

# Developing an Incomplete Reasoner in Five Minutes: the Large Knowledge Collider in Action

Alexey Cheptsov<sup>1</sup>

<sup>1</sup> High-Performance Computing Center Stuttgart, Nobelstr. 19,  
70569 Stuttgart, Germany

cheptsov@hirs.de

**Abstract.** The Large Knowledge Collider (LarKC) is a prominent development platform for the Semantic Web reasoning applications. Guided by the preliminary goal to facilitate the incomplete reasoning, LarKC has evolved in a unique platform, which can be used for the development of robust, flexible, and efficient semantic web applications, also leveraging the modern grid and cloud resources. As a reaction on the numerous requests coming from the tremendously increasing user community of LarKC, we set up a demonstration package for LarKC that is intended to present the main subsystems, development tools and graphical user interfaces of LarKC. The demo aims for both early adopters and experienced users and serves the purpose of promoting Semantic Web Reasoning and LarKC technologies to the potentially new user communities.

**Keywords:** LarKC, reasoning, workflow

## 1 Introduction

Development of the internet-scale data-centric applications has been recognized as the primary challenger in the Semantic Web community. As a reaction on this challenge, several leading Semantic Web research organizations and technological companies have joined their efforts around the project of the Large Knowledge Collider (LarKC), supported by the European Commission. The LarKC's main value is twofold. On the one hand, it enables a new approach for large-scale reasoning based on the technique for interleaving the identification, the selection, and the reasoning phases. On the other hand, through over the project's life time (2008-2011), LarKC has evolved in an outstanding, service-oriented platform for creating very flexible but extremely powerful applications, based on the plug-in's realization concept. The LarKC plug-in marketplace has already comprised several tens of freely available plug-ins, which implement new know-how solutions or wrap existing software components to offer their functionality to a much wider range of applications as even originally envisioned by their developers. LarKC is an open source development, which encourages collaborative application development for Semantic Web. Despite being quite a young solution,

LarKC has already established itself as a very promising technology in the Semantic Web world. Some evidence of its value was a series of Europe- and world-wide Semantic Web challenges won by the LarKC applications. As the most successful LarKC-based developments can be referred Bottari – the Semantic Challenge winner in 2011, and WebPIE – the Billion Triple Challenge winner in 2010. In order to promote the reasoning ideas and practical solutions of LarKC to new development communities, we set up a demonstration package that is guiding the early adopter through the main steps towards creation of a scalable reasoning application. We believe that our demo will allow the LarKC-based service-oriented reasoning technology to be promoted to several new user and developer communities.

## 2 Contents of the Demo

The demo will guide the user through the main steps towards creation of an exemplarily LarKC application (Fig. 1). The application will be used to answer queries executed against the Linked Life Data, Sindice, and other chosen RDF knowledge bases. The main steps include:

1. Downloading and getting started with the LarKC platform.
2. Browsing the LarKC Market Place and identification of the needed plug-ins.
3. Creation and execution of a simple workflow based on the identified plug-ins with the Workflow Designer application.
4. Making the basic workflow more complicated by merging and splitting the application data flow (e.g. merging results acquired from multiple data bases).

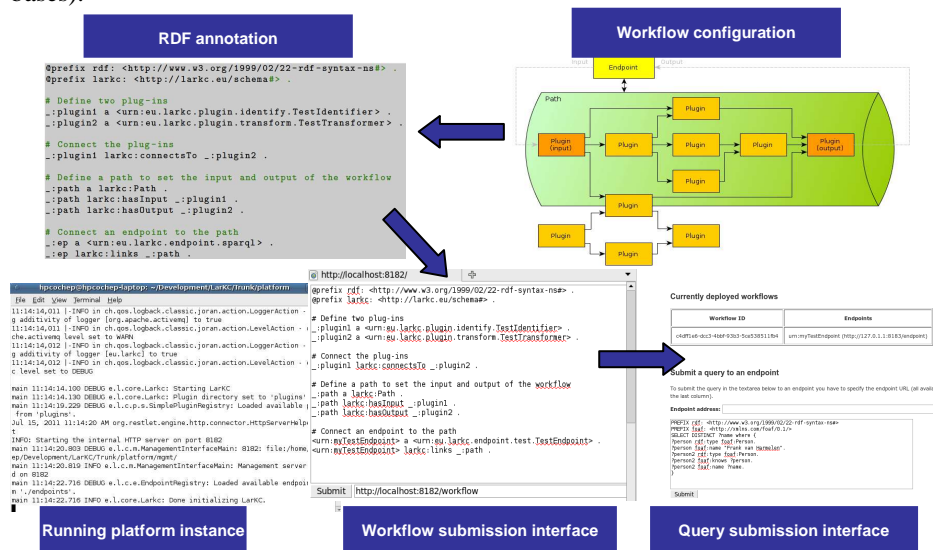


Fig. 1. Life-cycle of a LarKC reasoning application.

### 3 Getting Started With LarKC

LarKC is an open-source software architecture available for downloading at SourceForge<sup>1</sup>. The contents of the newest LarKC-release contains the following components:

- Platform (/platform) - the core of LarKC, a software infrastructure and run-time environment that enables flexible development and serves a deployment environment for the LarKC plug-ins (see more details below).
- Exemplary collection of the LarKC Plug-Ins (/platform/plugins) - self-contained, loosely-coupled and functionally-interoperable modules grouped in the following categories (according to the functionality, please refer to the documentation for more details), including:
  - LLDRReasoner - executes a SPARQL query against Linked Life Data repository
  - SOStoVBtransformer - used for transformation of a SetOfStatements to the VariableBinding representation
- Extra tools
  - Workflow Designer - a tool for visual workflow construction (/extra\_tools/workflow\_designer)
  - Plug-in Wizard - a tool for creating new plug-ins in the Eclipse environment (/extra\_tools/eclipse\_wizard)
  - Archetypes for plug-in and end-point development with Maven (/extra\_tools/larkc-plugin[endpoint]-archetype)
- Exemplary workflows (/workflows)
  - LLD\_Reasoning - executes a SPARQL query against the Linked Life Data repository
  - User and developer manual (/doc)

Once the LarKC platform has been installed properly, it can be started by executing a special start-up script provided in the release package. In case of the successful start, the platform is ready for submission and execution of workflows. This is performed by means of a Management Interface – a web service based frontend to the LarKC platform (Fig. 2). In order to access the Management Interface, the user only needs to point the browser to the local link <http://localhost:8182/>. Using the Management Interface, the user can submit new workflows (see [1] for more information about the LarKC workflows and corresponding platform's services) and execute queries against the previously submitted workflows.

The user can execute one of the basic workflows provided with the LarKC installation, e.g. the LLD\_Reasoning one. This can be done by copying the annotated workflow's text into the Management Interface and submitting it. In case of the successfully workflow's deployment, the newly-submitted workflow should appear in the Management Interface's list of the submitted workflows.

---

<sup>1</sup> <http://sourceforge.net/projects/larkc/>

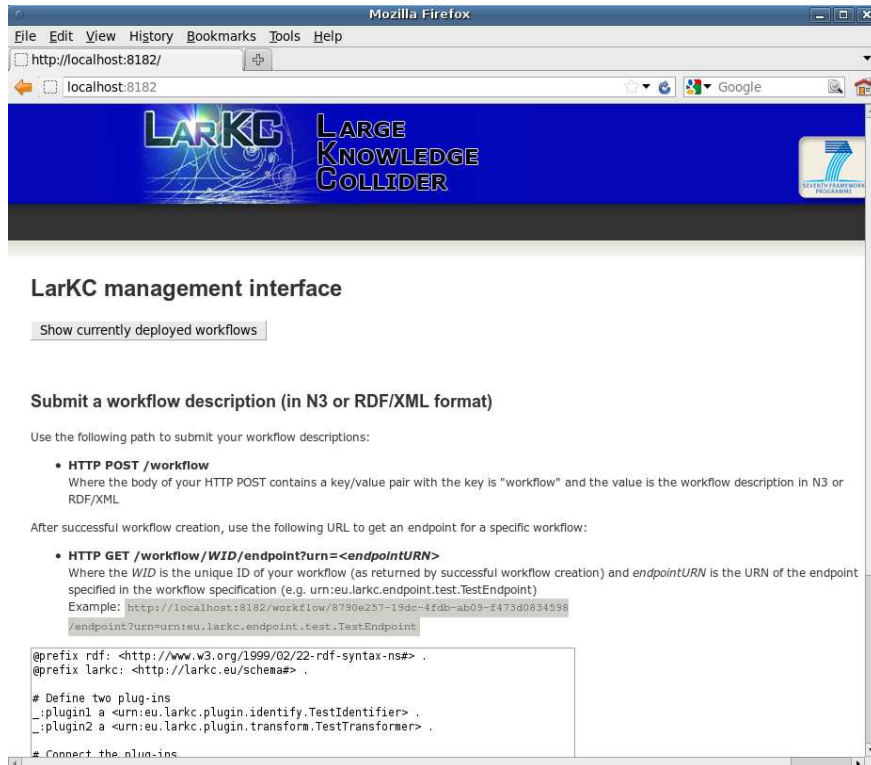


Fig. 2. LarKC Management Interface

## 4 Using The Workflow Designer Tool

Workflow Designer is a graphical front-end to LarKC. Workflow Designer is a middleware used on top of the management interface that allows users to construct, configure and execute workflows more efficiently. The designer is a JavaScript-based web application, offering an intuitive GUI for intelligent workflow design. Using Workflow Designer, the users can do the following:

- Retrieve the list of all available plug-ins from the plug-in registry, done through the platform's management interface (Fig. 3a).
- Retrieve the list of all available remote host templates, done through the platform's management interface (Fig. 3b).
- Create workflows by visually dragging and dropping LarKC plug-ins, specifying the dataflow dependencies (Fig. 3c).
- Easily and intuitively specify the deployment host for the plug-in (Fig. 3d).

- Convert the graphical workflow representation into RDF/XML or N3 formats, submit the created workflow description to the management interface, and instantiate the workflow within the platform.
- Choose the needed endpoint (Fig. 3e) and submit a query (e.g. SPARQL) to the chosen end-point (Fig. 3f).
- Visualize the query results in a separate window as an XML document (Fig. 3g).

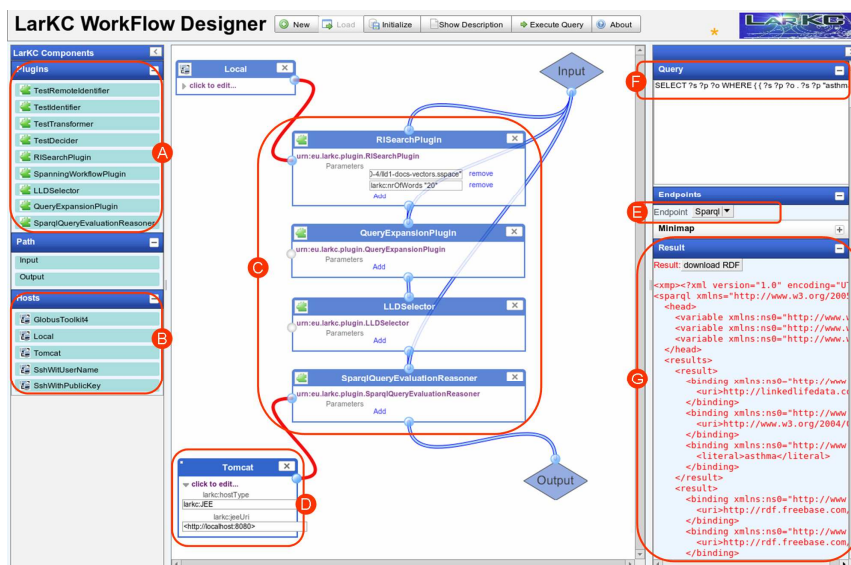


Fig. 3. The Workflow Designer's interface

## 5 Conclusion

The proposed demo is intended to give a brief introduction to the usage of LarKC. It guides the early adopter of the LarKC platform through all the steps needed to develop a simple, plug-in based reasoning application. It also provides an overview of the main LarKC front-ends, namely Management Interface and Workflow Designer.

## 6 References

1. Assel, M., Cheptsov, A., Gallizo, G., Celino, I., Dell'Aglio, D., Bradeško, L., Witbrock, M., Della Valle, E.: Large knowledge collider: a service-oriented platform for large-scale semantic reasoning. Proceedings of the International Conference on Web Intelligence, Mining and Semantics (WIMS'2011).