PureC

Project Startup Document

Authors:
Jakub Demczuk,
Mateusz Kaczmarek,
Maciej Więckowski,

Supervisor:
Sebastian R. Kruk

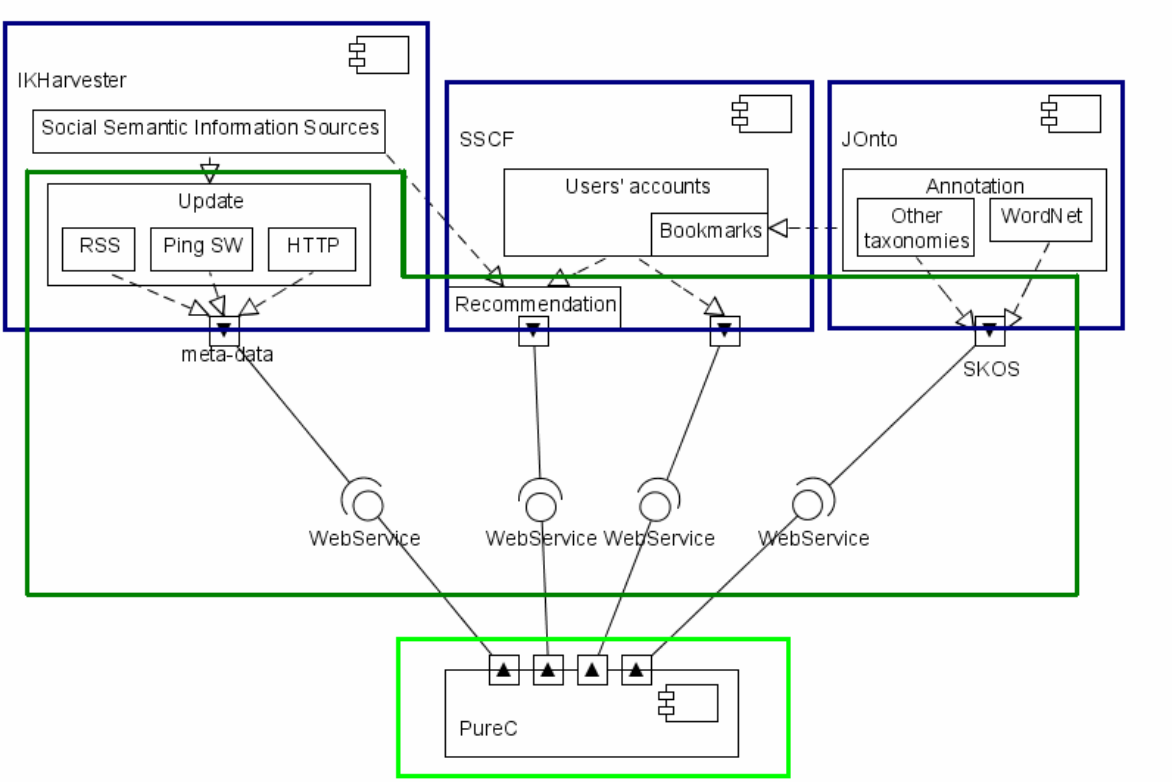
1. ABSTRACT

This document is intended to show how PureC application will make use of existing components which have already been implemented in various DERI projects and which are related to Semantic Web concept. It will be shown how these modules will work together as well as how they will extend their functionality with new features. This will be described in one general chapter concerning the architecture and four more detailed chapters about main functions.

2. PureC architecture

The picture below shows how three main components of PureC will cooperate together. All of them will be based on existing modules, namely IKHarvester[1], SSCF[2] and JOnto[3], each of them already implemented and tested in Jerome Digital Library[4] and notitio.us – social semantic bookmarking system. However, to meet PureC requirements their functionality will have to be expanded.

IKHarvester have been already used for harvesting data from particular sites (Wiki etc.) but it will have to use some other new sources of data and, what is probably most important, make use of its' own meta-data to describe them. This meta-data is yet to be thoroughly designed.



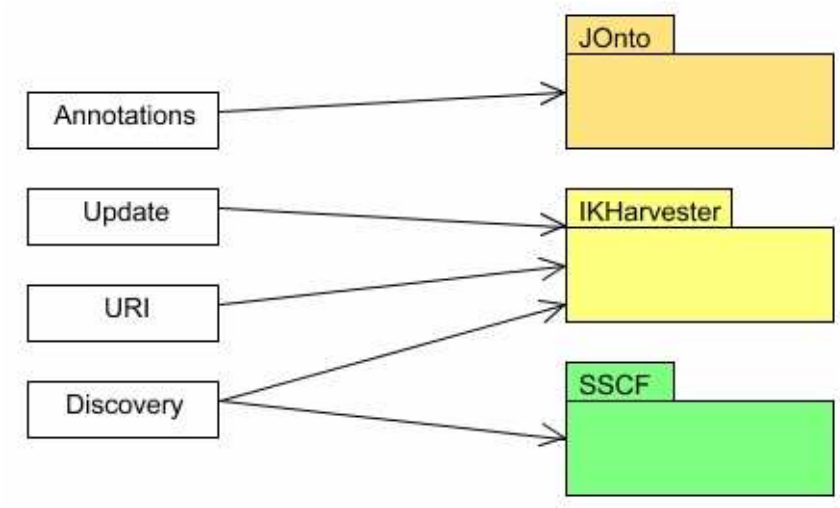
[1] http://wiki.corrib.org/index.php/Didaskon/IKHarvester
[2] http://s3b.corrib.org/
[3] http://sourceforge.net/projects/jonto
[4] http://www.jeromedl.org/

SSCF will provide its user management features, based on FOAFRealm[5] framework. Together with resources collected by IKHarvester it will be used in recommendation engine.

JOnto is a component for making annotations using different taxonomies. It will have to be redesigned to use standard SKOS notation (as described in ch. 4).

All of these components are intended to be able to work alone and to serve main PureC application through web services they will offer. PureC module will be responsible for putting this all together and presenting to the user with nice and easy interface.

In the following chapters, four main features of PureC will be described in more detail. Below you can see how these features are mapped to components mentioned before.



3. Detailed information about components used

Color frames used in the picture of PureC architecture shows which components are taken from DERI (blue), what will be added to them (dark green), and where is the "gluing" client application (bright green). Below we provide more information and describe the licensing and ownership issues of each of them.

- IKHarvester (Informal Knowledge Harvester) is a SOA layer which collects RDF data from web pages. It provides REST based Web Services for managing data available on Social Semantic Information Sources (SSIS): semantic blogs, semantic wikis and JeromeDL (the Social Semantic Digital Library). These Web Services allow saving harvested data in the informal knowledge repository, and providing them in a form of informal Learning Objects (LOs) that are described according to LOM (Learning Object Metadata) standard.

  It is a free Open Source project distributed under GPL licence and its code can be downloaded from Didaskon project SVN repository[6].

[5] http://wiki.corrib.org/index.php/FOAFRealm/
[6] https://didaskon.svn.sourceforge.net/svnroot/didaskon

- SSCF (Social Semantic Collaborative Filtering) is a part of S3B project led by DERI. SSCF is a type of collaborative filtering that makes use of existing, user maintained social network and semantic annotations. SSCF is based on the assumption that there is number of people that are experts on a particular subject, and it is possible to reach those people while traversing the graph of social network. The main difference between SSCF and classic collaborative filtering approach known from e.g. Amazon.com or Ringo, is that the social network is explicitly maintained by users themselves to solve the critical mass bootstrap problem.

  To improve a process of knowledge sharing users maintain their own classification schemata by annotating their bookmark folders with categorization taxonomies like DMoz or WordNet keywords. Users annotate resources with their classification schemata by adding resources to particular bookmark folders. User preferences are expressed by ranking values assigned to each folder-category created by user him/herself or imported from user's social network.

  SSCF is a free Open Source project, its license (BSD) is available on the Internet[7], as well as the source code (from S3B project SVN repository[8]).

- JOnto is actually the module used in SSCF for making annotations that were mentioned. JOnto delivers an open framework for utilizing various well established classification schemata like DMoz, WordNet or DDC. The aim is to make access to that ontologies as painless as possible - by cloaking all RDF related issues in meaningful java calls.

  JOnto is also free Open Source software distributed under the terms of the Apache Licence ver. 2.0[9] and can be downloaded from its own SVN repository[10].

- One thing that will be implemented is the code expanding functionality of the three components described above as well as web services as the ending points and interface for client application or (probably in the future) other systems. It will be a property of DERI distributed as free Open Source software which Tablane will make use of in PureC project.

- The client application finally is the user interface module which will be implemented as web application and will be used mainly for testing and presentation purposes. It will be a sole property od Tablane and its code will be private.

[7] http://s3b.corrib.org/index.php?option=com_content&task=view&id=25&Itemid=39
[8] https://s3b.svn.sourceforge.net/svnroot/s3b
[9] http://www.apache.org/licenses/
[10] https://jonto.svn.sourceforge.net/svnroot/jonto

4. URI

A standard notion, such as URI (Uniform Resource Identifier), should be created for the PureC project to uniquely identify and annotate fragments of information on the web (web clips, parts of multimedia files) to allow appliance of semantic technologies. No such way already exists so it will have to be investigated.

5. Annotations

PureC will allow users to annotate resources (that is whole created documents as well as web pages clips) with some taxonomies. It will make it possible to search them semantically in the future. For this purpose we will use JOnto component. JOnto delivers an open framework for utilizing various well established classification schemata like DMoz, WordNet or DDC. To make it more multi-purpose (that means managing more taxonomies) we will have to add SKOS (Simple Knowledge Organisation System[11]) handling.
SKOS is one standardized way to describe taxonomies, thesauruses, classification schemes etc. in RDF. Though it hasn't been announced an accomplished standard by W3C organization so far, it has already gained popularity in the Semantic Web domain. Implementing SKOS handling will make it possible to add new taxonomies in the course of time or maybe even user's taxonomies management in the runtime.

6. Discovery

Problem of discovering new sources and providing new source recommendations for users is one of the most important issues to deal with. Currently there are no means to provide such functionality in Tablane products. PureC will address this issue by using software currently under research in DERI. That is: Social Semantic Collaborative Filtering (SSCF) and IKHarvester. SSCF allows to create social networks of users by using FOAFRealm and expanding it with users' bookmarks and interests features. IKHarvester is a knowledge-base based on Learning Objects (see notitio.us).
Providing users with new recommended sources will be achieved in the following steps:

- Searching users' friends' bookmarks, created clips for resources matching users' interests (based on document tags, bookmarks, users FOAF profile etc)
- Browsing IKHarvester's knowledge-base for matching resources

This way every user will be given recommendations according to his/her profile and already created documents (clips, collections).

Since currently IKHarvester only supports Wiki-pages, SIOC described blogs, new supported formats like GRIDDL, microformats, blogger, DBPedia and SKOS (over SIOC) should be provided.

As for SSCF new SOA layer will be developed so that it will be accessible through Web-Services (based on REST architecture).

[11] http://www.w3.org/2004/02/skos/

7. Update

PureC should provide automated method for monitoring content used by users in their clips, allowing them to easily update their pages when the source page changes. Following ways to achieve that goal are proposed:

- RSS feeds – allows to easily monitor page content
- PingTheSemanticWeb.com[12] – "is a web service archiving the location of recently created/updated RDF documents on the Web"
- HTTP cache – using proxy like technique to monitor page content

Those mechanisms will provide a way to recognize update sources. Getting the update will be done by comparison of strings (in the prototype it will probably be a simple merge, but an interface for more advanced way will be provided).

PureC will provide a web service for end user application to get updates.

8. Use case scenario

Suppose there is a student called Sean who is collecting information necessary for writing his thesis. Sean is browsing many web sites, such as Wikis or blogs, in search of useful content. What's more, he has some friends, who are also students and are looking for similar information. The problem is that it is pretty uncomfortable to manage scraps found on web sites, keep them up-to-date and preserve references to them. Moreover, making use of Sean's friends' experience is also very limited – they can at most share the sources they have found by email or instant messengers.

With PureC, Sean will be able to easily create his own document composed of web sites scraps (let's call them webclips). Information about each webclip source will be saved by PureC and securely stored. What's even more important, PureC will regularly check if the source hasn't been changed and, if necessary, inform Sean about these changes to allow him to include them in his document. Of course, Sean will still be able to edit his document independently.

In the moment of creating new PureC document and importing webclips, Sean will be asked to annotate them using some predefined, well defined taxonomies. If other PureC users annotate their webclips (different from Sean's) similarly, the application will automatically prepare some recommendations. Based on "friendship level", which can be defined by users, these recommendations will be even more precise, assuming that Sean will be more interested in his friends' documents. This is where social semantic features of PureC will be used.

9. Summary

As one can see some development of DERI tools will have to be done to match PureC requirements. This prototype will not consider so called "deep internet" as a source of content to be monitored or recommended to users.

[12] http://pingthesemanticweb.com/about.php