Semantic Annotation and Search for resources in the next Generation Web with SA-REST

Ajith Ranabahu, Amit Sheth, Maryam Panahiazar, Sanjaya Wijeratne
Ohio Center of Excellence in Knowledge-enabled Computing (Kno.e.sis)
Wright State University, Dayton OH

Abstract

SA-REST, the W3C member submission, can be used for supporting a wide variety of Plain Old Semantic HTML (POSH) annotation capabilities on any type of Web resource. Kino framework and tools provide support of capabilities to realize SA-REST's promised value. These tools include (a) a browser-plugin to support annotation of a Web resource (including services) with respect to an ontology, domain model or vocabulary, (b) an annotation aware indexing engine and (c) faceted search and selection of the Web resources. At one end of the spectrum, we present Kino^{E} (aka Kino for Enterprise) which uses NCBO formal ontologies and associated services for searching ontologies and mappings, for annotating RESTful services and Web APIs, which are then used to support faceted search. At another end of the spectrum, we present Kino^{W} (aka Kino for the Web), capable of adding SA-REST or Microdata annotations to Web pages, using Schema.org as a model and Linked Open Data (LOD) as a knowledge base. We also present two use cases based on Kino^{E} and the benefits to data and service integration enabled through this annotation approach.

Introduction

The Web has evolved far beyond a collection of hyperlinked documents. The Web now serves a variety of rich content, enabled by contributions from the public at large and most of all has become programmable. Many of the human focused functions on the Web are also present via services, enabling a variety of applications to be built composed of these services. In this context, also taking the historical perspectives into consideration, we make the following observations.

1. SOAP based Web services has only become successful (and continue to evolve) in the enterprise space. The model of public service registries has not become successful (IBM, Microsoft and SAP has taken down their public UDDI registries); although private registries are being used in an intra-enterprise context. Public SOAP based Web services simply publish their WSDL files in the Web and let generic search engines index them or list their WSDLs in specialized service catalogs such as BioCatalogue (http://www.biocatalogue.org/). Although there are annotation schemes for formal service definitions (such as SAWSDL), doing so requires deep domain knowledge from individual annotators. Because of this, the rate at which existing services are annotated lags well behind the rate of development of new services.
2. RESTful services have become far more popular in the consumer space. Given that the guiding principles for RESTful services are far less rigid than SOAP services, many RESTful services lack a formal description. For service compositions, the developers simply perform keyword search in a general purpose search engine, read the documentation and perform the necessary programming, often modifying the example code fragments. Data interoperability and exchange becomes a difficult one-off exercise with little or no tooling support.

These observations lead us to anticipate the following trends in the future.

1. SOAP services will continue to be main stream in the enterprise space. It is unlikely for the SOAP services to overtake the RESTful services. This implies that any advancement we do in the SOAP Web services context are only going to affect a limited audience.
2. The consumption model for RESTful services is likely to remain as it is. The communities that interact with RESTful service interfaces primarily consist of programmers and they value the flexibility in

programming. Several tools that emerged to streamline the RESTful service composition process failed to meet their expectations and are no longer available (Google Mashup Editor, Microsoft Popfly), hinting on the possible continuance of the current pattern of service consumption.
3. General purpose search engines will be the most likely avenue that developers find their services. The fact that there is no explicit service registration process (the search engine crawlers find the Website without explicit inputs from the author) and the high availability of general purpose search engines would be the primary reasons for this trend. Prototype systems such as APIhut[1] have demonstrated that specialized indexing have benefits in service lookup but such approaches have not taken off in the open Web yet.

The anticipation of these trends prompts us to suggest that **modification of service descriptions via annotations is the best way to supplement the upcoming service consumption patterns**. This not only supports better indexing by general purpose search engines, but also provides the ability to support automation by gleaning a formal description of the human readable text. **Our philosophy is that given the strongly linked nature of the Web, disparate approaches that focus specifically on services or data is not sufficient to address the problem of data and service interoperability in its entirety**. Instead the focus should be on a universally applicable scheme that works for all forms of Web resources.

SA-REST for Service Annotation

SA-REST, a W3C member submission for Web resource annotations[2] is a compatible annotation scheme with the other schemes such as Microdata and RDFa[3] and applicable to many other domains, besides services. The Kino Enterprise project[4] makes heavy use of the SA-REST annotations to build a generic scientific document annotation and indexing system.

The flexibility of using SA-REST (or other equivalent form of annotation) comes in three forms:
1. The human readable descriptions do not need to be changed. Instead, annotations can be embedded in the documents and capable tools can exploit these annotations to extract formal descriptions or introduce faceting.
2. Annotations can be mixed, pointing to a variety of domains. This allows one to run sophisticated search queries, mixing domain terms and service terms. For example, a query can be formed to ask for all services that include the concept ‘event’ (as described in schema.org) as an input. At present, such queries can only be supported by custom indexing engines, capable of parsing and using the annotations such as the Kino back-end.
3. When search engine supported annotations schemes (such as Microdata) are used, mixed domain queries can even be performed over a general purpose search engine (provided that the search query mechanism supports filtering by annotations). This is an excellent way to discover services. Using the Web APIs provided by search engines, one can even build specialized service discovery engines.

[1] Karthik Gomadam, Ajith Ranabahu, Meenakshi Nagarajan, Amit. P. Sheth and Kunal Verma, 'A Faceted Classification Based Approach to Search and Rank Web APIs', In Proc of 6th IEEE Intl Conf on Web Services (ICWS), pp 177-184, Beijing, China, Sep. 2008. http://knoesis.org/research/srl/projects/apihut/
[2] http://www.w3.org/Submission/SA-REST/
[3] http://blog.ranabahu.org/2011/06/sa-rest-and-schemaorg-friend-not-foe.html
[4] http://wiki.knoesis.org/index.php/Kino

A Generic Architecture for Collecting, Enhancing and Utilizing Annotations

Figure 1 illustrates a generic architecture for collecting, enhancing and using annotations to retrieve any type of documents or Web resources, applicable to many domains.

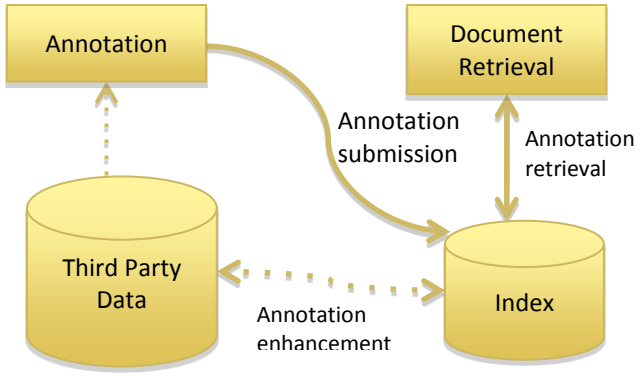

Figure 1 : A Generic architecture for managing annotations. The annotation process is performed in consulation with third part data sources and submitted (or collected) by the index. These are later used during a retrieval system such as a one supported faceted/semantic search.

Kino^{E}

The Kino Enterprise edition (formerly known simply as Kino[5]) uses SA-REST as the annotations mechanism and uses Apache SOLR as the faceting indexing engine. Kino^{E} annotation tools are bound to the bioportal managed by the National Center of Biomedical ontology (NCBO) and show the matching concepts for the annotator to select.

The Kino annotation component is a browser plugin, based on the Firefox browser. When the user highlights and right clicks on a word or a phrase, the browsers context menu includes the ***annotate as biological concept*** menu item. Selecting this menu item brings up the annotations window where the highlighted term is searched using NCBO RESTful API and a detailed view of the available ontological terms is shown to the user to select. The user can search or browse for a concept in any of the 300 or so ontologies hosted by NCBO's bioportal. Once all the annotations are added, users can directly submit the annotations to a predefined (configurable through an options dialog) Kino instance, by selecting the publish annotations menu item.

Kino^{E} components are organized following the generic design in Figure 1, as illustrated in Figure 2. It is freely available to the public as open source software[6].

Use cases

1. Biology oriented Web service annotation and retrieval

Many organizations, such as the DNA Data Bank of Japan (DDBJ), provide service interfaces for biological data retrieval tasks (such as Gene prediction). Biologists typically search and browse through a service catalog such as BioCatalogue and import the relevant service descriptions to a composer tool. Biologist would have to use descriptive terms to extract the most suitable services. Often these terms are imprecise, and several attempts are needed to get to the exact service required for the task at hand.

2. Scientific Document annotation

The scientists at Sanger Institute perform document annotation through a labor intensive process. This process is illustrated in detail online at the Kino evaluations page[7].

A detailed empirical evaluation is also present online at the Kino evaluations page.

---

[5] Kino is a character of deep diver who finds a big pearl in a lake in a celebrated novel "*The Pearl".*

[6] Download at: http://wiki.knoesis.org/index.php/Kino
2 min video: http://www.youtube.com/watch?v=MAQfCHwthI0
10 min video: http://www.youtube.com/watch?v=12R81HrlAF8

[7] http://wiki.knoesis.org/index.php/Kino_evaluations

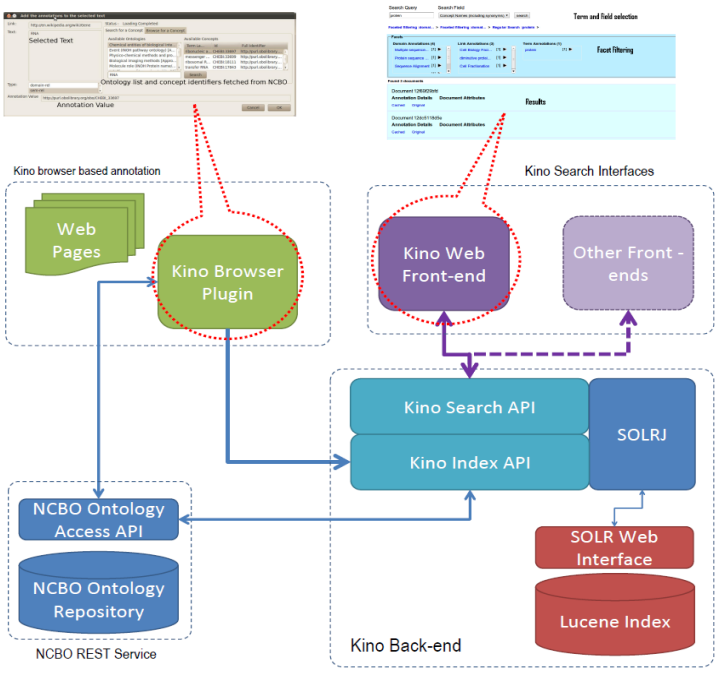

Figure 2 : Kino Enterprise edition (Kino^{E}) architecture. The browser plugin attaches annotations to the Web page source and submits them to the Kino indexing engine. These annotations are then enhanced using NCBO bioportal services during the indexing process.

Kino^{w}

Kino^{w} is a generic version of Kino, applicable to a wide variety of domains, going beyond life sciences, and suitable for use for a variety of Web resource (we will demonstrate Kino^{w} during the workshop). The experience from Kino^{E} indicated that a standard-driven annotation and indexing tool set is indeed applicable across multiple scientific domains. We believe that the same system can be applied to service descriptions to streamline existing annotation tasks, facilitate the use of existing ontologies and de facto representation models such as schema.org with grounding in existing community created knowledge/fact bases such as LOD.

Figure 3 outlines the architecture of Kino^{w}, applicable to resource annotation and retrieval. The browser plugin is modified to update Web content (via custom plugins to popular content management systems such as Drupal or Mediawiki). The site content is eventually crawled by annotation aware search engines (unless search engines provide APIs in future to submit/register annotations). The users can now direct targeted queries towards the search engines to filter by annotations to retrieve the relevant services.

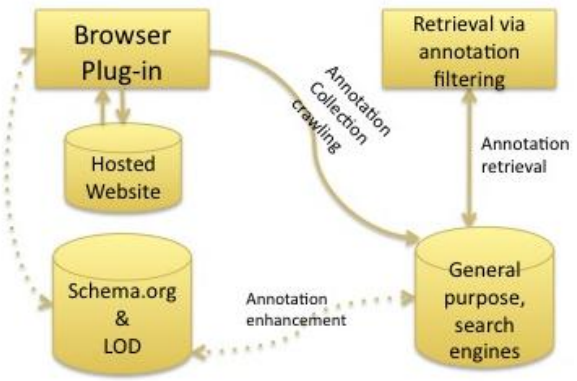

Figure 3 : Suggested architecture for Kino Web edition. The major search engines do not have submission APIs, hence the original Web pages need to be changed to let the generic search engines crawl the Web data.

A very simple example of using Kino[w] is that of annotating location “Dayton” on **http://knoesis.org/aboutus/contactus** with location concept in schema.org that is grounded in Geonames for Dayton in the LOD. The following is a specific example that uses the Flickr upload service description to illustrate the service annotations added by Kino[w]. The full example is available at http://wiki.knoesis.org/index.php/Kino_web. Listing 1 illustrates the mixed annotations made using SA-REST, highlighting the service parameters as well as the schema.org entities.

```
<!-- ########################################################## -->
 <div class="domain-rel" title="sarest:InputMessage">
       <h3>Arguments</h3>
<dl>
               <dt class="domain-rel" title="sarest:Parameter"><code class="domain-rel"
class="http://schema.org/Photograph">photo</code></dt>
               <dd>The file to upload.</dd>
               <dt class="domain-rel" title="sarest:Parameter"><code>title</code>
(optional)</dt>
               <dd>The title of the photo.</dd>
               <dt class="domain-rel" title="sarest:Parameter"><code>description</code>
(optional)</dt>
               <dd>A description of the photo. May contain some limited HTML.</dd>
               <dt class="domain-rel" title="sarest:Parameter"><code>tags</code>
(optional)</dt>
               <dd>A space-seperated list of tags to apply to the photo.</dd>
               <dt class="domain-rel" title="sarest:Parameter"><code>is_public, is_friend,
is_family</code> (optional)</dt>
               <dd>Set to 0 for no, 1 for yes. Specifies who can view the photo.</dd>
               <dt class="domain-rel" title="sarest:Parameter"><code>safety_level</code>
(optional)</dt>
               <dd>Set to 1 for Safe, 2 for Moderate, or 3 for Restricted.</dd>
               <dt class="domain-rel" title="sarest:Parameter"><code>content_type</code>
(optional)</dt>
               <dd>Set to 1 for Photo, 2 for Screenshot, or 3 for Other.</dd>
               <dt class="domain-rel" title="sarest:Parameter"><code>hidden</code>
(optional)</dt>
               <dd>Set to 1 to keep the photo in global search results, 2 to hide from
public searches.</dd>
```

Listing 1: Annotations embedded in the Flickr upload API page (http://www.flickr.com/services/api/upload.api.html), using SA-REST. The lines in bold illustrate the annotations that point to schema.org entities as well as the knoesis SA-REST ontology, published in the SA-REST specification.

Conclusion

The Web has evolved from a collection of linked documents to a fully programmable, service oriented information hub. The anticipated usage patterns of the Web likely point towards an extensive use of semantic annotations and a higher dependence on general purpose search engines that utilize these semantic annotations. We believe that SA-REST is a good option that enables flexible semantic annotations that can either use formal ontologies or light weight community developed schema and knowledge bases. This can be used to annotate most forms of Web resources varying from simple Web pages and media files to Web APIs and services. We present the Kino project as an example of architecture and tooling that demonstrate the viability of this approach. Extensive work in data (e.g. ontology alignment) and service interoperability complement this approach to address additional interoperability challenges.