Workflow Perspectives on the Open Semantic Framework

by Mike Bergman - Monday, April 25, 2011

http://www.mkbergman.com/956/workflow-perspectives-on-the-open-semantic-framework/

Advances in How to Transfer Semantic Technologies to Enterprise Users

For some time, our mantra at Structured Dynamics has been, "We're successful when we are not needed. [1]" In support of this vision, we have been key developers of an entire stack of semantic technologies useful to the enterprise, the open semantic framework (OSF); we have formulated and contributed significant open source deployment guidance to the MIKE2.0 methodology for semantic technologies in the enterprise called Open SEAS; we have developed useful structured data standards and ontologies; and we have made massive numbers of free how-to documents and images available for download on our TechWiki. Today, we add further to these contributions with our workflows guidance. All of these pieces contribute to what we call the total open solution. Prior documentation has described the overall architecture or layered approach of the open semantic framework (OSF). Those documents are useful, but lack a practical understanding of how the pieces fit together or how an OSF instance is developed and maintained. This new summary overviews a series of seven different workflows for various aspects of developing and maintaining an OSF (based on Drupal) [2]. In addition, each workflow section also cross-references other key documentation on the TechWiki, as well as points to possible tools that might be used for conducting each specific workflow.

Overview

Seven different workflows are described, as shown in the diagram below. Each of the workflows is color-coded and related to the other workflows. The basic interaction with an OSF instance tends to occur from left-to-right in the diagram, though the individual parts are not absolutely sequential. As each of the seven specific workflows is described below, it is keyed by the same color-coded portion of the overall workflow.
Each of the component workflows is itself described as a series of inter-relating activities or tasks.

**Installation Workflow**

Installation is mostly a one-time effort and proceeds in a more-or-less sequential basis. As various components of the stack are installed, they are then configured and tested for proper installation. The [installation guide](#) is the governing document for this process, with quite detailed scripts and configuration tests to follow. The blue bubbles in the diagram represent the major open source software components of [Virtuoso](#) (RDF triple store), [Solr](#) (full-text search) and [Drupal](#) (content management system).
Another portion of this workflow is to set up the tools for the backoffice access and management, such as PuTTY and WinSCP (among others).

Click here to see the tools associated with this workflow sequence, as described in the TechWiki desktop tools document.

**Configure & Presentation Workflow**

One of the most significant efforts in the overall OSF process is the configuration and theming of the host portal, generally based on Drupal. The three major clusters of effort in this workflow are the design of the portal, including a determination of its intended functionality; the setting of the content structure (stubbing of the site map) for the portal; and determining user groups and access rights. Each of these, in turn, is dependent on one or more plug-in modules to the Drupal system. Some of these modules are part of the conStruct series of OSF modules, and others are evaluated and drawn from the more than 8000 third-party plug-in modules to Drupal.
The **Design** aspect involves picking and then modifying a theme for the portal. These may start as one of the open source existing [Drupal themes](https://www.drupal.org/themes), as well as those more specifically recommended for OSF. If so, it will likely be necessary to do some minor layout modifications on the PHP code and some CSS (styling) changes. Theming (skinning) of the various semantic component widgets (see below) also occurs as part of this workflow. The **Content Structure** aspect involves defining and then stubbing out placeholders for eventual content. Think of this step as creating a site map structure for the OSF site, including major Drupal definitions for blocks, Views and menus. Some of the entity types are derived from the named entity dictionaries used by a given project. More complicated **User** assignments and groups are best handled through a module such as Drupal's [Organic Groups](https://www.drupal.org/project/organic_groups). In any event, determination of user groups...
such as anonymous, admins, curators, editors, etc.) is a necessary early determination, though these may be changed or modified over time. For site functionality, **Modules** must be evaluated and chosen to add to the core system. Some of these steps and their configuration settings are provided in the [guidelines for setting up Drupal](http://www.mkbergman.com) document. None of the initial decisions "lock in" eventual design and functionality. These may be modified at any time moving forward.

Click [here](http://www.mkbergman.com) to see the tools associated with this workflow sequence, as described in the [TechWiki desktop tools](http://www.mkbergman.com) document.

### Structured Data Workflow

Of course, a key aspect of any OSF instance is the access and management of structured data. There are basically two paths for getting structured data into the system. The first, involving (generally) smaller datasets is the manual conversion of the source data to one of the pre-configured OSF import formats of **RDF**, **JSON**, **XML**, or **CSV**. These are based on the [irON notation](http://www.mkbergman.com); a good [case study for using spreadsheets](http://www.mkbergman.com) is also available. The second path (bottom branch) is the conversion of internal structured data, often from a relational data store. Various converters and templates are available for these transformations. One excellent tool is **FME** from Safe Software (representing the example shown utilizing a spatial data infrastructure (SDI) data store), though a very large number of options exist for extract, transform and load. In the latter case, procedures for polling for updates, triggering notice of updates, and only extracting the deltas for the specific information changed can help reduce network traffic and upload/conversion/indexing times.
Click [here](#) to see the tools associated with this workflow sequence, as described in the TechWiki desktop tools document.

**Content Workflow**

The structured data from the prior workflow process is then matched with the remaining necessary content for the site. This content may be of any form and media (since all are supported by various Drupal modules), but, in general, the major emphasis is on text content. Existing text content may be imported to the portal or new content can be added via various WYSIWYG graphical editors for use within Drupal. (The excellent WYWIWYG Drupal module provides an access point to a variety of off-the-shelf, free WYSIWYG editors; we generally use TinyMCE but multiples can also be installed simultaneously). The intent of this workflow component is to complete content entry for the stubs earlier created during the configuration phase. It is also the component used for ongoing content additions to the site.

Content that is tagged by the scones tagger is done so based on the concepts in the domain ontology (see below) and the named entities (as contained in "dictionaries") used by a given project. Once tagged, this information can also now be related to the other structured data in the system. Once all of this various
content is entered into the system, it is then available for access and manipulation by the various conStruct modules (see figure above) and semantic component widgets (see below).

Click here to see the tools associated with this workflow sequence, as described in the TechWiki desktop tools document.

**Ontologies Workflow**

![Ontologies Workflow](image)

Though the next flowchart below appears rather complicated, there are really only three tasks that most OSF administrators need worry about with respect to ontologies:

1. Adding a concept to the domain ontology (a class) and setting its relationships to other concepts
2. Adding a dataset attribute (data characteristic) for various dataset records, or
3. Adding or changing an annotation for either of these things, such as the labels or descriptions of the thing.

In actuality, of course, editing, modifying or deleting existing information is also important, but they are easier subsets of activities and user interfaces to the basic add ("create") functions. The OSF interface provides three clean user interfaces to these three basic activities [3]. These basic activities may be applied to the three major governing ontologies in any OSF installation:

- The domain ontology, which captures the conceptual description of the instance's domain space
- The semantic components ontology (SCO), which sets what widgets may display what kinds of data, and
- irON for the instance record attributes and metadata (annotations).

All of the OSF ontology tools work off of the OWLAPI as the intermediary access point. The ontologies themselves are indexed as structured data (RDF with Virtuoso) or full text (Solr) for various search, retrieval and reasoning activities.
Because of the central use of the OWLAPI, it is also possible to use the Protégé editor/IDE environment against the ontologies, which also provides reasoners and consistency checking.

Click here to see the tools associated with this workflow sequence, as described in the TechWiki desktop tools document.

**Filter & Select Workflow**

The filter and select activities are driven by user interaction, with no additional admin tools required. This workflow is actually the culmination of all of the previous sequences in that it exposes the structured data to users, enables them to slice-and-dice it, and then to view it with a choice of relevant widgets (semantic components). For example, see this animation:
Considerable more detail and explanation is available for these semantic components.

Click here to see the tools associated with this workflow sequence, as described in the TechWiki desktop tools document.

**Maintenance Workflow**

The ongoing maintenance of an OSF instance is mostly a standard Drupal activity. Major activities that may occur include moderating comments; rotating or adding new content; managing users; and continued documentation of the site for internal tech transfer and training. If the portal embraces other aspects of community engagement (social media), these need to be handled as part of this workflow as well. All aspects of the site and its constituent data may be changed, or added to at any time.
Click [here](#) to see the tools associated with this workflow sequence, as described in the TechWiki desktop tools document.

**Moving from Here**

When first introduced in our [three-part series](#), we noted the interlocking pieces that constituted the *total open solution* of the open semantic framework (OSF) (see right). We also made the point -- unfortunately still true today -- that the relative maturity and completeness of all of these components still does not allow us to achieve fully, "We're successful when we are not needed." As a small firm that is committed to self-funding via revenues, Structured Dynamics is only able to add to its stable of open source software and to develop methodologies and provide documentation based on our client support. Yet, despite our smallness, our superb client support has enabled us to aggressively and rapidly add to all four components of this total open solution. This newest
series of ongoing workflow documents (plus some very significant expansions and refinements of the OSF code base) is merely the latest example of this dynamic. Through judicious picking of clients (and vice versa), and our insistence that new work and documentation be open sourced because it itself has benefitted from prior open source, we and our client partners have been making steady progress to this vision of enterprises being able to adopt and install semantic solutions on their own. Inch-by-inch we are getting there. The status of our vision today is that we are still needed in most cases to help formulate the implementation plan and then guide the initial set-up and configuration of the OSF. This support typically includes ontology development, data conversion and overall component integration. While it is true that some parties have embraced the OSF code and documentation and are implementing solutions on their own, this still requires considerable commitment and knowledge and skills in semantic technologies. The great news about today's status is that -- after initial set-up and configuration -- we are now able to transfer the technology to the client and walk away. Tools, documentation, procedures and workflows are now adequate for the client to extend and maintain their OSF instance on their own. This great news includes a certification process and program for transferring the technology to client staff and assessing their proficiency in using it. We have been completely open about our plans and our status. In our commitment to our vision of success, much work is still needed on the initial install and configure steps and on the entire area of ontology creation, extension and mapping [4]. We are working hard to bridge these gaps. We welcome additional partners that share with us the vision of complete, turnkey frameworks -- including all aspects of total open solutions. Inch-by-inch we are approaching the realization of a vision that will fundamentally change how every enterprise can leverage its existing information assets to deliver competitive advantage and greater value for all stakeholders. You are welcome aboard!

[1] This has been the thematic message on Structured Dynamics' Web site for at least two years. The basic idea is to look at open source semantic technologies from the perspective of the enterprise customer, and then to deliver all necessary pieces to enable that customer to install, deploy and maintain the OSF stack on its own. The sentiment has infused our overall approach to technology development, documentation, technology transfer and attention to methodologies.

[2] The first version of this article appeared as Workflow Perspectives on OSF on the OpenStructs TechWiki on April 19, 2011.

[3] The current release of OSF does not yet have these components included; they will be released to the open source SVN’s by early summer.

[4] The best summary of the vision for where ontology development needs to head is provided by the Normative Landscape of Ontology Tools article on the TechWiki; see especially the second figure in that document.