid resources, described, matched, retrieved and used. Workflow management therefore also is central to several FP6 projects:

- **K-Wf Grid**⁶ is most prominent with respect to this topic, because it develops a knowledge-based workflow system (expert system) for grid applications.
- **Akogrimo**, also develops a workflow manager, which operates on a central workflow repository. The workflows are stored using workflow templates.
- **DataMiningGrid** uses a workflow manager to control the data mining applications.

2.1.3 Distributed Data Mining

Distributed knowledge discovery is a central topic of SIMDAT, and distributed data mining can be subsumed under this heading. This is the central contribution of FhG-AIS to SIMDAT.

Currently, several ongoing projects aiming to couple data mining solutions with grid computing exist. With three ongoing or already finished projects in this field the data mining toolkit "Weka"⁷ is a very prominent example for the different approaches taken. All projects aim at a tight integration of grid technology (e.g. WSRF/GT4) into Weka itself rather than developing generic solutions for performing data mining in grids in general. All projects integrate grid technology into Weka's GUI itself or require users to start it from the command line rather than providing a high-level elaborated interface to grid-enabled data mining.

- **Weka4WS**⁸ is a data mining toolkit developed at the University of Calabria, which uses part if the functionality provided by the Globus Toolkit 4 (GT4). Weka4WS distinguishes between user, computing and storage nodes. While the user nodes provide the standard Weka user interface (Explorer-Panel) into which all grid functionality is integrated, computing nodes are host WSRF-compliant Web-services, which represent a subset of the data mining algorithms from the original Weka. Weka4WS requires that the respective services are pre-installed on every computing node. As transfer of file based data is realised via GridFTP, the storage nodes must run a GridFTP server. When an analysis is started the respective services deployed on the computing nodes access the files on the storage nodes and read in the required input data.

⁴ <http://www.gridprovenance.org/>

⁵ <http://www.datamininggrid.org/>

⁶ <http://www.kwfgrid.net/>

⁷ <http://www.cs.waikato.ac.nz/ml/weka/>

⁸ <http://Grid.deis.unical.it/weka4ws/>

While this approach is feasible for a quick and tight integration of a single data mining toolkit into a grid environment, it bears the following crucial disadvantages:

- The middleware employed (GT4) is incompatible to the one widely used in SIMDAT (GRIA) regarding applied standards (WSRF vs. WS-I), security (proxy certificates vs. no proxy certificates) and data transport (GridFTP vs. no GridFTP). This probably largely results from the different focuses of both middleware packages. While GT4 focuses mostly on the scientific community, GRIA has been developed from the start to enable inter-enterprise computing such as B2B scenarios.
- As the name already suggests, Weka4WS is based on WSRF-compliant Web services, meaning that its algorithms are wrapped inside these services. However, as a result these services have to be deployed on all computing nodes prior to starting an analysis and cannot be dynamically transferred and started on a different node. This also means that it is impossible to start an analysis on a cluster controlled by specialized clusterware such as PBS or Condor.
- As GT4 does not contain any resource scheduler for service invocation and the Weka-services cannot be deployed at runtime on a different machine, compute nodes are prone to “stall” in such cases when many users simultaneously start analyses on the same server.

Summarizing, the approach of developing service-wrappers and the incompatibility of GT4 with GRIA renders Weka4WS unusable in the context of SIMDAT.

- **Weka-Parallel**⁹, available at SourceForge.net, does not claim to be grid-enabled but rather implements distributed cross-validation for all algorithms implemented in Weka relying on Java Remote Method Invocation. Therefore, despite its name, Weka-Parallel is limited to cross-validation only, although significant performance gains have been reported¹⁰. Furthermore, it is only applicable when all calculations are done in a multi-processor environment without any firewalls such as a cluster. Nonetheless, as cross-validation can take very long, depending on the size of the data, the algorithm and the number of folds, this project may be of interest in later stages of the SIMDAT project for implementing use cases in which Weka is accessed through an application service provider.
- In **Grid-Weka**¹¹, which was developed in University College Dublin, execution of Weka tasks can be distributed across several computers in an ad-hoc “grid”. This work is similar to the Weka-Parallel project, but allows for performing more functions in parallel and/or on remote machines. The Weka function is distributed by partitioning the data set, processing several partitions in parallel on different available machines, and merging the results into a single resulting data set. Grid-Weka uses proprietary protocols and relies on a client-server architecture using Java object serialisation for data exchange. Therefore, Grid-Weka is not capable to perform distributed data mining across organizational boundaries, which are usually protected by firewalls. Similar to Weka4WS the application has to be installed on every computer designated to run it. Grid-Weka also provides a mechanism for load balancing. While this is a feasible solution in an environment where no middleware is present, its load balancing would interfere with other schedulers, which are often part of standard clusterware, such as Condor. An efficient environment for Grid-Weka

⁹ <http://weka-parallel.sourceforge.net/>

¹⁰ <http://weka-parallel.sourceforge.net/report.pdf>

¹¹ <http://smi.ucd.ie/~rinat/weka/>

would therefore require machines that are exclusively designated to running this package besides those running Condor for general scheduling.

We strongly feel that especially the absence of any standards related to grid technology or Web services and the incompatibility to established clusterware packages renders this Grid-Weka unusable for the purpose of SIMDAT at this stage.

- **DataMiningGrid** also features the development of distributed data mining tools, also developed in this context by FhG-AIS. See also the separate discussion of this project in section 2.2 below.

2.2 Relation SIMDAT – DataMiningGrid

2.2.1 Objectives

The SIMDAT objective is the development of generic grid technology for the solution of complex application problems, and the use of this new technology in several industrial application sectors. In this context, the goal of SIMDAT is to develop federated versions of problem-solving environments. Data mining provides a central component for these problem-solving environments.

The objective of DataMiningGrid is the development of generic and sector-independent data mining tools and services for the grid. The project will implement a test bed consisting of several prototypical data mining applications from a diverse set of sectors.

2.2.2 Synergies and Original SIMDAT Contribution

The philosophy in SIMDAT as well as in DataMiningGrid regarding the integration of data mining applications/algorithms in a grid environment is to avoid implementing specialized services and client-side components for each algorithm, if possible. In DataMiningGrid this has led to the development of a searchable XML-based meta-data schema, which describes each single algorithm regarding its options, inputs, outputs, resource requirements and additional information (vendor, version, etc.). This information can be used by generic client-side components (e.g. a workflow editor) for providing appropriate graphical means for specifying user settings and additional help for specifying resource requirements. This is necessary because of the heterogeneity of algorithms in this field. Algorithms vary a lot regarding their parameters (number, types, and effects), inputs, outputs and resource requirements. Even data mining suites (e.g. Weka) usually represent only a collection of independent data mining algorithms with a unified graphical user interface and underlying data structure. While the initial work done in DataMiningGrid is also to some extent feasible for the purpose of SIMDAT, this meta-data schema will be extended during the remaining time of SIMDAT and may result in an ontology for such algorithms, which also describes dependencies between algorithms and their inputs and outputs. Additional work in the SIMDAT project is also required to adjust the schema in order to reflect the differences between the different grid middleware packages used in both projects (GRIA in SIMDAT and GT4 in DataMiningGrid) regarding storage of meta-information, specification of resource requirements and for the client-side components, which have to be modified to reflect the ways of starting a grid job in GRIA and GT4.

Weka is used in SIMDAT as well as in the DataMiningGrid project for demonstrating the integration of a highly capable and recognised data mining application into a grid environment. However, the actual functionality of Weka is different from the one used in SIMDAT. In DataMiningGrid Weka is not in the centre of work, but is used without any changes to its code, while in SIMDAT also a partly distributed version of Weka was developed (parallel clustering), which is capable of processing data that is stored at different

servers in different organisations. Still, the meta-data descriptions of individual algorithms of Weka will be used in both projects with modifications of the underlying schema explained above.

3 Work Provided to the Application Activities

Knowledge discovery in a grid computing environment is that the centre of the SIMDAT project. As already stated in the introduction, knowledge discovery comprises more than only executing data mining algorithms on the grid. The goal is a seamless integration of knowledge discovery into the industrial production of environments. To achieve it, it is necessary to provide, among others, knowledge-based query and retrieval facilities, to describe and locate data, algorithms, and semantic informations. In the first 18 months of SIMDAT, the three application activities of Aerospace, Automotive, and Pharma received contributions from Knowledge Services, which are steps in this direction. We have grouped them into five sections, whose topics will now be explained (see Table 1 **Fehler! Verweisquelle konnte nicht gefunden werden.**).

	Aerospace	Automotive	Pharma	Workflow
Grid-enabled Data Mining	cost modelling (AIS): adjustment of requirements	grid-enabled Weka (AIS), distributed clustering (AIS): adjustment of requirements	knowledge-enhanced Pharma portal (AIS): adjustment of requirements	grid-enabled Weka (AIS)
Knowledge-based Directory/Registry Service	Interface for semantic queries (Ontoprise): adjustment of requirements	SimManager Interface for semantic queries (Ontoprise)	Semantic Service broker (NEC), Registry of Sequence Analysis tools (AIS): Adjustment of requirements	
Response Surface Modelling	Adjustment of requirements (AIS)			
Semantic Annotation tools			TUAM mapping tool (SCAI)	
Workflow Construction Advisor				workflow mining (AIS): adjustment of requirements

Table 1: work provided to the application activities until PM18

- **Grid-enabled data mining** provides algorithms to solve particular tasks of knowledge discovery in a grid environment.

-
- **Knowledge-based directory/registry services** enable users to locate the resources they need by evaluating their semantic queries about data and algorithms.
 - **Response surface modelling** provides an integrated high-level, and possibly dynamic representation, or fusion of low-level simulation results from several sources.
 - **Semantic annotation tools** can be used to store abstract knowledge as semantic information is in-place together with numerical or other data. Semantic annotation is a prerequisite for the use of knowledge-based registry services.
 - **Workflow construction advisors** assist users in constructing new workflows by proposing next steps, parameters, and indicating workflows similar to the one currently being assembled. This task can only be solved by workflow mining tools, which heavily rely on the evaluation of knowledge stored in the workflows.

3.1 Grid-Enabled Data Mining

The most prominent activity of the reporting period was the development of the Knowledge Services prototype, featuring grid-enabled Weka. We therefore start with the description of this activity.

3.1.1 Knowledge Services PM12 Prototype (grid-enabled Weka)

3.1.1.1 Description of the Prototype

For the reasons discussed elaborately in deliverable D8.1.2 the Knowledge Services prototype is implemented as a stand-alone version by FhG-AIS in cooperation with InforSense. It is intended to demonstrate how in a grid environment data mining algorithms can be applied to data sets, which are distributed on servers of several different organisations. For this purpose 49 different standard data mining algorithms from the data mining toolkit Weka are made accessible via the GRIA grid middleware. Additionally the k-means clustering algorithm from Weka is modified to process geographically dispersed data sets located at sites of different GRIA providers. Figure 2 illustrates the architecture of this prototype. All algorithms can be executed at different GRIA providers (Provider B in the figure) without prior installation of Weka. This allows mining data close to the data storage instead of transferring possible large data sets to the application. For designing and executing data mining tasks the workflow editor provided by InforSense is used.

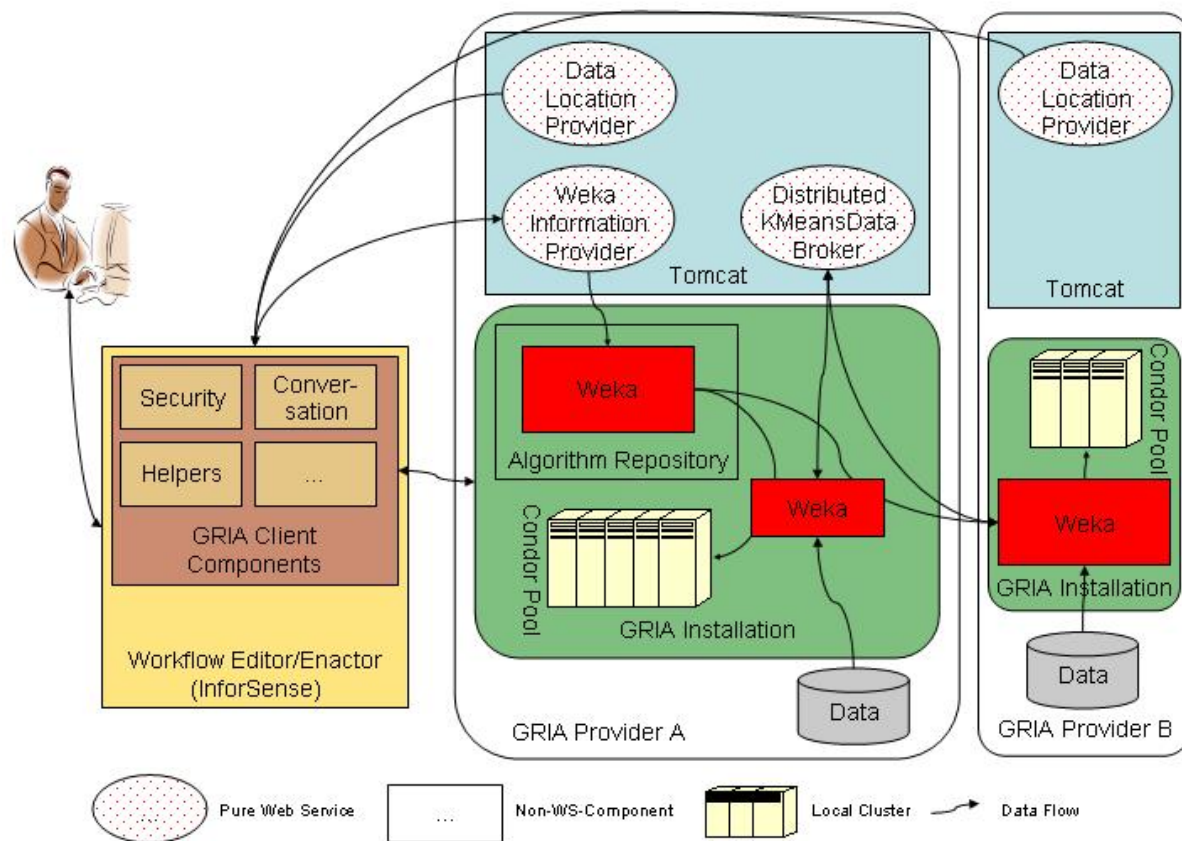


Figure 2: Architecture of the PM12 Knowledge Services prototype including data flow between services and components

As partly illustrated in the figure the following SIMDAT modules are used:

- Workflow: The workflow editor and enactor are provided by InforSense.
 - The Execution Management:
 - Resource Management:
 - Monitoring and Information Services:
 - Security:
 - Grid Service Core:
- } These modules are all part of the GRIA Grid middleware.

Although implemented as a stand-alone piece of software, the PM12 Knowledge Services prototype is already developed with the Automotive: Auto-1 PM24 Prototype in mind and is supposed to be integrated into this upcoming prototype without substantial changes (see also section 3.1.2). However, as it was stand-alone we did not receive any feedback from the respective partners in the Automotive activity, but tested it internally regarding user-friendliness and technological soundness.

3.1.1.2 Lessons learned

As the PM12 prototype was not integrated into any of the prototypes delivered by the application activities, this prototype was tested internally at FhG-AIS. The chosen users were experts on data mining but only had little knowledge about grid technology. The internal test revealed that the PM12 prototype was well accepted regarding its functionality, but also that it lacked user-friendliness even though it is operated through the workflow editor as depicted in Figure 3. Three main reasons for this perception were discovered:

1. Users had to know about the sequence in which certain individual nodes of the workflow had to be executed due to lack of coordination by the workflow editor.
2. Users had to know the application provider offering the application to execute including the exact URI due to a missing service registry.
3. Users had to allocate sufficient resources for the task at hand without any help from the system. As a result data miners were required to familiarise themselves with details of the internals of the prototype. This also led users to frequently allocate too many resources (e.g. disk space) in order for them to be sure that the job will not fail.

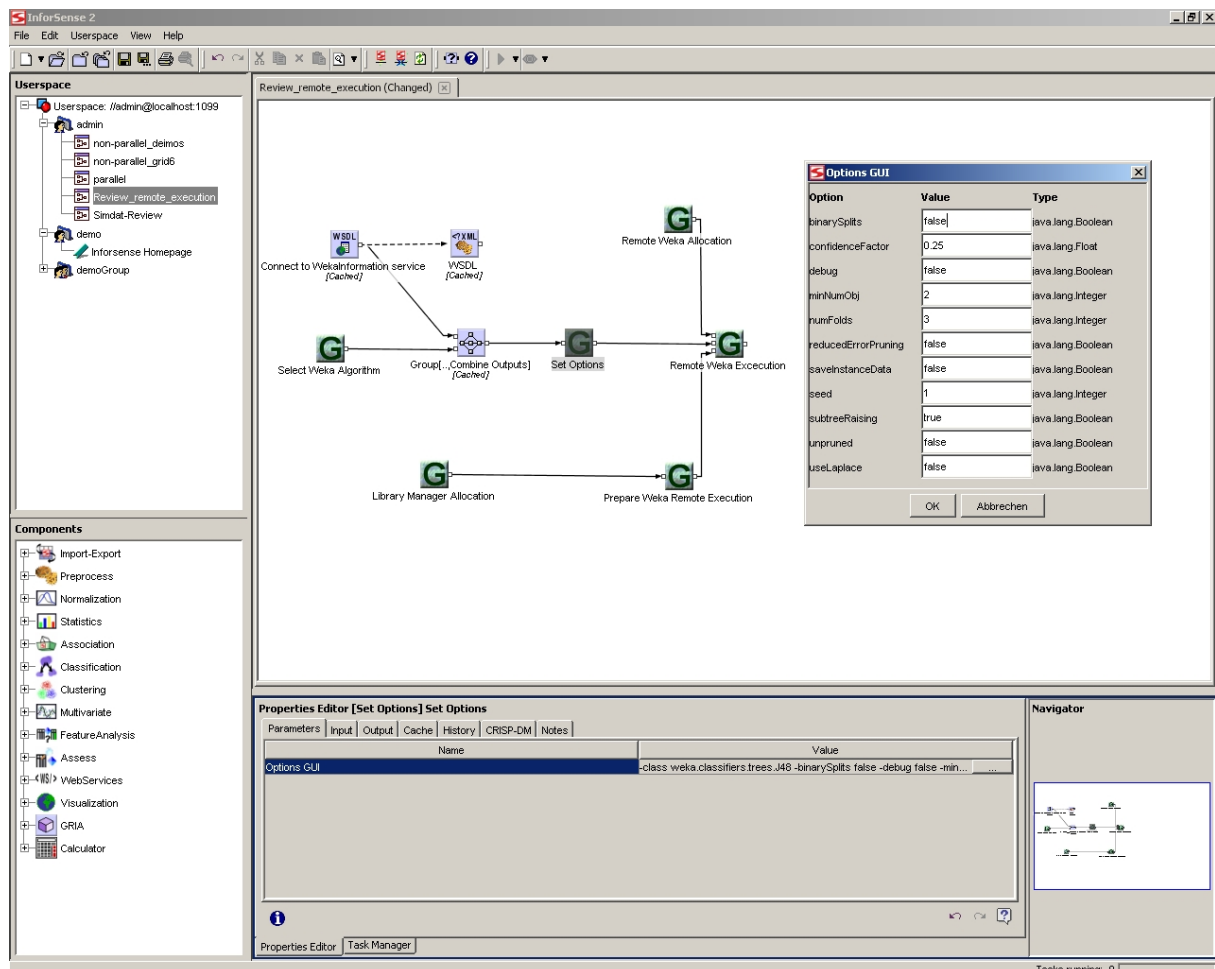


Figure 3: Workflow of the PM12 prototype for distributed data mining

At the technical side GRIA proved to be a reliable grid middleware package, providing much help regarding setup and administration. Also integrating applications into GRIA proved to be feasible without much effort.

On the other hand implementing a distributed version of k-means clustering proved to be much more challenging than expected. While communication inside applications running in parallel at different machines can normally be established by traditional means, such as Java RMI, MPI¹² or simple sockets, this is not possible when crossing administrative boundaries that are protected by firewalls. For bridging firewalls a broker service was implemented, which coordinated the parallel instances of the clusterer. The resulting complexity from using

¹² The MPI (Message Passing Interface) is a library specification for message-passing in distributed systems, proposed as a standard by a broadly based committee of vendors, implementers, and users (<http://www-unix.mcs.anl.gov/mpi/>).

Web services, e.g. transmitting very complex data types, session management, notifications, were clearly underestimated. Furthermore, the solution is only applicable to the specific algorithm (k-means) and cannot be re-used for other parallel applications.

As a result FhG-AIS has decided to concentrate on the development of an easy-to-use framework for integrating a large number of data mining applications in a user-friendly way as described before, instead of implementing other parallel applications.

3.1.1.3 Detailed Description of Requirements Resulting from PM12 Prototype

The feedback from selected end users revealed that the workflow editor as the user interface of PM24 prototype has to be much more self-explaining and has to provide as much help as possible, e.g. for searching for data mining application by certain keywords and by providing some help for resource allocation. As a result the following requirements, which apply to the Automotive: Auto-1 PM24 Prototype and the Aerospace: Cost-Modelling PM24 Prototype discussed later, have a strong focus on these aspects.

3.1.1.3.1 One-click Execution of Workflows

One-click execution of workflows is required to disburden users of the workflow editor from invoking parts of the workflow in a predefined order, which is not visible to users from the workflow itself. Instead any workflow has to be executable with a single “execute” command (e.g. on the last node).

Name	One-click execution of workflows		
Application Activity	Aerospace, Automotive		
Prototype(s)	Aerospace, Automotive 1		
Date Created	03-13-2006	Priority	Medium
Created By	Jörg Kindermann	Technology component	Workflow
First Implementation Date		SIMDAT module targeted	Workflow
Description	“One-click execution” of workflows designates a mechanism that enables users to start execution of a workflow with a single command to the workflow editor.		
Relation to prototype	<ul style="list-style-type: none"> User support in the PM12 prototype was not optimal. The user had to manually activate several workflow nodes, and there were certain restrictions in the activation sequence of nodes. 		
Requested functionality	<ul style="list-style-type: none"> The "one-click execution" should enable the user to start workflow execution without manually guiding the right sequence 		
Validation	<ul style="list-style-type: none"> Workflows for the different prototypes must be executable with one “execute” command only. 		

3.1.1.3.2 Meta-data Schema for Describing Data Mining Applications

Data mining applications often consist of a number of individual algorithms that may be executed independent from each other and vary greatly regarding purpose, complexity, number and types of parameters, resource requirements and the like. Instead of developing

individual solutions for each application or algorithm, and providing additional help to end-users on the basis of these specialised solutions, a generic meta-data schema, suitable for all data mining applications, is required that contains as much information as possible, which can be displayed to end-users (e.g. general description, parameter types, description of parameters, possible values, resource requirements, etc.).

Name	Meta-data schema for describing data mining applications		
Application Activity	Aerospace, Automotive		
Prototype(s)	Aerospace, Automotive 1		
Date Created	03-13-2006	Priority	High
Created By	Jörg Kindermann	Technology component	Knowledge Services
First Implementation Date		SIMDAT module targeted	Workflow, Information Services
Description	Data mining applications vary greatly in their purpose, options, inputs, outputs and resource requirements and constraints. In order to help users in applying these applications in a grid environment a meta-data schema for describing data mining applications based on a fixed XML schema is required.		
Relation to prototype			
	<ul style="list-style-type: none"> Users should be able to track down in data mining applications which fit their needs on the grid. The data mining applications must therefore be describable in a general schema, including their capabilities, and possible parameters. This schema must also be used to enact the data mining applications by means of generic workflow components (see below). 		
Requested functionality			
	<ul style="list-style-type: none"> The schema should allow describing all relevant aspects of data mining applications The schema should allow describing some of the resource requirements/constraints of data mining applications (see below) 		
Validation			
	<ul style="list-style-type: none"> The schema will be tested with many different data mining applications, including Weka and text mining applications developed at FhG-AIS. 		

3.1.1.3.3 Generic Workflow Components for Processing Data Mining Application Descriptions

For processing and displaying meta-data of individual data mining applications appropriate client-side components are required, which have to be integrated into the workflow editor. This includes a generic panel, which dynamically adjusts to the selected application regarding its parameters.

Name	Generic workflow components for processing data mining application descriptions		
Application Activity	Aerospace, Automotive		
Prototype(s)	Aerospace, Automotive 1		
Date Created	03-13-2006	Priority	high
Created By	Jörg Kindermann	Technology component	Workflow
		SIMDAT module targeted	Workflow, Information Services
Description			

Implementation of generic workflow components (nodes) for processing and visualizing data mining application descriptions on the client-side.			
Relation to prototype			
<ul style="list-style-type: none"> A generic workflow component should be able to parse the data mining application description schema and present its content (general descriptions, options, input & outputs, etc.) in an appropriate way. 			
Requested functionality			
<ul style="list-style-type: none"> Implement generic components for: <ul style="list-style-type: none"> I. Visualization/user specification of an application's description, options, input & output II. Execution of the respective application according to the user's inputs 			
Validation			
<ul style="list-style-type: none"> The workflow components will be tested with meta-data from the different data mining applications. 			

3.1.1.3.4 User Support for Resource Requirements Declaration and Resource Allocation

As the meta-data description of each application also includes information about its resource requirements, corresponding components have to be integrated into the workflow editor, which display this information and provide some help to end-users for allocating appropriately dimensioned resources.

Name	User support for resource requirements declaration and resource allocation		
Application Activity	Aerospace, Automotive		
Prototype(s)	Aerospace, Automotive 1		
Date Created	03-13-2006	Priority	Medium
Created By	Jörg Kindermann	Technology component	Workflow
First Implementation Date		SIMDAT module targeted	Workflow, Information Services
Description	User support for resource requirements declaration and resource allocation		
Relation to prototype			
<ul style="list-style-type: none"> User support in the PM12 prototype was not yet optimal. The user had to manually declare resource requirements, request resource offers, and select from the offers, which is often a difficult task given the great variety of data mining applications. 			
Requested functionality			
<ul style="list-style-type: none"> As far as specified in the application's description, required and optional resources should be explicitly marked. The user should be guided through resource allocation step-by-step by the workflow editor. 			
Validation			
<ul style="list-style-type: none"> This functionality will be validated against meta-data from the different data mining applications. 			

3.1.1.3.5 Searchable Registry for the Descriptions of Data Mining Applications

As data mining has gained a lot of momentum in the past decade, the number of available data mining applications has grown rapidly and is still growing. Therefore, it is important for data miners who intend to use grid technology, to be able to search for specific applications with respect to the following information:

- The identity of the grid provider
- The type of information that is extracted (e.g., predictive models, association patterns, cause-effect relationships, detection of affinity similarity-based groupings, deviation detection)
- The format of the induced information (e.g. rules, decision trees, correlation networks, association patterns, neural networks, matrices, visualization),
- The type of data the application operates on (e.g., digital images, text, discrete, continuous, sequence, temporal)
- The domain for which the application is developed (e.g., finance, engineering, science, life science, manufacturing, marketing)

Name	Searchable registry for the descriptions of data mining applications		
Application Activity	Aerospace, Automotive		
Prototype(s)	Aerospace, Automotive 1		
Date Created	03-13-2006	Priority	high
Created By	Jörg Kindermann	Technology component	Workflow, Ontologies
First Implementation Date		SIMDAT module targeted	Information Services
Description	Searchable grid-wide registry for the description of data mining applications		
Relation to prototype	<ul style="list-style-type: none"> • The data mining application description schema must be stored in a grid-wide repository or registry, in which the users can search for applications which meet their needs 		
Requested functionality	<ul style="list-style-type: none"> • Store XML-based data mining application descriptions • Match descriptions against user requirements (query) 		
Validation	<ul style="list-style-type: none"> • The selected registry must be capable of processing complex user-defined queries. 		

3.1.2 Automotive: Auto-1 PM24 Prototype

3.1.2.1 Outline of the PM12 Use Case

This section is cited from deliverable D8.1.1, because it is still applicable to the Automotive scenario.

Due to the various departments involved in vehicle development, NVH data generally need to be stored in a distributed manner. Data integration forbids itself for several reasons: the data volume is very large, and, more importantly, the data may be proprietary. However, metadata can be computed locally, which also considerably reduces its size. Still the amount of data may be too large to integrate, depending on the particular data mining task. Therefore a grid-enabled version of WEKA is needed. In this way distributed WEKA algorithms can be sent to the data sources and the achieved results can be integrated.

Weka is a widely known and accepted data mining application, especially in academic and research areas, containing many state of the art data mining algorithms. With Weka being able to execute in grid environments users can choose from a rich set of services to perform various data mining tasks. These services range from algorithms for pre-processing and visualization to actual data mining algorithms including decision trees, clustering, association rules, support vector machines, and meta-learners. Although implemented as a single-user, stand-alone Java application, developers are able to modify Weka in a relatively easy manner thanks to its well-defined interfaces and modular concept that makes heavy use of Java Beans. Such modifications include addition of service wrappers, parallelization of algorithms, and decoupling of components to reduce the total number of classes to be transferred to remote machines for execution. However, although Weka is fairly easy to modify, modifying the application to use the Java Data Mining API is currently not planned for this prototype due to the tight schedule and the extent of the modifications necessary.

The scenario involves automotive engineers as typical users. They have specific in-depth knowledge on the data being processed, but only limited knowledge with respect to the data mining algorithms they want to use. They mainly work on-line, which requires short to moderate system response times.

Objective	Description
Knowledge Discovery to identify car components which are important for NVH analysis	<ul style="list-style-type: none"> • A CAE engineer wants to perform a Knowledge Discovery experiment in the CAE-BENCH Environment. • As a result of a query he receives a subset of simulation results for his analysis. This may include results from all federated simulation environments. • The meta data is extracted locally. • The available metadata entries from the simulation results are listed and the engineer selects an appropriate subset. • As a first step in the analysis (or for smaller data mining queries) a subset of the data is extracted and sent back to the environment, from which this action was started. The data objects are aggregated and a related matrix with meta data information is written to a specific file. Outside of the environment the WEKA tool is used to analyse the extracted data elements. • As a second step the WEKA tool is sent to all those grid services that have been involved in the meta data processing. WEKA is then run locally. The analysis results are sent back to the CAE engineer. Depending on the search capabilities to be integrated into CAE bench during SIMDAT further pre-selection may prove necessary. Here attribute selection algorithms will be particularly relevant. • A distributed cluster algorithm is developed and integrated in the distributed WEKA framework in order to allow for cluster analysis on large data volumes with spatial attributes (not part of the 12-month prototype).

3.1.2.2 Outline and Purpose of the PM24 Prototype

In the Automotive activity the PM24 prototype will progress further towards integrating functionality resulting from Knowledge Services into the Auto-1 prototype. The knowledge use case of the PM 24 Automotive knowledge prototype is to investigate the impact of design changes on the NVH behaviour of the SAMD prototype.

3.1.2.3 Initial Specification of the Prototype and its Use Case

As described in deliverables D7.1.3-7.1.4 and D11.1.4, FhG-SCAI and MSC will implement a client application/portal to extract meta-data from a SimManager system, which holds the simulation results and the so-called input deck used for the simulation. The resulting meta-data is required as input for the actual data mining step performed by FhG-AIS as depicted in Figure 4.

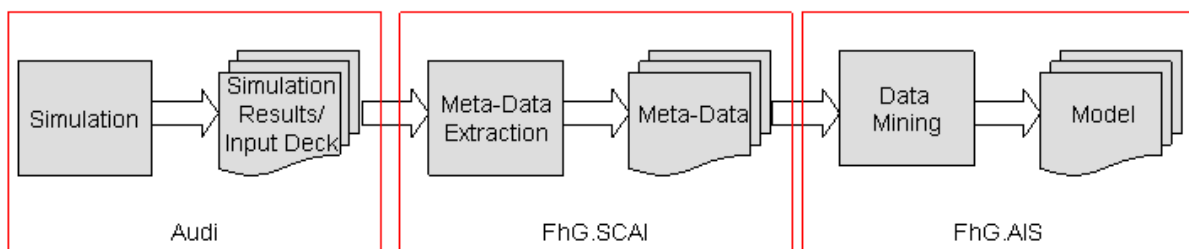


Figure 4: Separation of work for data mining in the Auto-1 prototype

The Weka data mining tool depicted at the right in Figure 5 represents basically the PM12 Knowledge Services prototype, which was shown at the review in November as a stand-alone version. The modules required for the data mining step consist of the following:

- **Workflow editor provided by InforSense:** InforSense's workflow editor will be used for performing the data mining step.
- **Registry for available data mining applications:** The registry contains descriptions of all available data mining applications. It allows users to search for specific applications and, together with the components integrated into the workflow editor, also provides some help to users on how to operate the selected application.
- **A data repository:** The data repository contains the meta-information in the Weka format .ARFF.
- **Distributed Weka data mining toolkit:** This module is already available and may be used virtually without any modifications.

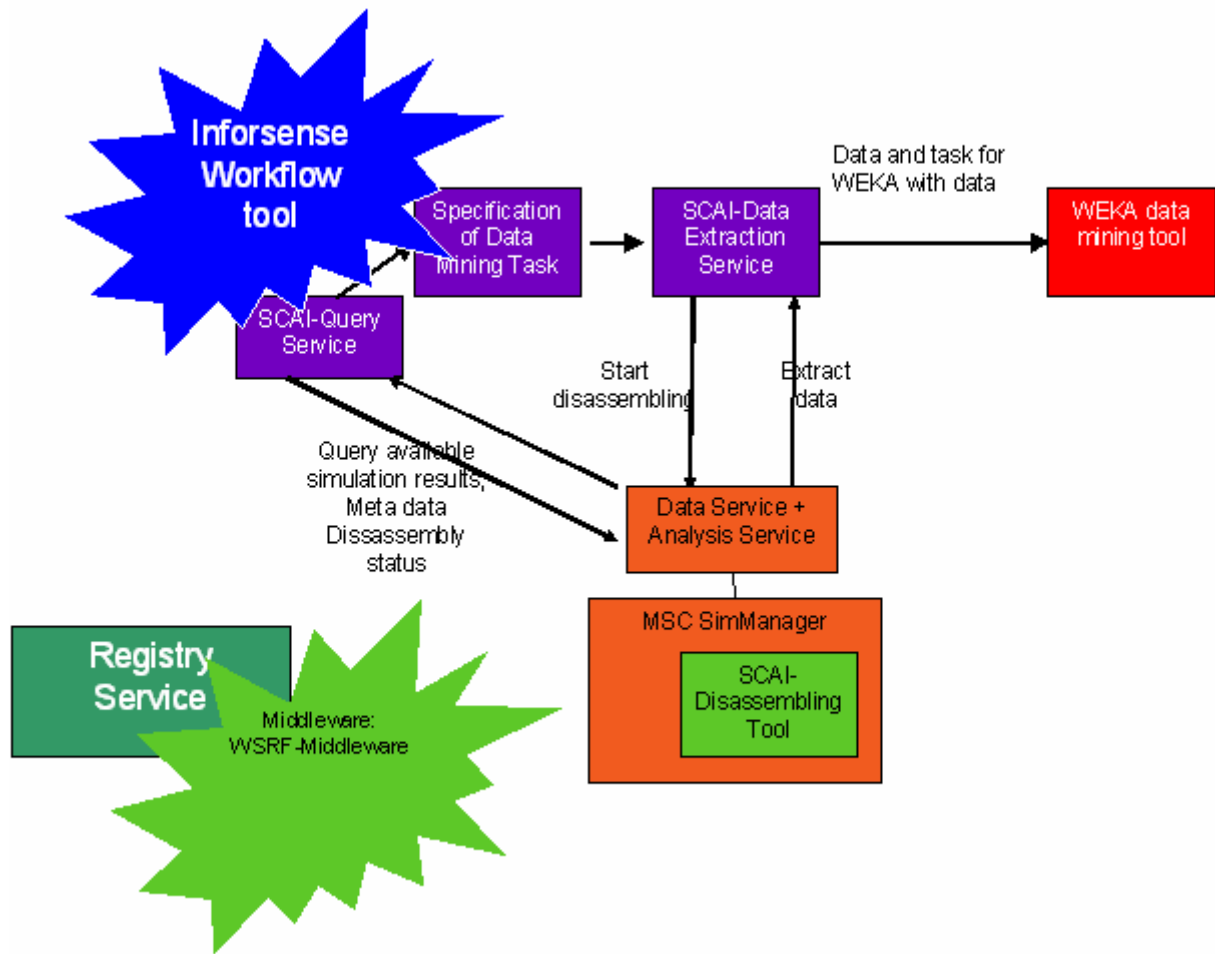


Figure 5: Architecture of the PM24 Auto-1 prototype

As the stand-alone PM12 Knowledge Service prototype was already tailored towards data mining with respect to the Auto-1 scenario, the use cases already introduced in deliverable D8.1.1 require only slight modifications. Please note that these use cases only apply to the data mining step illustrated in Figure 4.

Name	Remote Weka computation		
Use Case ID	UC8.1.6		
Date Created	2004-12-10	Source	FhG-AIS
Last Updated By	Thomas Niessen	Date Last Updated	2006-03-22
Project Phase	WP8.1 Task 6	Priority	High
Parent Package	Knowledge Discovery	Revision	1.1
Use Case Goal			
The user wants to analyse non-distributed meta-data about NVH behaviour with Weka			
Actors			
<ul style="list-style-type: none"> User (CAE engineer) 			
Pre-conditions			
<ul style="list-style-type: none"> User has the key accepted by the servers Requested data can be accessed by server-side scripts grid middleware is available on both sides Conversion from server-side data format and the format requested by the user is available 			

Post-conditions
<ul style="list-style-type: none"> • Result of Weka computations is available at user's client
Flow of Events
<ol style="list-style-type: none"> 1. User starts workflow editor on his client computer 2. User searches for appropriate data mining algorithm from Weka in the grid 3. User selects one data mining algorithm presented as result of his query 4. User selects the data server 5. a) Weka sends an instance of its selected component to the data server b) The data is accessed via OGSA-DAI 6. a) Data servers execute Weka component on selected data b) Data is transferred to the machine where Weka is running 7. Weka sends results to user 8. Results are displayed to user by Weka
Exceptions
<p>Server access is denied, network communication is down, grid middleware is not working at any side.</p> <p>Unable to accomplish request – requested data are not stored on the servers, data cannot be accessed by server-side scripts, data format conversion cannot be done.</p> <p>Able to accomplish request partially – some of requested data are available, but some are not.</p>

Name	Distributed cluster algorithm		
Use Case ID	UC8.1.7		
Date Created	2004-12-13	Source	FhG-AIS
Last Updated By	Thomas Niessen	Date Last Updated	2006-03-22
Project Phase	WP8.1 Task 7	Priority	High
Parent Package	Knowledge Discovery	Revision	1.1
Use Case Goal	The user wants to group documents of a distributed collection into clusters		
Actors	<ul style="list-style-type: none"> • User (CAE engineer) 		
Pre-conditions	<ul style="list-style-type: none"> • User has the key accepted by the server • Requested data can be accessed by server-side scripts • grid middleware is available on both sides • Training data (unlabeled) is available • Conversion from server-side data format and the format requested by the user is available 		
Post-conditions	<ul style="list-style-type: none"> • Requested data are stored on the server in format specified by the user and are available to download 		
Flow of Events			

1. User starts workflow editor on his client computer
2. User searches for appropriate distributed cluster algorithm from in the grid
3. User specifies (unlabeled) training data on different servers.
4. The user specifies a procedure for selecting data.
5. The data is selected on different servers.
6. The cluster procedure is distributed on suitable servers and a single iteration is performed.
7. Cluster processes synchronize on a central server
8. Check on the central server if results are sufficient, else go to 6.
9. The cluster attributes are stored on the central server or with the documents on the different servers.
10. Optionally: Cluster results are collected on a central client and visualized with a specific GUI.
11. User is informed when the job is finished

Exceptions

Server access is denied, network communication is down, grid middleware is not working at any side.

Unable to accomplish request – requested data are not stored on the server, data cannot be accessed by server-side scripts and data format conversion cannot be done.

Able to accomplish request partially – some of requested data are available, but some are not.

3.1.2.4 Detailed Description of the Requirements

3.1.2.4.1 Access to Distributed Data Repositories

Today, data is often stored in databases, which may differ significantly in the way the data is accessed (e.g., relational versus XML-databases). Furthermore, the required information may also be distributed across several sites. For accessing these information for data mining purposes, access to such distributed data repositories is required.

Name	Access to distributed data repositories		
Application Activity	Aerospace, Automotive		
Prototype(s)	Aerospace, Auto-1		
Date Created	03-13-2006	Priority	High
Created By	Jörg Kindermann	Technology component	Distributed Data Access (OGSA-DAI)
First Implementation Date		SIMDAT module targeted	Data
Description	Access to distributed data repositories such as databases (relational & possibly XML) and flat files.		
Relation to prototype	<ul style="list-style-type: none"> One central goal of grid-enabled data mining and knowledge discovery is the ability to run applications on distributed data. This can be achieved in two ways: the distributed data can be transferred to a central data mining server, or the algorithms can be executed in a GRIA-environment at the data servers. 		
Requested functionality			

- Data transfer via OGSA-DAI
- Remote execution of data mining applications via GRIA on the data server

Validation

- Issue queries against the relational and possibly XML databases deployed at
- Remote execution of data mining application is already part of the PM12 Knowledge Services prototype and thus has already been tested.

3.1.2.4.2 Other Requirements

Furthermore, the requirements 3.1.1.3.1 up to 3.1.1.3.5 also apply.

3.1.3 Aerospace: Cost-Modelling PM24 Prototype

3.1.3.1 Outline and Purpose of the Prototype

The PM 24 prototype in the Aerospace activity will integrate data mining functionality resulting from the technology area of Knowledge Services. The cost modelling scenario will use data mining techniques to generate cost estimates for future manufacturing based on data from a variety of sources. For example supplier manufacturing costs may be supplied as an OGSA-DAI database and historically material costs accessed via a web service or excel spreadsheet. These cost estimates are important for engineers to assess future costs of changes in the design of components, which will arise at the time the respective components will be built. For this purpose the different construction sites at BAE its partners will be connected to the site at FhG-AIS, which is running the data mining application. In later stages (after PM24) the data mining application may also run at one of the construction sites. For more detailed description of the prototype see D17.1.3. Fehler! Verweisquelle konnte nicht gefunden werden.

3.1.3.2 Initial Specification of the Prototype and its Use Case

Working with the initial PM24 prototype consists of two distinct steps – building the model and predicting future costs on the basis of this model. For these purposes it is required to establish access to the data stored at the different construction sites for building the model and to deploy a service to perform the cost-prediction step using a model build previously by data mining experts at FhG-AIS. The following modules related to data mining are part of this prototype:

- **Workflow editor provided by InforSense:** InforSense's workflow editor will be used to produce the models, which are part of the later prediction-step.
- **Registry for available data mining applications:** The registry contains descriptions of all available data mining applications. It allows users to search for specific applications and, together with the components integrated into the workflow editor, also provides some help to users on how to operate the selected application.
- **One or multiple data bases:** These data bases contain the required data for building the model.
- **Data mining applications:** At this time it is unclear which data mining application is most appropriate for this task. While Weka will be the first choice, due to the efforts already invested to make Weka available through GRIA, other data mining applications may deliver more accurate results.
- **Cost Prediction Service:** This service is accessed by the engineer for obtaining cost predictions about a particular component.

3.1.3.2.1 UC8.1.8 - Building Model for Cost Predictions

Name	Building Model for Cost Predictions		
Use Case ID	UC8.1.8		
Date Created	2006-03-22	Source	FhG-AIS
Last Updated By	Thomas Niessen	Date Last Updated	2006-03-22
Project Phase	WP8.1 Task 6	Priority	High
Parent Package	Knowledge Discovery	Revision	1.0
Use Case Goal			
The user wants to build a data mining model for cost modelling			
Actors			
<ul style="list-style-type: none"> User (data miner) 			
Pre-conditions			
<ul style="list-style-type: none"> User has the key accepted by the servers Requested data can be accessed by server-side scripts or OGSA-DAI grid middleware is available on both sides Conversion from server-side data format and the format requested by the user is available 			
Post-conditions			
<ul style="list-style-type: none"> Result (model) of data mining computations is available at user's client and stored in the grid for later predictions 			
Flow of Events			
<ol style="list-style-type: none"> User starts workflow editor on his client computer User searches for appropriate data mining algorithm in the grid User selects one data mining algorithm presented as result of his query User selects the data sources The data is accessed via OGSA-DAI Data is transferred to the machine where the data mining application is running The model is build by the application The application sends the results to the user The results are displayed to user The results are stored in the grid by the user 			
Exceptions			
Server access is denied, network communication is down, grid middleware is not working at any side.			
Unable to accomplish request – requested data are not stored on the servers, data cannot be accessed by server-side scripts, data format conversion cannot be done.			
Able to accomplish request partially – some of requested data are available, but some are not.			

3.1.3.2.2 UC15.1.6 - Supplier Chain Knowledge Extraction

Name	Supplier Chain Knowledge Extraction		
Use Case ID	UC15.1.6		
Date Created	01/10/2005	Source	BAE SYSTEMS
Last Updated By	Jamil Appa	Date Last Updated	08/03/2006
Project Phase	WP11.2	Priority	High
Parent Package	Aerospace Activity	Revision	1.1

Use Case Goal			
The estimation of future product cost information by exploiting data mining of historical supplier product information.			
Actors			
<ul style="list-style-type: none"> • Contractor (A) • Supplier (B) • Product Data Service (C) • Data Mining Service (D) 			
Pre-conditions			
<ul style="list-style-type: none"> • Existing Contractor – Service provider relationship with agreed QoS • Fixed/known service locations • Data exchanges are secure • Controlled access to services 			
Post-conditions			
<ul style="list-style-type: none"> • Account remains persistent for fixed period 			
Flow of Events			
<ul style="list-style-type: none"> • A – Create account at B • A – Request Product Cost Information at a future date • B – Process Query by engaging data mining service D • D – Apply data mining to data provided by C • B – Supply results to A 			
Exceptions			
<ul style="list-style-type: none"> • Service Failure needs to be handled gracefully 			

3.1.3.3 Detailed Description of the Requirements

3.1.3.3.1 Cost Prediction

This requirement is the heart of the whole prototype.

Name	Cost Prediction		
Application Activity	Aerospace,		
Prototype(s)	Aerospace, Knowledge Services		
Date Created	03-13-2006	Priority	High
Created By	Mike Turner	Technology component	Knowledge Services
First Implementation Date		SIMDAT module targeted	Knowledge Services
Description	Use of knowledge toolkits to identify relationships and patterns in data for cost predication.		
Relation to prototype	<ul style="list-style-type: none"> • Part of the aero scenario will look at trying to predict the cost of manufacturing a specific part in the future based on current and historical data. 		
Requested functionality	<ul style="list-style-type: none"> • Ability to estimate future costs based on historic cost data. • Ability to discover patterns given distributed data sets. 		
Validation	<ul style="list-style-type: none"> • 		

3.1.3.3.2 Access to distributed Data Repositories

This requirement is the same as for the Automotive: Auto-1 PM24 Prototype (see 3.1.2.4.1 Access to Distributed Data Repositories).

3.1.3.3.3 Other Requirements

Furthermore, the requirements 3.1.1.3.1 up to 3.1.1.3.5 also apply.

3.1.4 Pharma

3.1.4.1 Outline and Purpose of the Prototype

According to the new Annex I Knowledge Services will continue to collect detailed requirements from the Partners in the Pharma activity. However, contributions of the Knowledge Services area to the Pharma prototypes are not planned for PM24. We therefore do not reproduce the Pharma use case of deliverable D8.1.1 here. In this sense, the requirements described in the following sections of preliminary, and they will be complemented by an appropriate use case later.

3.1.4.2 Detailed Description of the Requirements

3.1.4.2.1 Addition of Mapping of relevant Biological Terms to Sequence Analysis Reports

This requirement describes a knowledge service, which can add information to the results of a database similarity search by means of text mining applied to peripheral data sources like Medline, GO, Ensembl, etc.

The SIMDAT-Pharma prototype produces annotation of the cDNA sequences that are submitted to the workflow by the scientist. This annotation essentially reflects the contents of the “description” field present in the public sequence databank(s) in which a hits were identified by the BLAST procedures (blastn against the EMBL nucleic acids databank or blastx against the protein databank Swissprot). Till now, no effort has been done to exploit the large knowledge present in the complete documentation of these public sequence databanks, notably the cross-references of each entry to PubMed, the Gene Ontology database (GO), the Ensembl genomic databank, and several others. This correlated information is an ideal candidate for an effort in data mining as mentioned in the objectives of WP8.1 of the Annex 1 (“annotation and indexing of heterogeneous resources that are stored on distributed servers”).

The data mining procedure would consist in several steps like:

- Collect the complete sequence entries corresponding to each significant hit identified by a blast workflow step, as well as their counterparts in the related databanks (protein ↔ nucleic acids)
- Index the documentation of these entries
- Evaluate the frequency of words and co-occurring words
- Collect from their respective Internet (GO, PubMed, Ensembl, etc.) servers the various cross-referenced entries
- Index these datasets and their ontology-based contents
- Produce a statistical evaluation and clustering of relevant terms
- Produce a searchable report, integrated with the raw workflow result

However, currently this requirement is rather abstract, being the result of ongoing discussions between Pharma and Knowledge Services (See also deliverable D14.1.3). It will be formulated more precisely in the course of the project.

Name	Addition of mapping of relevant biological terms to sequence analysis reports		
Application Activity	Pharma		
Prototype(s)	Pharma		
Date Created	03-16-2006	Priority	high
Created By	Richard Kamuzinzi, Jörg Kindermann	Technology component	Workflow, Knowledge Services
First Implementation Date		SIMDAT module targeted	Information Services
Description	Sequence analysis reports can be enriched by the addition of mapping of relevant biological terms related to discovered similarities.		
Relation to prototype	<ul style="list-style-type: none"> • Each databank similarity search, which produces a ranked list of relevant hits, is submitted to data mining searches in peripherals data sources (Medline, GO, Ensembl, ...) • The results enrich the documentation of each entry in the project database 		
Requested functionality	<ul style="list-style-type: none"> • This data mining product is a module that can be triggered by a corresponding workflow step • A set of specific terms (or keywords) provided by the scientist might orient the data mining process towards his/her specific area of interest. • The report integrates global statistics on terms discovered by the data mining processes. 		
Validation	<ul style="list-style-type: none"> • A rate of relevance provided by an expert of a specific domain has to be reached 		

3.2 Knowledge-Based Directory/Registry Service

3.2.1 Pharma: Semantic Service Broker

The PM12 Pharma prototype was a distributed grid application, which provides federation of SRS installations running on remote servers. That consists of three main components:

- Directory services (Semantic Broker)
- Node broker (SRS nodes)
- Federated Portal (user interface).

See also Figure 6 The directory service is knowledge-based. It is described in detail in deliverable D14.1.2. We will give a short overview here. The directory service consists of three modules:

- The ontology repository (knowledge model) stores the description of computational and data resources.

- The semantic broker provides functionality for both client and service provider to publish or discover services, according to a controlled vocabulary (to ontology).
- The annotation service annotates service instances in order to link the formal description and the reference to the service.

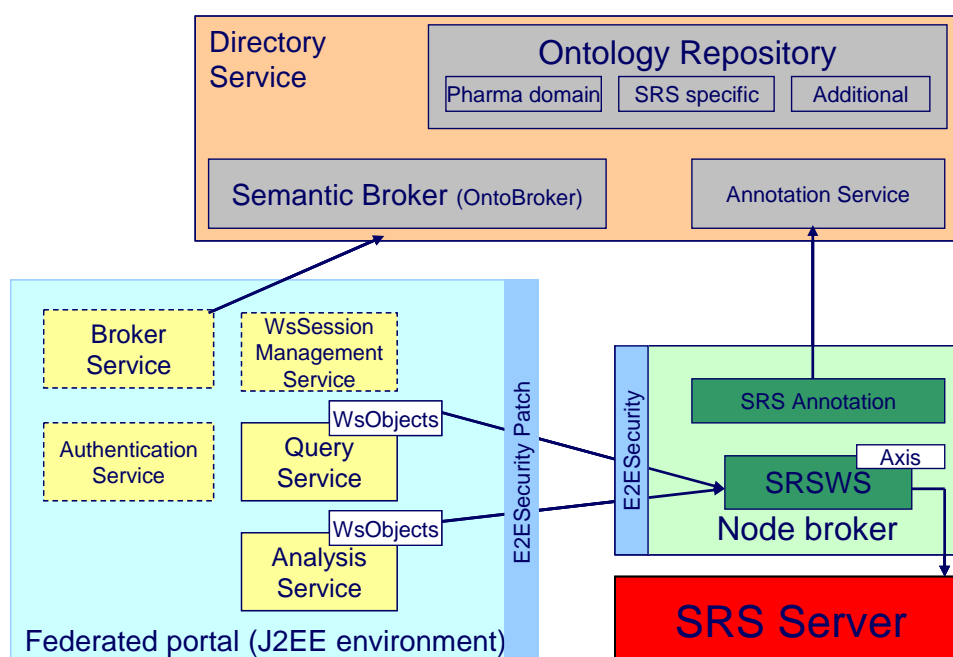


Figure 6: Federated Pharma portal overview

3.2.1.1 Ontology Repository (Knowledge Model)

The knowledge model is built from multiple connected ontologies, serving as formal models of the knowledge domains in question. They help to structure information sources and describe complex dependencies between them efficiently. The ontologies of the knowledge model are conceptionally based on OWL-S. In particular, these are:

- The service type ontology describes the available computation services
- The data source ontology describes the available data sources
- The subject ontology describes the scientific terminology

3.2.1.2 Semantic Broker

The broker provides functionality to discover or publish services, according to the knowledge model. It supports both the client side (end user) and the service provider. It operates on the F-Logic based bioinformatics ontology and the OWL-S based service annotations for discovery and matchmaking. In addition, it also implements knowledge model browsing for the client side. Furthermore, there is a client side query API to wrap OWL-S specific query syntax.

3.2.2 Pharma: Searchable Registry of Bioinformatics Sequences Analysis Tool

3.2.2.1 Detailed Description of the Requirements

3.2.2.1.1 Searchable Registry of Bioinformatics Sequences Analysis Tools

The goal of this requirement is to implement a service which helps bioinformatics scientists to locate analysis tools, which are suitable for their analysis tasks and data. FhG-AIS will implement the service, which supports the selection of bioinformatics analysis tools required for the completion of a workflow. The service will be based on the two-way mapping of an ontology of biocomputing terms against the names of biocomputing applications. This mapping will be provided through a joint action by Pharma and Ontologies. Basic grid Infrastructure will provide the technology to implement the grid service. The service will be integrated into the Semantic Broker developed by NEC.

Name	Searchable registry of bioinformatics sequences analysis tools		
Application Activity	Pharma		
Prototype(s)	Pharma, Knowledge Services		
Date Created	03-16-2006	Priority	Medium
Created By	Richard Kamuzinzi, Jörg Kindermann	Technology component	Workflow, Ontologies, Knowledge Services, Basic grid Infrastructure
First Implementation Date		SIMDAT module targeted	Information Services
Description	Searchable registry to mine available analysis tools required for workflow completion		
Relation to prototype	<ul style="list-style-type: none"> To enable a workflow composition based on bio-computing terms instead of technical terms (i.e. tool names), the description of analysis tools must provide a mapping between bio-computing terms (e.g. searching similarities to nucleic acid databanks) and concrete application names (e.g. blastn or fasta) 		
Requested functionality	<ul style="list-style-type: none"> Store the relationships between the abstract bio-computing functionalities and the concrete analysis tools available Allow querying of this mapping repository (from abstract term to concrete tool or <i>vice versa</i>) 		
Validation	<ul style="list-style-type: none"> Any analysis tool must map to a functional class and any functional class must contain at least one tool 		

3.2.3 Automotive: SimManager interface for semantic queries.

An OntologyService query interface – with OntoBroker behind – to be used by a SimManager client is part of the Auto 2 Prototype PM24. For details see deliverable D 6.1.4

3.2.4 Aerospace: interface for semantic queries

Ontoprise plans to provide an ontology-based interface for semantic queries to Aerospace. For details see deliverable D 6.1.4

3.3 Response Surface Modelling

In the Aerospace activity BAE is also planning to develop and integrate Knowledge Services for response surface modelling. BAE intends to investigate using response surfaces as a surrogate data model at each analysis service. The idea is that each analysis service can build up a response surface based on previous runs that is particular to the characteristics of that analysis (i.e. the structural analysis response might be more linear than the aerodynamics). Then in the Aerospace scenario when the design service requests data from each analysis service they can either calculate the design point or give an answer based on the response surface. For details see the Aerospace deliverable D17.1.3, in particular the section on Knowledge Services.

3.4 Semantic Annotation Tools

3.4.1 TUAM¹³ Annotation Tool

This section describes a semantic annotation tool for the Pharma activity. The proper annotation of data and information could greatly facilitate the discovery of relevant scientific information in large data sets, but so far the introduction of “semantic glue” between the data is a time and resource-consuming effort. TUAM (Tool for Universal Annotation and Mapping) has been designed to support the collaborative annotation of data and to enable efficient expert knowledge capturing. TUAM is based on a simple subject-predicate-object format, which gives the user the possibility to establish n:m relationships between any type of data records; e.g. controlled vocabularies and database entries, databases and ontologies, molecules and text sources. When annotations are centred on controlled vocabularies like ontologies, they become semantic mediators, conferring well-defined meanings to diverse data source elements. Annotations generated using TUAM are non-destructive and are stored persistently in a relational database system. They can be exported in a generic XML-format, which allows subsequent processing for report generation or data analysis with special software add-ons. The system can handle objects in relational databases, in XML documents (RDF, GraphMC, and OWL), in ASCII files or tab-delimited file formats and Web site content. The annotation procedure itself is supported by intuitive tabular or graphical navigation through the annotation space.

3.5 Workflow Construction Advisor

The workflow construction advisor as shown in Figure 7 is a prototype which fits into several application activities. Its purpose is to assist users in constructing their workflows by proposing next workflow elements, parameters and indicating similar workflows to the one currently being assembled. For this purpose the prototype will contain several distinct modules from different technology champions (InforSense and FhG-AIS) of which only the one related to pattern matching is discussed here. For a detailed description of the whole prototype see deliverable D5.1.4.

¹³ www.scai.fraunhofer.de/fileadmin/download/flyers/Projektblatt_TUAM.pdf

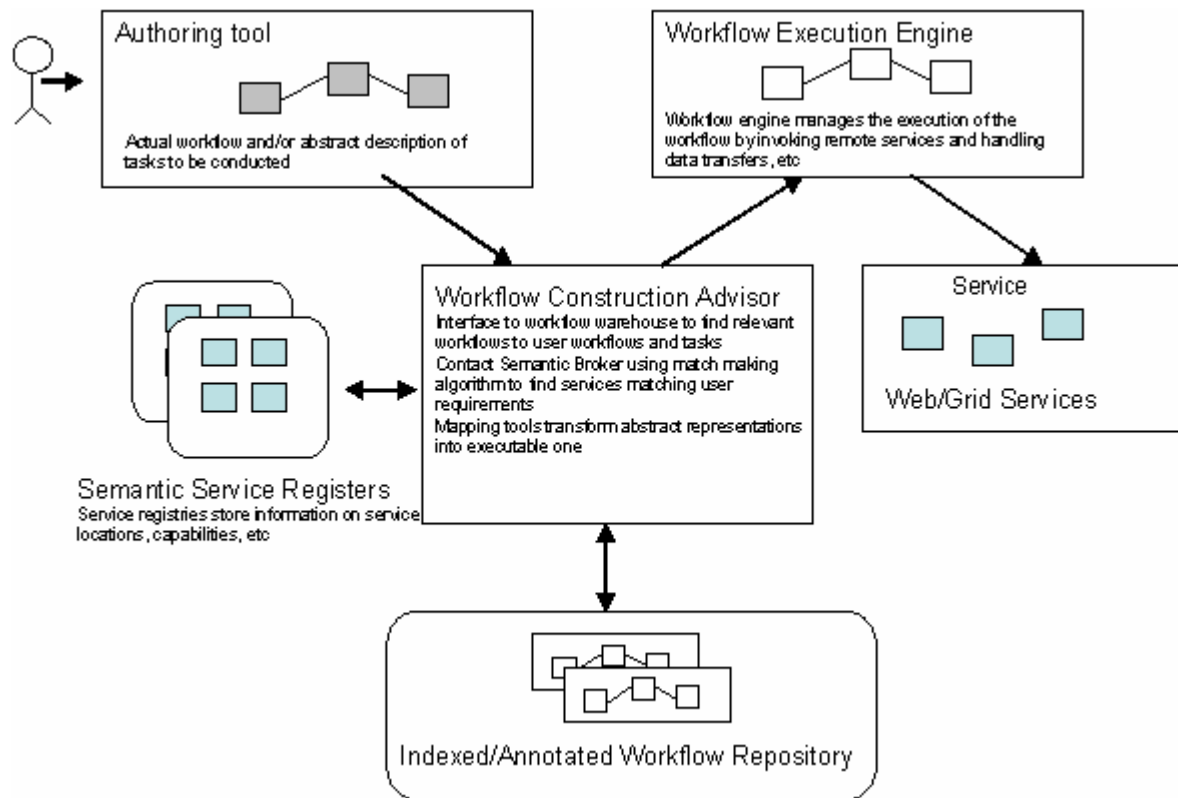


Figure 7: Overview of Workflow Construction Advisor

The workflow advisor will help a user in the construction of his workflows by proposing next steps as requested by the user. The workflow advisor will base its proposals not only on the properties of data and algorithms to be used in the workflow, but also on "best practice" experience gained from the workflow warehouse or repository. The knowledge about best practice procedures within the workflow is a result of data mining applied to the workflow repository.

FhG-AIS will provide the required workflow pattern mining algorithms. Workflows can be represented as directed annotated graphs. They can be compared, matched, classified, and clustered by using graph-mining algorithms. These inductive approaches can be complemented by rule-based procedures. The rules themselves can be either induced (e.g. by decision trees), or added manually. Integration of our algorithms in the workflow advisor will occur after project month 24.

4 Implementation Plan

Task	Description	Partners Involved	Time	Application Area	Prototype
Application Meta-Data Description Infrastructure					
Development of an appropriate meta-data schema	A XML-schema for describing data mining applications. This will take place internally.	FhG-AIS	PM19 – PM21	Automotive, Aerospace	PM24 Prototype Auto-1 & Aerospace: Cost-Modelling
Integration into a suitable registry including query mechanisms	The meta-data information has to be stored in a grid-wide registry and must allow users to search for specific applications. This will initially take place internally.	FhG-AIS	PM20-PM21	Automotive, Aerospace	PM24 Prototype Auto-1 & Aerospace: Cost-Modelling
Integration to Workflow Editor	Searching for suitable data mining applications and processing the returned meta-data requires appropriate client-side components to be integrated into the workflow editor.	InforSense, FhG-AIS	PM21-PM22	Automotive, Aerospace	PM24 Prototype Auto-1 & Aerospace: Cost-Modelling
Automotive: Auto-1					
Access to distributed data repositories	The meta-data generated by FhG-SCAI may reside in different repositories. Furthermore, the required information for cost-prediction in the Aerospace activity will reside in distributed repositories.	FhG-SCAI, FhG-AIS	PM20-PM22	Automotive, Aerospace	PM24 Prototype Auto-1 & Aerospace: Cost-Modelling
Integrating the PM12 Knowledge Services prototype into the Auto-1 prototype	The PM12 Knowledge Services prototype already was developed with the aim of later integration into the Auto-1 prototype.	FhG-AIS, FhG-SCAI, MSC	PM22-PM23	Automotive	PM24 Prototype Auto-1
Aerospace: Cost-Modelling					
Development of a	The actual cost-prediction on the	FhG-AIS	PM22-PM24	Aerospace	PM24 Prototype

service for cost-prediction	basis of the previously build model has to be done by a dedicated service.				
Workflow Authoring Assistant					
Requirements gathering for Authoring Assistant	Requirements are to be captured from the Aerospace based on existing workflows used in that sector	InforSense	PM19-PM21	Aerospace	Requirements gathering
Design of Authoring Assistant	Based on the requirements gathered from the Aerospace activity, the Authoring assistant will be designed.	InforSense	PM21-PM4	Aerospace	No prototype
Implementation of interfaces for integration of workflow comparison algorithms	The workflow pattern mining algorithms defined by FhG-AIS will be used to design and implement an interface where they can be plugged into an authoring assistant. These patterns will be based on the workflows provided by the University of Southampton and InforSense.	InforSense, FhG-AIS, University of Southampton	PM27-PM30	Aerospace	No prototype
Prototype Implementation	Once the algorithms are implemented into the authoring tool, a prototype will be deployed in an application area	InforSense, FhG – AiS, University of Southampton	PM24-PM30	Aerospace	Prototype of workflow authoring assistant

5 Future Work: Outline of the PM30 Prototypes

5.1 Workflow

FhG-AIS will develop graph-based workflow mining algorithms based on workflow data provided by Inforsense. These algorithms will be tested in cooperation with Inforsense, and integrated into the prototype of Inforsense's workflow authoring assistant. In the subsequent step it is planned to deploy the workflow as a prototype in one of the application areas. See also the deliverable D5.1.4.

5.2 Aerospace

The Knowledge Services module of the PM 24 prototype will be operational mainly with respect to its technical features, i.e. it will only be proof-of-concept. The result will be an operational service for cost modelling on the basis of a knowledge model developed by a FhG-AIS. We will develop data mining algorithms suitable for cost-modelling and provide the resulting knowledge models until PM 30. See also the deliverable D17.1.3.

5.3 Automotive

A preliminary requirement of the Auto-3 prototype is a knowledge service which is applicable to the crash simulation meta-data database. Whether this knowledge service will be similar to the one integrated into the Auto-1 prototype, remains to be worked out. See the deliverable D11.1.5.

5.4 Pharma

The requirement "addition of mapping of relevant biological terms in sequence analysis reports" will be refined, augmented, and complemented by a scenario description. Based on this work, we will select data mining algorithms suitable to fulfil this key requirement.

6 Conclusion

This document describes the major achievements of Knowledge Services in SIMDAT until PM 18, the design and implementation plans until PM 24, and future work. As already said in the introduction, Knowledge Services is at the center of SIMDAT research and development. But it was only natural that during the first phase of SIMDAT. But it was only natural during the first "Connectivity" phase of SIMDAT (PM1 - PM18) that Knowledge Services evolved only slowly, the main reason being that Knowledge Services must rely on operable grid infrastructure, and further components from the other technology areas. The application areas themselves are so had to get connected to the technologies, and they had to define the requirements resulting from their applications evermore precise.

In the first 18 months, major developments in SIMDAT have been conducted in the following areas: The Knowledge Services prototype features grid in a bid to the car as an example of **grid-enabled data mining**. It was operational already at PM 12. Although not be integrated into an application prototype, the new PM 24 developments of cost modelling in Aerospace and distributed clustering in Automotive will be based on our prototype. **Knowledge-based Registry services** have been established for Automotive by Ontoprise, and Pharma by NEC. A similar service is envisaged for Aerospace until PM 24. Other Knowledge Services are going to be operational at PM24, among them **Response Surface Modelling** in Aerospace, **semantic annotation** in Pharma. A knowledge-based **workflow construction advisor** is planned for PM30.

To summarize, and we can state that the technology uptake of Knowledge Services in the application areas of SIMDAT is gaining momentum. This is a very positive result, which shows that the starting difficulties have been overcome. In particular, the design of the PM12 Knowledge Services prototype, by FhG-AIS and Inforsense has proven to be appropriate, because it will be taken as a basis of developments in Automotive and Aerospace.