VIP ANR-09-COSI-03
Virtual Imaging Platform

D1.2.1: Data and tools repositories with basic query interface

Alban Gaignard	MODALIS (I3S)	alban.gaignard@i3s.unice.fr
Johan Montagnat	MODALIS (I3S)	johan@i3s.unice.fr
Tristan Glatard	CREATIS	tristan.glatard@creatis.insa-lyon.fr
Rafael Ferreira Da Silva	CREATIS	silva@creatis.insa-lyon.fr

Abstract

This document describes the setup of data and tools repositories and their basic querying through the NeuSemStore framework deployed in the VIP platform. NeuSemStore is a Semantic Data Store supporting scientific workflows, and aimed at persisting and retrieving semantic annotations. Semantic data consist of service annotation, model annotation and invocation annotation (provenance information collected at workflow run-time). This software was initiated in the NeuroLOG project ANR-06-TLOG-024 (http://neurolog.polytech.unice.fr) and is being extended in VIP project ANR-09-COSI-03 (http://www.creatis.insa-lyon.fr/vip).

Contents

1 Introduction

The Virtual Imaging Platform (VIP) project builds a platform dedicated to the simulation of multi-modal, multi-organ and dynamic medical images. Integrating several simulators into a unique platform raises interoperability issues. A semantic approach has been adopted to tackle interoperability challenges and enhance the reusability of simulator components (tools), models and simulated data. To assist users in the setup of new simulation experiments (simulation workflows) or in the parametrization of existing simulators, the system will rely on knowledge bases describing simulation models, simulated data and simulation components.

This document briefly describes:

1. The VIP data repository (section 2), which consists of an interface to grid data storage, a database of indexed VIP data, and a graphical interface to browse, upload and download data.

2. The VIP semantic repository (section 3), which consists in a set of software components dedicated to the storage and querying of generic semantic annotations, but also dedicated to their exploitation to populate and query simulation components (tools) and to track workflow provenance, a first step toward the automated production of domain annotations during simulation experiments.

This deliverable (D1.2.1) constitutes the core software component on which will be built the simulation workflow designer (D1.2.2) and the VIP client to semantic and execution services (D2.3.4), both planned for year 3 of the project.

2 VIP data repository

To cope with both CPU and Data intensive simulation experiments, tasks performed through the VIP platform exploit the EGI[1] grid infrastructure. This infrastructure also offers several Petabytes of storage distributed on various sites. Raw data sets are stored on EGI Storage Elements and they are identified by their Logical File Name (LFN). Consequently, to store and retrieve files, the VIP platform relies directly on the EGI file catalog (LFC). To improve the reliability of the LFC, a cache mechanism has been

[1]EGI, European Grid Infrastructure, http://www.egi.eu

developed (milestone M2.2.2) and guaranties that raw files have been properly replicated over several available storage elements.

A module of the VIP portal was developed to offer online access to this storage system. Its main features are:

- upload/download of data files to/from EGI storage elements;
- upload of local directories to EGI storage elements (using an applet);
- automatic file replication on EGI storage elements (to improve reliability);
- caching of grid files on the platform (to improve efficiency);
- soft file deletion in *Trash* directory;
- private and shared storage spaces;
- asynchronous "pool" file transfers (to avoid network congestion);

The backend implementation relies on the vlet library[2] to access LFC and storage elements. For performance and reliability reasons, native gLite clients are used in case they are installed on the portal machine. A vlet agent was developed to centralize and sequentialize the transfers done from the portal. It is available online[3].

Figure 1 shows the VIP file transfer interface. The left panel lists the directories that the user can access. The *Home* directory is private while the other *Group* directories are shared (read and write) with the other group members. Controls allow to browse, cut, paste, rename, upload, download and delete files and to create directories. The right panel shows the transfer history. Coloured icons indicate the nature and status of the transfers.

Administrative interfaces also show a global history of file transfers (for all users) and statistics about cached files (file sizes and hits).

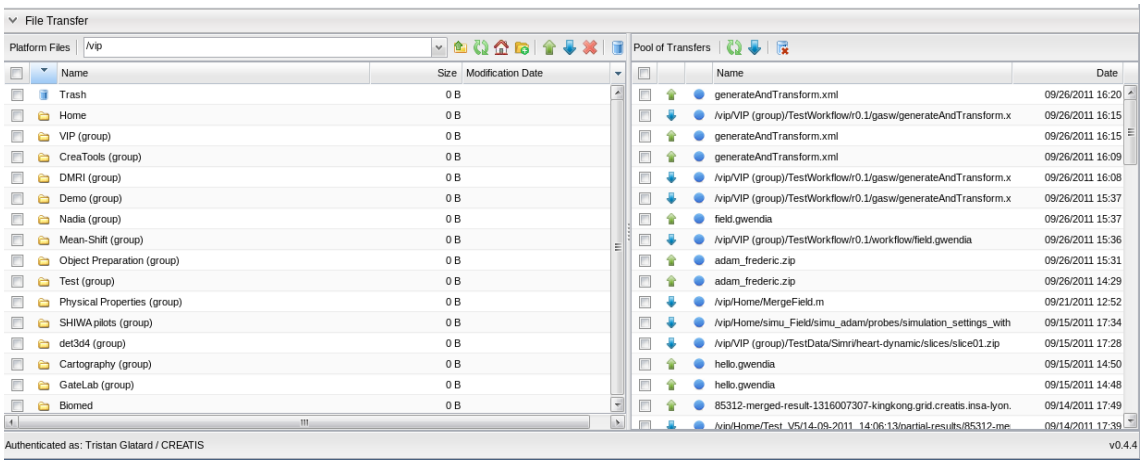

Figure 1: VIP file transfer interface.

[2] http://www.nikhef.nl/~ptdeboer/vlet/
[3] http://kingkong.grid.creatis.insa-lyon.fr:9002/projects/vletagent

3 VIP semantic repository

VIP's semantic repository is implemented through a set of Java libraries, web services, and graphical user interfaces seminally developed in the context of the NeuroLOG project (ANR-06-TLOG-024) and known as the NeuSemStore framework. It supports scientific workflows design and execution with fully enabled semantic technologies This framework is delivered as a software available from: http://nyx.unice.fr/projects/neusemstore/wiki/WikiStart.

3.1 Cataloging simulation models and simulated data

Simulation models and simulated data share the same components as they are both composed of raw data (files) and completed by their descriptive metadata (semantic annotations). While raw data are stored and indexed through the data management layer provided by the EGI middleware (LFC), the semantic annotation are managed through the NeuSemStore framework.

Finally, semantic annotations also encompass raw data LFNs so that, as illustrated in figure 2, it is possible to retrieve raw files through semantic queries exploiting the VIP ontology.

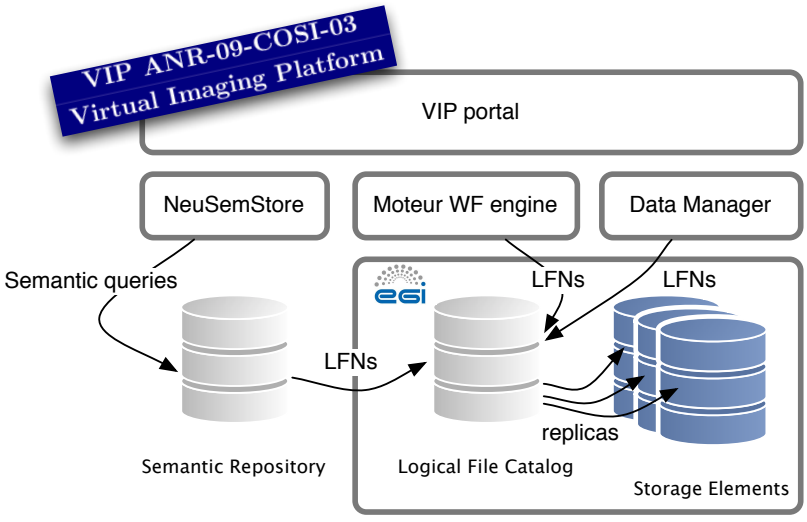

Figure 2: Raw data and Semantic data repositories in the VIP platform

3.2 Persisting and querying semantic annotations

The NeuSemStore core module handles semantic annotation persistency and querying. To persist semantic annotations, represented as RDF statements, this module rely on the standard semantic web framework JENA[4] and it's relational database back-end SDB[5]. Finally, RDF statements are stored into a MySQL relational database whose stability and

[4]JENA semantic web framework, http://www.openjena.org
[5]JENA SDB, http://www.openjena.org/SDB

scaling need no further proof. To address potential performance issues, the persistency backend could easily be re-configured to rely on the optimized TDB[6] backend.

The retrieving of semantic annotations is realized through the Corese/KGRAM[7] semantic engine implementing W3C standards such as RDF(S), SPARQL 1.1 (Query/Update) and SPARQL rules for RDF.

3.3 Cataloging simulation tools

To be semantically browsable, VIP simulation component must beforehand be annotated by domain experts with concepts of the VIP ontology. Being under development the VIP ontology is not yet available to describe simulation components. However in this deliverable, we rely on the OntoNeuroLOG[8] ontology to annotate the functionality of the simulation tools and to describe their input and output parameters (OntoNeuroLOG shares the same conceptual framework as the VIP ontology).

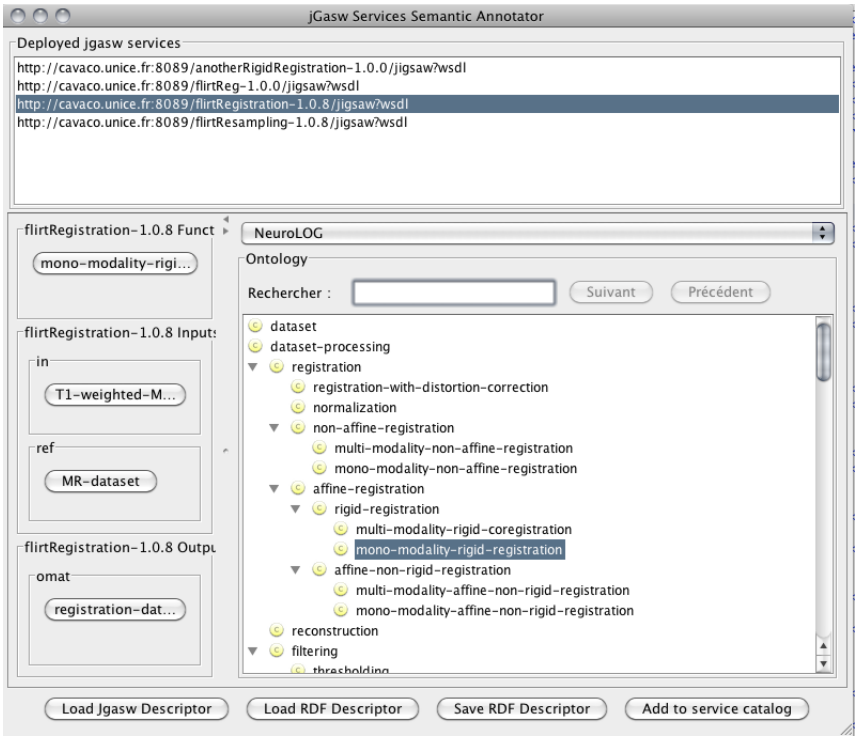

Figure 3: Semantic service annotation GUI

Figure 3 illustrates the annotation of a medical image registration tool. End-users are able to select the tool they want to annotate into a repository of invocable Jigsaw services (tools packaged and deployed as standard web services) . Then, they are able to (1) browse two taxonomies describing dataset-processings (the functionality of tools) and datasets (the semantic nature of data) and to (2) drag-and-drop concepts into functionality or input/output parameters areas. Finally, resulting RDF files can be stored locally or published into the semantic tool catalog.

[6] JENA TDB, http://www.openjena.org/TDB
[7] Corese/KGRAM, http://www-sop.inria.fr/edelweiss/software/corese
[8] OntoNeuroLOG ontology,

Indexing semantic annotation of simulation component, the semantic catalog itself is deployed as a JAX-WS web service so that experts are able to publish new tool descriptions, or simulation workflow designers are able to browse the catalog by semantic functionality to select the most suitable component. SPARQL is used as the underlying query language and complex semantic queries are hidden to the end-users.

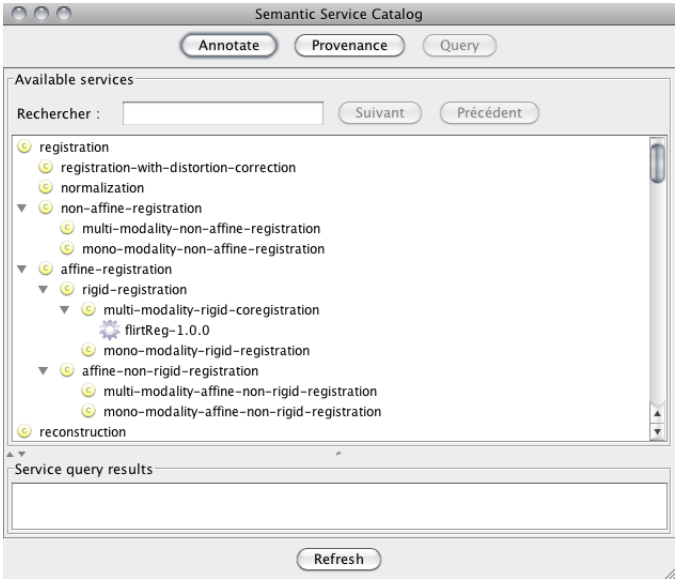

Figure 4: Tools semantically classified by functionnality

3.4 Tracking workflow provenance

During simulation experiments, provenance information is recorded through the NeuSemStore-provenance module, enabling the tracking of each simulation component invocation and the data it consumes and produces. Based on the OPM[9] specification and its representation as an ontology, provenance statements are recorded on-the-fly and stored into a semantic repository provided by the NeuSemStore-core module. Hiding complex SPARQL queries to the end-users, a specific graphical user interface has been developed and is illustrated in figure 5. End-users are able to list all simulation experiments. For each experiment, they can list all invocations of simulation components and the data globally[10] consumed as input and produced as output. Finally, when an invocation has been selected, the system is able to retrieve the consumed input data, and the produced output data, at the scale of this single invocation.

Figure 6 illustrates a semantic query based on the path expression language of SPARQL 1.1 allowing to retrieve the genealogy of a processed data. More precisely, it is possible to list all originating data for a selected output data. Conversely end-users are also able to list all derived data from a selected input data.

[9] Open Provenance Model, http://openprovenance.org
[10] at the scale of the simulation workflow

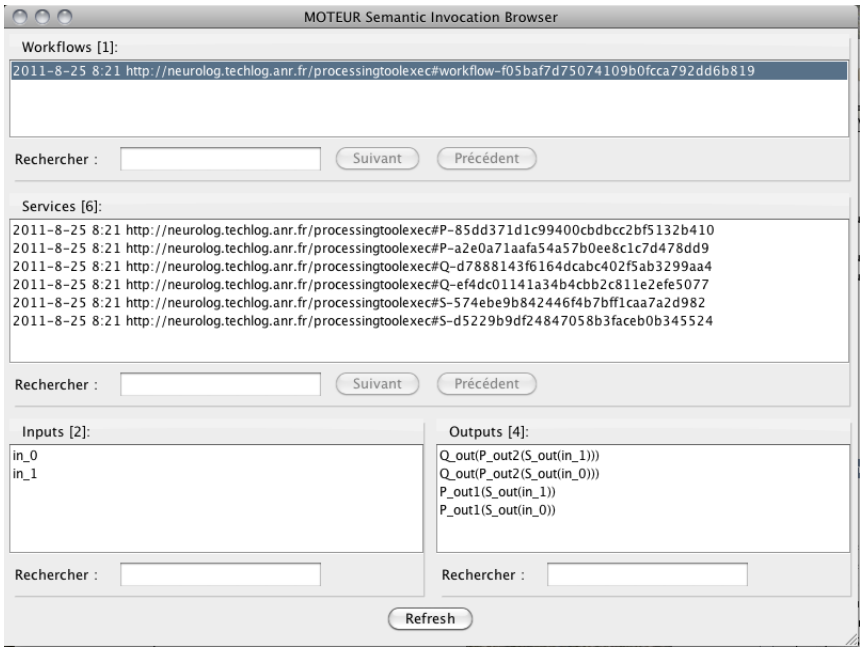

Figure 5: Navigating workflow provenance

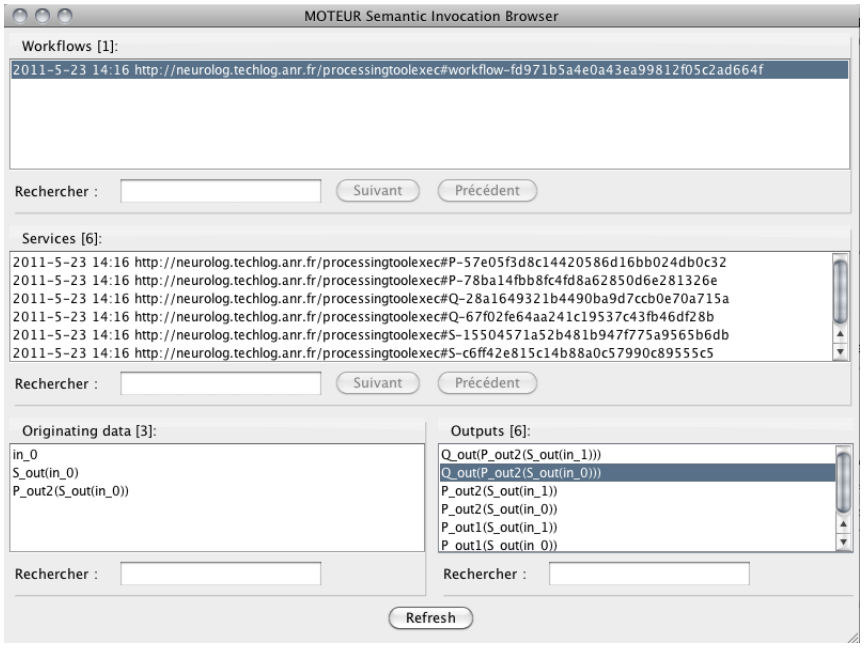

Figure 6: Retrieving the genealogy of processed data

4 Conclusions and future work

Through the NeuSemStore framework, this document describes how semantic annotations are used as an entry point to catalog simulation tools, simulation models and simulated data. The core and graphical user interface components presented here constitute a first basis toward (1) a full integration into the VIP portal which is planned for deliverable D2.3.4 and (2) the development of the simulation workflow designer planned for deliverable D1.2.2, exploiting the richness of the VIP ontology.

The provenance facet of NeuSemStore is currently under tests to assess the scalability and reliability of the system, mainly because it relies on inference rules. The results of these tests will be decisive to automate the production of new semantic annotations at simulation experiment run-time.