SMILA

(SeMantic Information Logistics Architecture)

Creation Review

Executive Summary

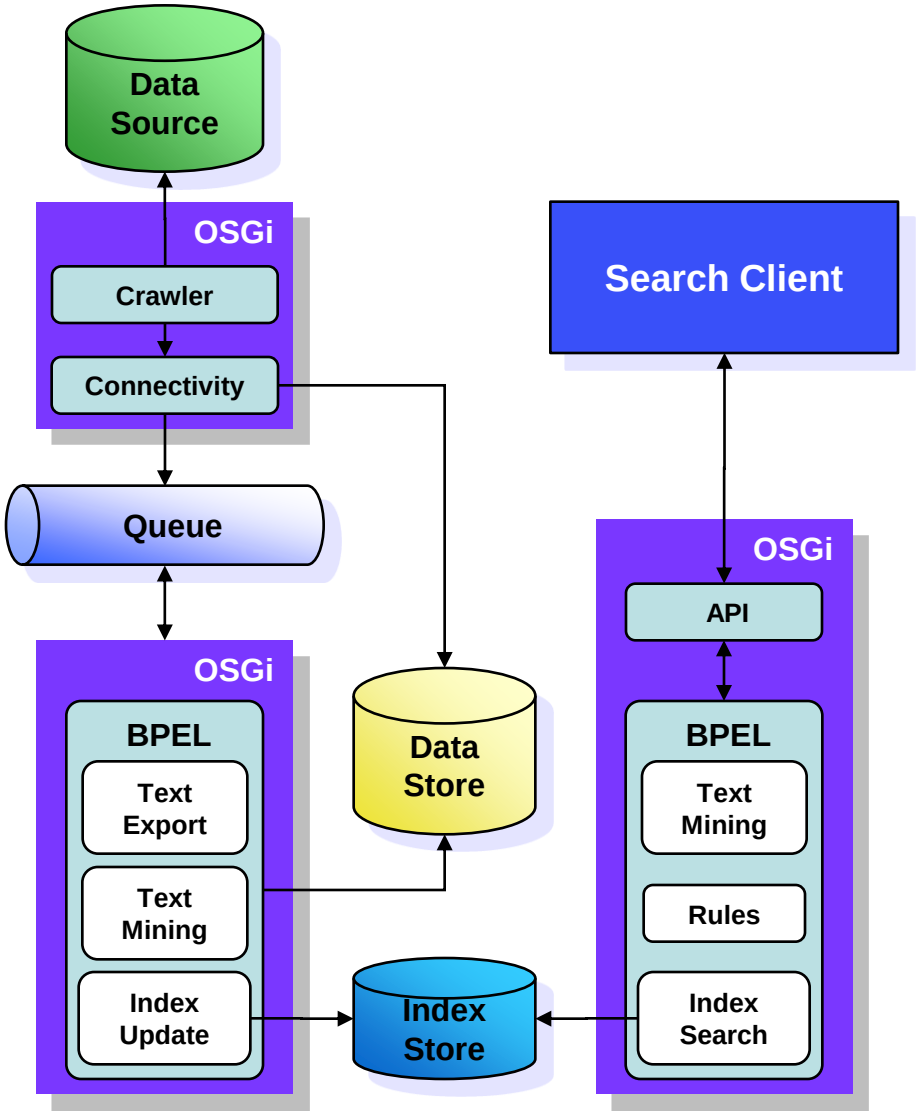
- The amount and diversity of information is growing exponentially, mainly in the area of unstructured data, like emails, text files, blogs, images etc.
- Poor data accessibility, user rights integration and the lack of semantic meta data are constraining factors for building next generation enterprise search and other document centric applications. Missing standards result in proprietary solutions with huge short and long term cost.
- SMILA is an extensible framework for building search solutions to access unstructured information in the enterprise. Besides providing essential infrastructure components and services, SMILA also delivers ready-to-use add-on components, like connectors to most relevant data sources.
- Using the framework as their basis will enable developers to concentrate on the creation of higher value solutions, like semantic driven applications etc.
- The long term goal of SMILA is to establish an industry standard by attracting as many parties as possible to use the framework and/or participate in the surrounding eco-system

» Unleashing the potential of unstructured data sources

Goals

- Define and implement an extensible framework based on SOA principles and standards (e.g. BPEL, SCA), which is dedicated to the access and integration of (unstructured) information
- Provide ready-to-use framework components (data source connectors and service implementations) that help to demonstrate and leverage its capabilities
- Deliver interfaces for management, operation and monitoring of the framework and its components

Architecture Overview



Indexing:

Crawler crawls the **Data source** and hands out the gathered data to the Connectivity module.

Connectivity pushes the information into the Queue server.

BPEL engine listens to the queue and consumes the messages.

BPEL services **Text Export** and **Text Mining** process the information stored in the message.

The service **Index Update** finally stores the document into the **Index Store**.

While processing the data all framework components and services can use the **Data Store** for persisting their data.

Search:

Search Client uses **API** to communicate with the framework.

The Query processing is done within the **BPEL** engine.

Finally the BPEL service **Index Search** returns a search result back to the **Search Client** via **API**.

Scope

- **Standardized and Scalable Architecture:** SMILA will deliver a highly scalable architecture for Information Access Management (IAM) processes based on well-established standards (e.g. BPEL, JMS, SNMP & JMX).
- **Componentization:** A major focus of SMILA will be on componentization of the overall system architecture, thus ensuring that other open source tools, products by different vendors or even project-specific extensions can easily be plugged into the system.
- **Exemplary implementation of vertical use cases** like search, classification, text extraction, text annotation and other semantic analysis functions
- **Data Source Management (integration and access):** The objective is to make available a set of connectors (crawlers) for the most relevant data sources (e.g. file system, database systems, web).
- **Management, Operation & Monitoring:** SMILA will provide interfaces to allow for system management, monitoring and operation of its components
- **Authentication and Authorization support:** The end-users can interact with the system only according to their actual access rights. This is not only true for accessing and storing the information but also for the process execution within the framework.
- **Status and performance reporting:** Analytics and business intelligence reporting are essential parts to any IAM system. The information provided by the system not only allows to optimize its usage but also to identify missing information via knowledge gap analysis and similar approaches.

Out of Scope

- SMILA will not conduct significant tooling effort
- SMILA will not develop generic management and monitoring solutions, but rather integrate into existing infrastructure
- It is not in the scope of SMILA to create a generic reporting/data rendering engine or business intelligence/data mining tools
- SMILA will not develop document indexing and retrieval components but rather allow for integration of such

Project Name and Position

- Full name: Semantic Information Logistics Architecture
- Short name: “SMILA”
- Motivation for change:
  - Original proposed name: “EILF” (Enterprise Information Logistics Framework) was not accepted by the community
    - » Difficult to pronounce
    - » Too long: Acronym should have at most 3 syllables
  - Suggestion from community to include “semantic web” in project name
- In contrast to the statement in the proposal, SMILA will be incubated below the Eclipse Runtime Project, not the Technology Project
- Implications of changes at eclipse.org
  - Newsgroup to be moved/renamed from eclipse.technology.eilf to eclipse.rt.smila
  - create a symbolic link /proposals/smila pointing to /proposals/eilf

Mentors

- Wayne Beaton (Eclipse Foundation)
  - Eclipse Evangelist
- Markus Knauer (INNOOPRACT GmbH)
  - Project Lead Eclipse Packaging Project, g-Eclipse

Initially participating parties

- empolis GmbH
- brox IT-Solutions GmbH

Initial Committers (1/2)

- August Georg Schmidt (brox IT-Solutions GmbH): Co-lead

  Georg is co-founder and CTO of brox IT-Solutions. He is leading the development department there since almost 10 years and he was the initiator of the EIF (Enterprise Information Framework), the virtual predecessor of SMILA

- Igor Novakovic (empolis GmbH): Co-lead

  Igor is a senior developer/architect with solid experience in JEE and server-side programming. He has been working in the past 8 years as application developer in the area of distributed client/server applications. Since 2006 he managed the development of the solution "empolis: Service Lifecycle Suite".

- Jürgen Schumacher (empolis GmbH): Committer

  Jürgen is a senior developer/architect with solid experience in JEE and server-side programming. He has been working in the past 9 years as application developer in the area of distributed client/server applications.

- Daniel Stucky (empolis GmbH): Committer

  Daniel is a senior developer/architect with solid experience in JEE and server-side programming. He has been working in the past 7 years as application developer in the area of distributed client/server applications.

Initial Committers (2/2)

- Thomas Menzel (brox IT-Solutions GmbH): Committer

  Thomas is a member of the SMILA team and acts as a technical lead for storage related subjects of the SMILA framework, in particular: XML- and binary storage. He has a strong background in relational databases, data structures, XML and has been a Java developer for almost 10 years, next to other languages. In the past he has worked as a developer, architect and project manager in several projects.

- Ralf Schumann (brox IT-Solutions GmbH): Committer

  Ralf is a member of the SMILA team and acts as a developer for storage related subjects of the SMILA framework, in particular: Binary Storage. He has a strong background in relational databases and data structures, Java developer for over 10 years, other languages as well. In the past he has worked as a developer and architect in several projects.

- Ralf Rausch (brox IT-Solutions GmbH): Committer

  Ralf is a member of the SMILA team with focuses on the build process, provisioning, test and documentation parts. He has been developing proofs of concepts (PoC) for the SMILA project and developing in Java for 4 years.

- Sebastian Voigt (brox IT-Solutions GmbH): Committer

  Sebastian is a member of the SMILA team. His activities have been focused on connectivity framework and BPEL process engine integration, both as a developer and architect. Sebastian is an experienced Java developer and architect with a strong background in distributed systems and security from his PhD thesis.

Code Contribution

Empolis GmbH and brox IT-solutions GmbH provide an initial code contribution

- No legacy code - all code is written from scratch
- Publication under EPL
- Desired namespace: org.eclipse.smila
- Also a large number of 3rd-party bundles will be contributed to Orbit

Community Response & Interested Parties

- Community response has been positive
- Interested parties
  - INNOOPRACT GmbH (Jochen Krause), DE
  - DFKI (German Institute for Artificial Intelligence), DE
  - Arexera Information Technologies GmbH, DE
  - University of Hildesheim, DE
  - SAP AG, DE
  - Aduna/Aperture (Aidministrator Nederland B.V.), NL
  - Applied Relevance LLC, US
  - Exalead S.A., FR
  - Coveo Solutions Inc., CA

Relationships to other Eclipse projects

SMILA intends to incorporate the results of several Eclipse projects:

- Equinox and presumably other “fellow” projects from the Eclipse Runtime Project, like Swordfish
- BPEL
- STP
- TPTP
- Higgins
- BIRT
- g-Eclipse

Tentative Plan

- **2008-07 Version 0.5 M0**
  - Basic architecture settled and implemented
  - Simple search application available
- **2008-09 Version 0.5 M1**
  - General configuration management
  - More data sources accessible
  - Advanced incremental update
- **2008-12 Version 1.0 – Release 1.0**
  - Cluster readiness
  - Implementation of the security concept
  - Annotation of metadata
  - Performance & monitoring tools