

# 9 - CBIIT Project Recommendations

Current Working Draft

- [Assessment Report](#)
- [Tools Recommended for Semantic Infrastructure Work](#)
  - [Triple Store Access](#)
  - [Semantic Knowledge Store](#)
  - [Data Conversion and Artifact Access](#)
  - [Integration Support](#)
  - [Inference, Rules and Expert Systems](#)
    - [RDF and OWL Tools](#)
    - [General Purpose](#)
  - [Flow Management for Services, Processes and web Applications](#)
  - [Design](#)
  - [Component Repository](#)

## Assessment Report

This section will provide the Assessment Report generated during the prototyping work of the inception phase.

## Tools Recommended for Semantic Infrastructure Work



### Additional Information

Supported CBIIT application structures, such as the Java platform, Tomcat, JBOSS, Ant, and Maven, are discussed in the architecture sections of this roadmap and the caGrid 2.0 Roadmap, [6 - caGrid 2.0 Architecture](#) and [6 - Semantic Infrastructure 2.0 Architecture](#). Additional standards are discussed in section [10.4 - CBIIT Project Recommendations](#) of the caGrid 2.0 Roadmap and on the [VCDE WS Standards Efforts](#) page.

The tools and libraries listed here have been identified as being of possible help in certain aspects of the Semantic Infrastructure 2.0 Roadmap project and development of the architecture. The tools listed below have not been formally described as supported by CBIIT at this time. However, they suggest the type of components and architecture expected to best satisfy project requirements.

## Triple Store Access

Access tools represent methods for interacting with information represented using Resource Description Framework (RDF) and Web Ontology Language (OWL) representations of metadata. In addition, these tools may provide additional support for inference engines.

**Jena ([sourceforge.net](http://sourceforge.net))** - The Jena Semantic Web Framework is a tool for building semantic web applications. It provides a programmatic environment for RDF, RDF Schema (RDFS) and OWL as well as a SPARQL engine and rule-based inference. This is a general purpose tool that supports combinations of models in memory as well as transactional persistence of triples. Jena works well with ontologies with large numbers of classes and individuals which require read and write functionality.

**OwlAPI ([sourceforge.net](http://sourceforge.net))** - The OWL API is a Java API for reference implementation for creating, manipulating and serializing OWL ontologies. This interface provides a fast, in-memory representation for manipulating OWL ontologies and provides persistence to OWL in XML format.

**Sesame ([opendrf.org](http://opendrf.org))** - Sesame is an open source RDF framework with support for RDF Schema inferencing and querying.

**ARQ ([openjena.org](http://openjena.org))** - Supports the W3C standard for RDF queries. This is packaged with Jena and can be used against Jena models.



### Additional Information

These systems can be used in combinations. For example, OWL files created with OwlAPI can be read by Jena. However, Jena TDB is specific to Jena, and you must use Jena to access it. See below for persistence methods.

## Semantic Knowledge Store

Stores represent the ability to store RDF and OWL representations. These tools provide various features including support for larger datasets, transactional updates, and integration with traditional relational databases. Although the tools listed below are open source, there are also non-open source tools which provide similar functionality including AlegraGraph and Oracle 11G. These system may also use Jena or Sesame for access.

**Jena TDB ([openjena.org](http://openjena.org))** - This is an integration between the Jena model factory which provides a high speed persistence of RDF triples. Writes to this system are not fully transactional, and so care must be taken to manage transactions externally, where required.

**Jena SDB ([openjena.org](http://openjena.org))** - Provides a persistence model using a variety of relational database management system (RDBMS) back ends. Utilizing JDBC connections, SDB persists into traditional tables the RDF triple information. This solution is fully transactional, but suffers from insert and query limitations for performance.

**Sesame SAILS ([opendrf.org](http://opendrf.org))** - Sesame provides a way to integrate various representations for access (called SAILS (Storage And Inference Layer)). Sesame SAILS have been created for in-memory, relational database, and a variety of other formats.

**OwLIM ([ontotext.com](http://ontotext.com))** - OwlLIM is implemented as a Sesame SAIL and comes in both an open source and commercial implementation. OwlLIM boasts that it is the most scaleable semantic repository in the world (the commercial implementation). It also offers a high speed reasoning system built into the system.

**db2rq ([wiwiss.fu-berline.de](http://wiwiss.fu-berline.de))** - Not technically a general purpose store, db2rq provides a semantic reference layer to an existing RDBMS environment, allowing for SPARQL and other interactions within a given environment.



#### Additional Information

These systems can also interact. Specifically Jena can access Sesame through a model factory component and Sesame can access Jena through a SAIL. This does not indicate compatibility, but rather an abstraction. This means that the database tables created by Jena SDB do not match the RDBMS tables generated by Sesame. However, as an example, code written using Jena as an interface, if written correctly, can be independent of the persistence method.

## Data Conversion and Artifact Access

These are tools which aid in the programmatic access to artifact sources to aid in transformations and processing.

**poi ([apache.org](http://apache.org))** - Poi is a general purpose tool for accessing Microsoft documents in the .DOC, .XLS, and traditional Microsoft proprietary formats. In addition Poi supports the openXML standard including Microsoft .DOCX and related formats through the openxml4j api. Information about this standard can be found [on openxml.org](http://openxml.org).

**OBO-Edit ([oboedit.org](http://oboedit.org))** - The editor provides an API for accessing OBO-based ontologies.

**Eclipse EMF ([eclipse.org](http://eclipse.org))** - The core component of Eclipse EMF is a tool for accessing models in XML for conversion to other representations. Other aspects of the Eclipse EMF may also be useful.

In addition there are standard access tools for accessing RDF and OWL representations. See [#Semantic Knowledge Store](#) above.

## Integration Support

**Spring ([springsource.org](http://springsource.org))** - Spring provides a number of components which are designed to either ease the adoption of new technologies, or to provide greater control over certain integrations. The number of Spring components is large; some significant components include Spring Framework, Spring Flow, Spring Web Services, and Spring Security. These components are all based on certain core patterns that make the components more flexible.

## Inference, Rules and Expert Systems

These are defined as a way to provide methods of representing and executing decision support, orchestration, analysis and many other aspects of application functionality. They share a way to represent certain behaviors for which a more concise language has been created than traditional programming languages. Some inference and rule systems support standards such as OWL DL (Description Logics), or RuleML (Markup Language), or RIF (Rule Interchange Format). In addition, there may be extensions or additional functionality which make them suitable.

**SILK ([silk.semwebcentral.org](http://silk.semwebcentral.org))** - SILK extends the expressive power of SPARQL, OWL-RL, RIF-BLD, and is designed with biomedical projects in mind.

**SQI-Prolog ([swi-prolog.org](http://swi-prolog.org))** - SQI-Prolog provides implementation of the Prolog language with support for RDF, OWL and SPARQL. Scalable to available memory and can be embedded in Java applications.

**DLVHEX ([kr.tuwien.ac.at/research/systems/dlvhex](http://kr.tuwien.ac.at/research/systems/dlvhex))** - DLVHEX is a prototype application for providing reasoning with OWL ontologies, with SPARQL plugin support.

## RDF and OWL Tools

**Pellet ([clarkparsia.com](http://clarkparsia.com))** - Pellet is an OWL 2 (partial) reasoner providing the core classification functionality. Pellet is broadly used and integrated into various platforms including Protege 4 and TopBraid Composer. Pellet is written directly in Java and so can easily be integrated into other java applications directly without external configurations or implementations.

**Fact++ ([owl.man.ac.uk](http://owl.man.ac.uk))** - Fact++ is an implementation of an OWL 2 (partial) reasoner written in C++. Fact++ requires the implementation of a component which is accessed by applications. Because it is written in C++ it has the potential to be faster than Java implementations.

**HermiT ([hermit-reasoner.com](http://hermit-reasoner.com))** - HermiT is an OWL 2 reasoner implementing a high performance algorithm in Java. It is dependent on the OwlAPI.

**TopBraid SPIN ([topquadrant.com](http://topquadrant.com))** - TopBraid SPIN is an implementation of the SPARQL Inferencing Notation. It is an open source implementation, and can be integrated in a number of ways. It has many uses including an RDF constraint language, a rules language, a SPARQL function language (used as a way to extend SPARQL), and a method of storing reusable queries. Envisioned and implemented by TopBraid, it expands functional behaviors in ways that are impossible to declare in DL, or where it would be inappropriate. Use of TopBraid SPIN requires the use of the Jena API.

## General Purpose

**Jess ([jessrules.com](http://jessrules.com))** - Jess stands for Java Expert System Shell. Jess is an implementation of the rete algorithm and supports a number of rule definition languages. It is the reference implementation of JSR 94 standard for java rule engines. Jess supports CLIPS (C Language Integrated Production System) and RuleML languages, as well as its own XML representation of CLIPS. Jess provides many ways to extend the functionality into Java Applications in both direction (able to call java functions from the rules, as well as call rule functionality from java). Jess is available without cost for academic uses as well as through various commercial licenses. It does not have a cost for development, as there is a trial download that times out after a number of days, and can be re-downloaded.

**Drools Expert (jboss.org)** - Drools is a component of the JBoss community. Drools is described as a business logic integration platform. It has a number of components which may be integrated to provide different support including a managed rule repository. Drools is an implementation of the rete algorithm. Drools supports a proprietary language as well as an XML representation of its own language. Transformations of RuleML to Drools may be available.

**Ontobroker (ontoprise.de)** - Ontobroker is a commercial package which provides support for high performance reasoning for W3C standards such as OWL, RIF, RDF(S), and SPARQL. Ontobroker integrates to multiple database systems and supports web service interfaces. Ontostudio provides a visual modeling tool for work with Ontobroker.

**FLORA-2 (flora.sourceforge.net)** - Flora is an object-oriented rule language and implementation of the RIF standard. It does not appear to be currently developed, but may have application here.

## Flow Management for Services, Processes and web Applications

**Open ODE (apache.org)** - Open ODE is an Apache project which utilizes the Web Services Business Process Execution Language (WS-BPEL) standard for organization of work flow. It is supported by Apache ServiceMix and can be used to manage web service choreography where that is appropriate.

**Drools Flow (jboss.org)** - Drools flow is an integration of the Drools rule engine designed to manage business or process flow. Drools Flow definitions can be rendered in the Business Process Modeling Notation (BPMN) notations, but an Eclipse plugin is also provided for visual design of workflows. Drools Flow also works with **Drools Guvnor** to provide a repository of workflows, and provides audit and control over workflow processes. Drools Flow has built in support to provide monitoring of flow activities. Drools Flow is an implementation of Business Process Modeling (BPM).

**Bonita Open Solution (bonitasoft.com)** - The Bonita Open Solution is an implementation of BPM that provides an environment for designing, managing and executing flow control. Flows in Bonita are designed graphically and can be executed directly through deployment as applications, or uploaded to a functional engine for execution. Bonita provides both an API for integration, and a web-based tool to provide execution points for state transitions and management.

**Spring Flow (springsource.org)** - Using an integration with the Spring MVC (model-view controller), Spring Flow allows for the definition of flows which control activity within a given session. This tool separates the page flow from the business logic, allowing for many alternative flows using the same pages. This simplifies applications which perform activities in different modes (create versus edit) or through different means (create "Wizards").

## Design

OWL and RDF provide the ability to represent information as metadata and as functional components of a system. As a result, individuals may produce RDF or OWL ontologies which will be integrated into the fabric of the system. In addition to the standard tools supported by CBIIT relating to design, the use of good ontology editors will help promote the consistency of representation and functionality. In some senses these are integrated development environments (IDEs) in the fact that development occurs; however, they can also be considered as Platform-specific Model (PSM) design tools because the output becomes a documentable model representation.

**Protégé 4 (stanford.edu)** - Protégé is developed and supported at Stanford University and has been used for ontology development for many years. Protégé 4 is an attempt to reach beyond the frame-based roots of Protégé and provide a newly envisioned representation of OWL ontologies. Protégé 4 utilizes the OwlAPI for accessing OWL ontologies and so shares the limitations of the OwlAPI. Specifically Protégé 4 does not support persistence to general purpose triple stores, and must be able to load the ontology entirely in memory. However, the use of the OwlAPI in design and editing where it is appropriate, gives Protégé 4 a performance advantage in the loading of ontologies, and provides unique functionality as related to ontology integration. Protégé 4 is an Eclipse Rich Client Platform (RCP) Application.

**TopBraid Composer (topquadrant.com)** - TopBraid Composer is available in both a community edition and a commercial license version. TopBraid Composer Community Edition provides an Eclipse plugin-based approach to ontology editing. This allows for the integration of other tools via the OSGI standard, and provides a basis for using other Eclipse-based tools. TopBraid Composer uses the Jena tool to access ontologies and shares its limitations. In addition, the community edition is limited to the editing of OWL or RDF files and does not support access of database stores. It does support SPARQL as part of its functionality, and utilizes the Eclipse approach of projects.



### Additional Information

Since there is no open source or community tool for accessing stores other than text files, many developers use Protege 4 or TopBraid Composer, and then create scripted or programmed solutions to upload models into those stores. However, there are other significant additions to functionality in the commercial versions of TopBraid Composer that are not addressed here.

## Component Repository

There are some current repository systems that may help in the management of elements such as rules and flow controls.

**Drools Guvnor (jboss.org)** - Drools Guvnor is a tool which provides access to a common rule repository, flow repository and other aspects of the Drools system, providing browsing and access control. In addition it integrates with graphical editor for rules and flows.