9-2011

Kino: A Generic Document Management System for Biologists using SA-REST and Faceted Search

Ajith Harshana Ranabahu
Wright State University - Main Campus

Priti Parikh
Wright State University - Main Campus, priti.parikh@wright.edu

Maryam Panahiazar
Wright State University - Main Campus

Amit P. Sheth
Wright State University - Main Campus, amit@sc.edu

Kino : A Generic Document Management System for Biologists Using SA-REST and Faceted Search

Ajith Ranabahu *, Priti Parikh*, Maryam Panahiazar*, Amit Sheth* and Flora Logan-Klumpler†‡
*Ohio Center of Excellence in Knowledge-enabled Computing (Kno.e.sis)
Wright State University
Dayton OH, USA
ajith,priti,mary,amit@knoesis.org
† The Wellcome Trust Sanger Institute,
Wellcome Trust Genome Campus,
Hinxton, Cambridge CB10 1SA, UK
fl2@sanger.ac.uk
‡ Department of Biochemistry,
Tennis Court Road,
Cambridge CB2 1QW, UK,

***Abstract*—Document management has become an important consideration for the scientific community over the last decade. Human knowledge is central to many scientific domains, thus it is not possible to completely automate the document management process. Managing scientific documents require a semi-automatic approach to overcome issues of large volume, yet support the human participation in the process.**

In this paper we present Kino, a set of tools that streamline the document management process in life science domains. Kino is integrated with National Center for Biomedical Ontology (NCBO), providing scientists access to quality domain models. Annotated documents are indexed using a faceted indexing and search engine that provides fine grained search capabilities to the scientists. We present two use cases that highlight the pain points in managing scientific literature and also include an empirical evaluation.

***Keywords*-Biological Literature Annotations, Semantic document management, SA-REST, Kino**

I. Introduction

Document management has become a significant consideration for many scientific domains. The rate of document accumulation is far more rapid than a decade ago, and many scientists struggle with managing an enormous influx of data and documents relevant to their research and experiments.

The fundamental issue in managing scientific documents is the extreme volume. Tremendous volume of the scientific documents and/or literature makes manual organization impractical. This situation is further complicated by the presence of multiple formats due to poor standardization.

Many researchers have focused their energy on solving the case of extreme document volumes by implementing improved search and indexing techniques for scientific literature. The need for formal domain modeling is identified as an important prerequisite and as a result, major developments have been made in establishing formal models to represent scientific domains. In an increasing number of cases, these models are ontologies, usually defined in Web Ontology Language (OWL), the widely adopted W3C standard for ontology syntax. An ontology features relationships as first class objects, enabling rich modeling capabilities. Life sciences have been one of the domains to see early development of comprehensive ontologies. Coverage of current life science ontologies range from generic scientific terms (NCI Thesaurus[1]) to highly specific life cycle of parasites (Ontology for Parasite Life cycle (OPL)[2]).

Ontologies are useful as the guiding knowledge bases to implement advanced, domain specific, document management systems. The fundamental driving principle in using ontologies for document management is *annotation*. Annotation refers to embedding labels pointing to ontologies (or other models). The exact syntax of an annotation depends on the format of the document being annotated.

Using accurate annotations pointing to even a single ontology can improve the quality of lookups in a scientific document management system dramatically. This is exemplified by the experiences of the Gene Ontology (GO) [1]. Literature annotations using GO terms produce very high quality, species-specific meta-data and brings the information about the gene product into a format that can easily be used further in high-throughput experiments. Thus, annotation of scientific literature is an extremely worthwhile process in the long term. However, complete automation of the annotation process is often not practical or possible, due to the presence of contextual and domain specific details and the need for deep domain knowledge. Hence annotation of scientific literature still remains a human-oriented task.

[1]http://www.obofoundry.org/cgi-bin/detail.cgi?id=ncithesaurus
[2]http://wiki.knoesis.org/index.php/Parasite_Life_Cycle_ontology

Lack of good tools and integration has hampered the use of ontologies in many systems. For example, NCBO[3] currently hosts around 260 ontologies containing nearly 5 million terms. Although these ontological terms are the ideal candidates for annotations, biologists hesitate to look for standard ontological terms given that it's a time consuming process. Only a handful of life science ontologies, such as GO, have seen wide adoption. Furthermore, many existing systems follow different annotation methods and custom workflows that lack standardization. The requirement for a standards driven, well integrated suit of tools, has become obvious for biologists.

The goal of this research is to combine SA-REST [2], a W3C member submission that specifies a general purpose Web resource annotation framework; and a faceted indexing and search engine to create a generic annotation and indexing mechanism for biology-oriented documents. We focus on better integration, as well as the use of standards where applicable. Our intention is to provide biologists with convenient tooling to overcome the issue of large volumes as well as the presence of multiple formats to some extent.

Thus, our contributions are:

1) A comprehensive architecture for annotating and indexing biology oriented documents enabling; faceted search, based on existing ontological concepts.
2) Two practical use cases that address different document management problems in a biological context; and, demonstrate the advantages of the proposed architecture using these two use cases.
3) Kino toolkit, highlighting two key components, that facilitate the annotation and indexing process.

We deliberately did not perform a system level evaluation of Kino for two reasons. The first reason is that existing systems, such as Biocatalogue[4], represent significant contributions from the community over several years. It is extremely difficult to collect a similar data set without considerable time and effort. The second reason is the extra complexity introduced into the faceting aspect by the use of multiple ontologies. It would be unfair to do a system level comparison (such as a performance or storage requirements), due to the difference in the underlying system assumptions.

Hence we performed an empirical evaluation, highlighting specific cases where our system shows clear advantage over the existing ones. We opted to release Kino tools to the public, and plan to collect the experiences from the adopters at large.

The rest of this paper is organized as follows. Section II discusses the background, and Section III presents our motivating use cases. Section IV discusses in detail the system architecture, the tools, and their functionality; followed by the empirical evaluation in Section V. Section VI describes the relevant related work.We conclude with a discussion of the future work in Section VII.

[3]http://www.bioontology.org/

[4]http://www.biocatalogue.org/

II. Background

We organize the pertinent background work in two sections. Section II-A provides details on SA-REST, the selected annotation framework. Section II-B covers the details of the Kino faceted indexing and search framework.

A. SA-REST

SA-REST is a Plain Old Semantic HTML (POSH) format to add additional meta-data to (but not limited to) REST API descriptions in HTML or XHTML [3], [2]. Being *POSH* means that the embedded annotations are similar in nature to Microformats, but may not necessarily have gone through a rigorous open community process.

SA-REST is flexible enough to use meta-data from different models such an ontology, taxonomy, or a tag cloud. This embedded meta-data permits various enhancements, such as improve search, facilitate data mediation, and provide easier integration of services.

SA-REST has three *properties* (types of annotations) that can be applied to an XHTML document.

1) **domain-rel**: This property allows a domain information description of a resource. If a given resource has content spanning multiple domains, it may be necessary to add multiple domain-rel property entries, each corresponding to a section of the resource. If such a separation cannot be made, then the resource may be attached with an enumeration of values as the domain-rel property value.
2) **sem-rel**: The sem-rel property captures the semantics of a link, and evolves from the popular *rel* tag. This property enables the addition of externalized annotations to third party documents. A sem-rel property may only be used with an anchor ($<$ a $>$) element.
3) **sem-class**: This property can be used to markup a single entity within a resource. The entity may be a term, a text fragment or embedded objects such as a video.

B. Faceted Indexing and Search

The faceted indexing and search engine is called Kino, referring to the talented young pearl diver mentioned in John Steinbeck's novel *Pearl*[5]. Kino is the descendant of the APIhut project that introduced faceted indexing and search capability to service descriptions [4]. Kino supports generic domain annotations, and is capable of providing facets on any domain. Kino is built on top of Apache SOLR[6], a facet capable indexing and searching engine that is easily extensible.

[5]http://en.wikipedia.org/wiki/The_Pearl_(novel)

[6]http://lucene.apache.org/solr/

The current Kino framework supports three facets based on the SA-REST specification. The index manages content of each annotation, the annotated text and the content of the document, hence the users have to flexibility to search on the annotated concept as well as the document content similar to a text based search engine.

III. Use cases and Motivation

We present two use cases that encompass two different document management tasks, encountered by biologists.

A. Scientific Worklfow

The first use case is a scientific workflow. This type of workflow is routinely used in a bioinformatics or a system biology laboratory. What a gene product does and how various factors affect its function are the fundamental questions to biologists. For this purpose, they carryout genome sequencing of the organism of interest; and, use the results for gene prediction, sequence alignment/comparison, identification of cellular location, and, function prediction. Figure 1 illustrates this process.

Genome sequencing is a process that determines the complete DNA sequence of an organism's genome. This is a chemical process that is usually automated and a machine outputs a large number of nucleotide sequences. A nucleotide sequences is simply a long array of characters, consisting of predefined characters that represent types of chemical compounds. However, this sequence does not provide any information on genes, their location, or their functions (information that are critical for biomedical researchers). Therefore, they need to analyze this long genomic sequence for various purposes using the data repositories that are available. The steps include:

1) Identify open reading frames/genes. This is a complex process as it varies with the type of organism (e.g., for bacteria and parasites there are different platforms available to run the same task).
2) Align or compare gene sequences of the relevant organism with others in the repositories to find the top hit for each gene.
3) Translate the amino acid sequence from the predicted gene to search either the protein database, or a translated nucleotide database. For some organisms that have phylogenetically important related organisms; one may also want to align the sequences against those genomes individually.
4) Identify motifs/domains that can give clues as to function for genes that do not have a close hit with a gene from another organism whose function has already been predicted. Researchers also analyze their sequences to give information about cellular location to determine if it is likely to be secreted or membrane bound.

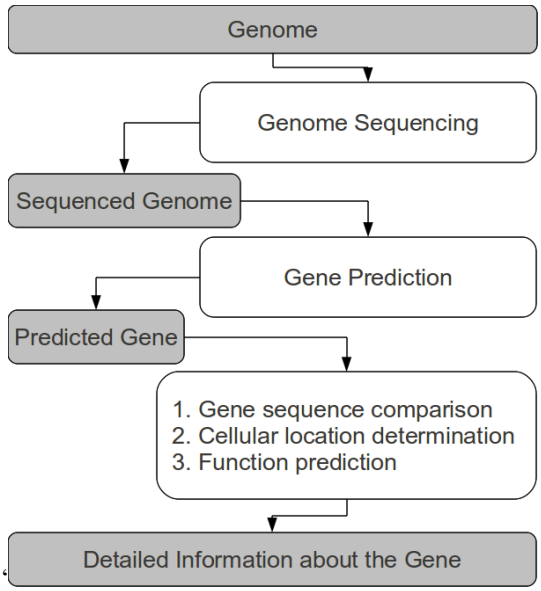

Figure 1. A typical Web service driven workflow, routinely encountered in gene identification

For such analysis (steps 1-4), researchers may use Web services and hence these tasks are the most likely candidates to become part of a service oriented workflow. This is indeed the case in many instances, as exemplified by myExperiment, a repository for scientific workflows[7]. Many organizations, such as the DNA Data Bank of Japan (DDBJ)[8], provide service interfaces for some of these operations. Biologists typically search and browse through a service catalog such as BioCatalogue and import the relevant service descriptions to a composer tool.

The difficulty of this task is that a biologist would have to use descriptive terms to extract the most suitable services. Often these terms are imprecise, and a few attempts are needed to get to the exact service, required for the task at hand.

B. Document Annotation

Our second use case comes from a genomics and genetics research group. The genome database GeneDB[9], at the Wellcome Trust Sanger Institute (WTSI), maintains a collection of more than 40 genomes, predominantly of pathogenic organisms, that is constantly updated and annotated. These annotations are prepared via rapid information and knowledge exchange between teams of literature annotators and data curators. Curation of the annotations is a collaborative effort that involves teams of scientists and bioinformaticians at four institutions. Annotations include literature and other database cross-references; GO terms inferred from the literature and user comments; and, phenotype curations that currently use a semi-controlled vocabulary.

[7]http://www.myexperiment.org/
[8]http://www.ddbj.nig.ac.jp/
[9]http://www.genedb.org

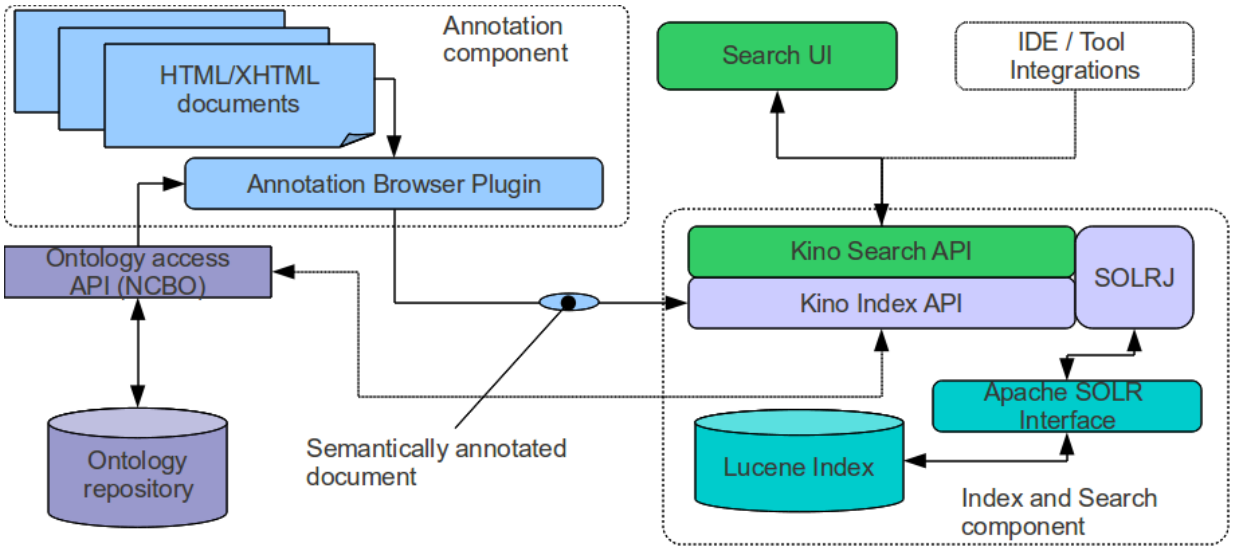

Figure 2. System architecture for the indexing and searching pipeline. The annotations performed via a tool, such as a browser plugin, are sent as an XML to the indexing engine. The indexer looks up further details, (such as synonyms) from NCBO and indexes them along with the document attributes

The primary tool used in the process is Artemis [5] and the associated Gene builder that provides the capability to annotate a coding sequence, or other sequence, on the database. Gene builder provides four types of annotation capabilities:

1) *Properties*: These contains feature properties, such as synonyms and time last modified. Synonyms are categorized using the controlled vocabulary tables in Chado[10]. The last modified time is updated when a change to that feature is written back to the database.
2) *Core*: The core annotation contains any annotation that does not fit into the other sections. For example, free text comments and cross-links to the scientific literature. Hyperlinks are provided for predefined databases (e.g., UniProt[11], EMBL[12], PubMed[13] and others), that open up a local browser.
3) *Controlled vocabulary (CV)*: The CV module in the schema is concerned with controlled vocabularies or ontologies. Artemis uses biological ontologies to allow very precise and expressive annotation. Currently GO is the primary ontology used in Artemis, although the capacity to include other ontologies is present.
4) *Match*: orthologue and paralogue links can be added to other genes in the database in this section.

The current workflow of annotating a document starts with the bioinformatician using the browser based Zotero[14] plugin to attach notes to the document. The subsequent steps involve curators going through these notes looking for GO terms manually and updating the annotations. The terms that are not defined in GO are related to other sources of literature. Completion of the annotations takes several rounds of annotation and correction by bioinformaticians and curators.

C. Motivation

The two use cases highlight many instances where improved integration and faceted search can aid the biologist.

In the case of a scientific workflow, the search for services in the catalog is based on imprecise terms and tags. One issue in this case is the differing interpretations of what the service does, which is reflected in the description as well as the tags applied to describe this service. Even though the existing ontologies provide an excellent source of standard vocabulary, most of the existing prominent bioinformatics service catalogs have a cumbersome service registering process that makes applying standard tags an extremely time consuming process.

In the case of document annotation, the lack of integration across tools is the most important issue. When the bioinformaticians add notes to the document, they do not have the capability to immediately verify the existence of a GO term. Similarly, when multiple ontologies are used, there is no tooling capability that lets them search the presence of ontology terms in all the relevant ontologies at once.

Both these use cases motivated us to introduce an annotation process with the following features.

- Driven by integrated tools such as browser plugins.
- Intuitive (i.e., can be performed with minimum training and effort).

[10]http://gmod.org/wiki/Chado_CV_Module
[11]http://www.uniprot.org/
[12]http://www.ebi.ac.uk/embl/
[13]http://www.ncbi.nlm.nih.gov/pubmed/
[14]http://www.zotero.org/

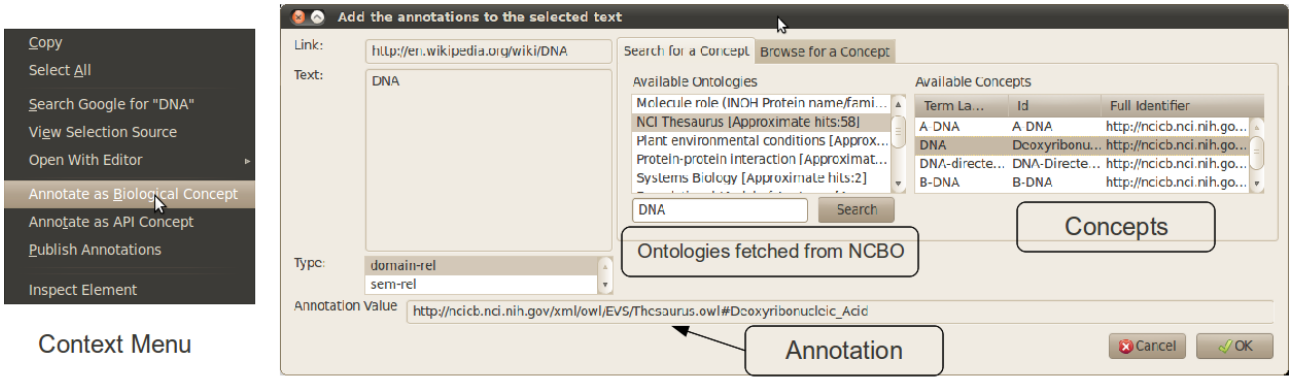

Figure 3. Browser plugin tool, shown with the right click menu and the ontology / concepts acquired by NCBO. The right click menu of the browser includes the annotation menu items. Once selected, the popup window shows NCBO ontology names and concepts that can be selected as the relevant annotation

- Provides convenient access to existing ontology terms at the point of annotation.
- Flexible and easily extensible.

IV. Architecture and Tool Details

This system is designed around the basic workflow consisting of three steps; annotate, index, and search. Figure 2 illustrates the major components of the system.

1) Annotation: In the annotation step, users provide annotations via various tools. The illustrated case is the use of browser plugin, but, it can be through a Web site, or an Integrated Development Environment(IDE). Once the annotations are added, the augmented document is submitted to the indexing engine.

2) Indexing: Indexing is performed using Apache SOLR. SOLR can be installed as an independent application and exposes multiple interfaces for client programs. SOLR provides isolation for the index as well as built-in faceting support, which can be controlled via a configuration file.

3) Search: The search uses a Javascript driven Web UI. It presents a typical search engine like interface, as well as the ability to filter the results via the facets. The current UI is based on the Kino JSON API that can be used to integrate any other tool or IDE.

This particular architecture makes the Kino system flexible, in terms of adopting it to different types of resources. For example, using PDF documents, only requires a PDF parser in Kino; the rest of the Kino components will be unaffected.

A. Browser Plugin for Annotation

Figure 3 illustrates the user interface of the annotator plugin. When the user highlights and right clicks in 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 the 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 ontology hosted in NCBO. 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.

The annotator, when used with highlighted text, modifies the HTML source as exemplified in Listing 1. Note that this is not the only modification the plugin may perform. For example, when the text is already contained by a logical grouping element such as a *div*, the plugin attempts to modify the existing element rather than adding a new one.

Listing 1. HTML Source Annotation added by the Browser Plugin

```
<span
  sarest:displayname="DNA"
  sarest:conceptid="Deoxyribonucleic_Acid"
  sarest:ontologyid="42693"
  title="http://ncicb.nci.nih.gov...
Thesaurus.owl#Deoxyribonucleic_Acid"
  class="sem-class">DNA</span>.
```

The required fields are *title* and *class* attributes, as mandated by the SA-REST specification. However the back-end requires certain NCBO specific details to be associated with this annotation (e.g., the ontology identifier and the concept identifier are required in the later operations of Kino). These details are added as extra attributes under the SA-REST namespace.

In submitting the documents, the plugin currently sends the full serialization of the internal document, in XML form, to the indexer. Although this does not include certain details such as the styling data, sending the whole document enables the index manager to keep a cache of submitted documents.

The annotator plugin is available to the public via the sourceforge hosting site[15].

[15]http://sarestannotator.sourceforge.net

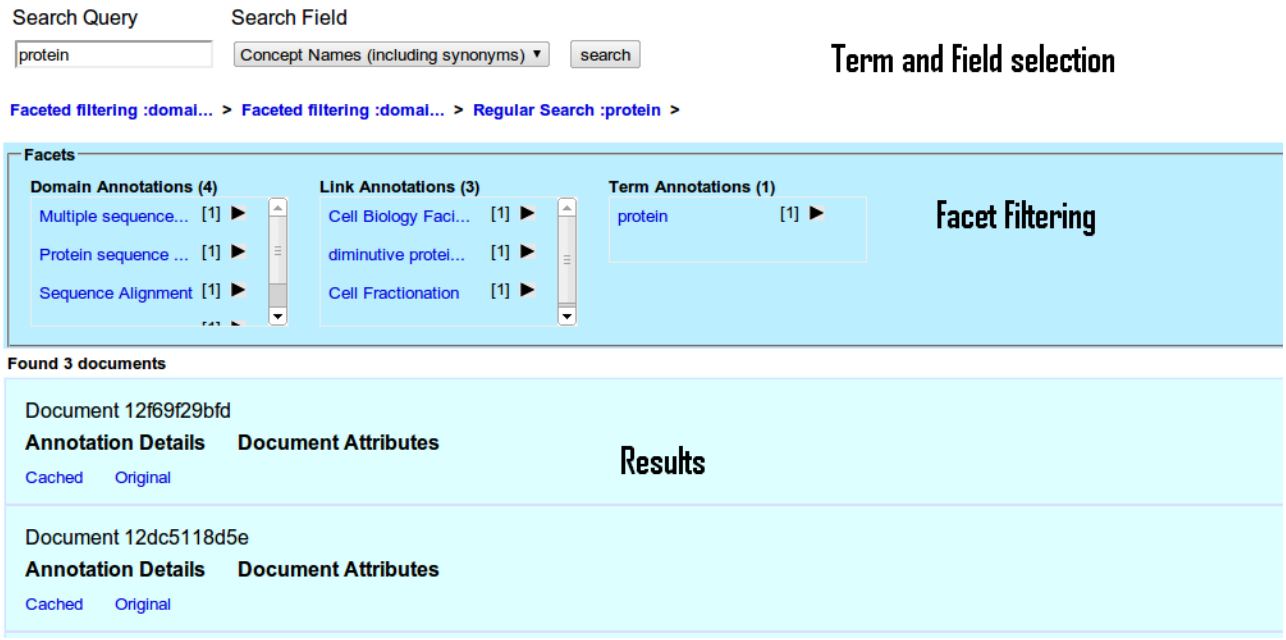

Figure 4. Web based User Interface for Searching. The UI includes search field selection and facet filtering facilities to assist the user in selecting the required results

B. Kino Index and Search Manager

The Kino index manager is based on the Java JSP/Servlets technology and consists of two major components.

Document Submission API: This consists of a HTTP POST based receiver that currently handles only XML input. The required XML format is a simple wrapper around the XHTML document. Once a document is submitted via this API, a process is spawned to index that document and a response is sent to the submitter.

The Kino indexer is added with extensions to search the NCBO ontologies for synonyms before indexing a document. Extensions can be added to support more functions, such as fetching the ancestor hierarchy. During the indexing process, the annotations are extracted and indexed as different fields. This enables independent search using each facet, as well as concept names, synonyms or any other extra details that get attached to the documents. The importance of these synonyms is presented in our empirical evaluation in Section V.

Search API: This is the primary API that supplies details to the front-end. The current search UI is a javascript client that utilizes this search API to generate a dynamic UI. This UI is illustrated in Figure 4 (The key sections are highlighted).

The UI includes a facet selection section that allows the user to filter the results. The users can issue keyword queries towards selected fields, including concept names, synonyms, or the text content. This is important for the power user, who needs the flexibility to switch between multiple search options.

Kino is also available for public use from the sourceforge hosting site[16].

V. Empirical Evaluation

As discussed in Section I, we refrain from doing a traditional evaluation. Instead, we performed an empirical evaluation with domain experts to highlight several observations where the use of this system is more advantageous than the existing ones.

In the case of biological Web services, we observed that BioCatalogue returns about 75 Web services for the search term *gene prediction*. However, it returns only 20 Web services for the term *gene finding*, even though gene prediction and gene finding are synonyms[17]. Similarly, for the terms *homology modelling* BioCatalogue returns results, but no results are returned for *comparative modelling* (two more synonyms [18]). We provide more commonly available synonyms in Table I. The essence of these observations is that such synonyms and cross references have been added to existing ontologies with significant effort and investment; although, the lack of integration leads to under utilizing these resources.

In the case of the document annotation, the WTSI is considering an alternate workflow using the integrated tools. Figure 5 illustrates the current workflow and the suggested workflow for annotating a document.

[16]http://apihut.sourceforge.net
[17]http://bioportal.bioontology.org/visualize/45158/?conceptid=EDAM:0000109
[18]http://bioportal.bioontology.org/visualize/45158/?conceptid=EDAM:0000175

Concept Label	Available Synonyms	Reference
gene finding	gene prediction	http://bioportal.bioontology.org/visualize/45158/?conceptid=EDAM:0000109
homology modelling	comparative modelling	http://bioportal.bioontology.org/visualize/45158/?conceptid=EDAM:0000175
nucleic acid sequence analysis	neucleotide sequence analysis	http://bioportal.bioontology.org/visualize/45158/?conceptid=EDAM:0000096
sequence alignment	sequence comparison	http://bioportal.bioontology.org/visualize/45158/?conceptid=EDAM:0000182
genetic mapping	genetic linkage, linkage mapping	http://bioportal.bioontology.org/visualize/45158/?conceptid=EDAM:0000103

Table I
TERMS AND SYNONYMS/CROSS REFERENCES ALREADY KNOWN BUT NOT TAGGED IN BIOCATALOGUE

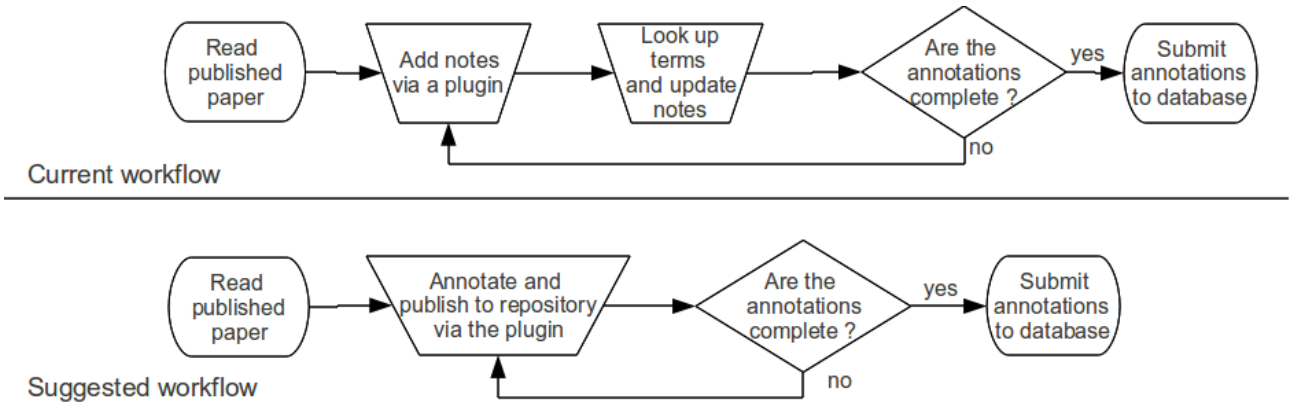

Figure 5. Current and suggested workflows for annotating a document at the WTSI

The suggested workflow eliminates the term lookup task that takes significant effort. The biologists can now directly annotate a document and submit it to an index. The current Kino framework does not support updates, a major feature we are planning to add to support the suggested workflow at WTSI. The main database is of production quality, thus the intermediate annotations may be published to a temporary repository.

VI. RELATED WORK

Using ontologies as the driving knowledge base for document management is not new. Applying ontologies for document management has been investigated early by the Webocracy project from the European Union. It is based on a combination of annotating documents with concepts from a knowledge base and grouping documents together into clusters [6]. The *webocrat* system, the primary software component developed by this project, has been tested in several real world settings.

Life sciences is one of the scientific domains that embraced ontologies at an early stage for document management. Textpresso is one of the projects for full text literature searches for specific organism, text classification and mining literature for database curation and make a link between biological entities in RDF and online journal articles to online databases [7]. The focus of Textpresso however, includes text analysis to support a semi-automated annotation process. This is possible only when the base ontology is predefined. Our tools do not force the use of a single ontology and depend on the intelligence and choice of the human annotators to select the most suitable concept for annotations. The core annotation technology however can still be used with automated annotators in case the ontologies to be used are predefined.

Artemis and the associated Gene builder, discussed in Section III, provide tooling for annotating a coding sequence or other sequence on the database. Artemis is highly specialized to address the particular case of annotating a genome sequence in supported formats.

The case in all these tools is that they are specialized, limiting their applicability to specific domains. Kino encapsulates the basic document management workflow in a generic fashion using standardized technologies, making it widely applicable across many domains. It also integrates NCBO, the premier resource for life science ontologies, utilizing the results of collaborative modeling efforts of a large community of scientists.

SA-REST has been listed as a candidate for service annotations and ultimately automating RESTful service compositions [8]. An early version of SA-REST has been used to demonstrate partially automated RESTful service composition tasks [9]. SA-REST, however, has not been considered as a general purpose annotation framework, thus has not seen usage outside the RESTful service community.

Kino promotes the use of SA-REST in a generic fashion, applying it to all applicable Web resources.

VII. Future Work

A. Collaboration Features

Given that most of these document annotation tasks are performed by teams rather than individuals, collaboration features are important additions. There are at least 3 cases where integrated collaboration features may be useful.

1) When determining the accuracy of an annotation. A typical data curation process takes several rounds with the experts. The ability to interact through an integrated environment could greatly reduce the cost and effort of this collaboration.
2) When determining the reputation of a curator/annotator. Similar to a product rating, an annotation can have a rating, and this rating can be factored in when determining the reputation of the annotator.
3) When working with geographically and demographically dispersed teams. Timezone incompatibilities as well as cultural differences can be easily offset with a good collaborative tool. For example, integrated instant messaging support can be used to enable instant communication between geographically dispersed team members.

We plan to integrate some of these features in future releases of Kino.

B. Utilizing Further Knowledge from Ontologies

The current system uses only synonyms from the ontologies during the indexing process. There is great potential to use more sophisticated data from ontologies. These include:

1) The class hierarchy, especially the ancestor hierarchy.
2) Cross references.
3) Other information such as "owl:sameAs" or "owl:differentFrom" property entries.

This information can be readily extracted from NCBO and included in the index during the indexing process. However, a balance has to be found regarding the extent of knowledge extracted from the ontologies to avoid noise. For example, using the entire ancestor hierarchy may be expensive and noisy in a deep ontology. A limited ancestor extraction may be required in such a case.

VIII. Conclusion

We have presented our standard-driven annotation and indexing tool set, showing applicability across multiple scientific domains. These tools help to streamline existing annotation tasks, facilitate the use of existing ontologies and enable the full benefit of using ontologies as knowledge bases for document management. Given that there are many steps in scientific document management processes that are cumbersome due to the disconnectedness of the tooling, we conclude that the integration demonstrated in these tools is indeed useful.

Acknowledgements

This work is supported by NIH R01 Grant #1R01HL087795-01A1.

References

[1] M. Ashburner, C. Ball, J. Blake, D. Botstein, H. Butler, J. Cherry, A. Davis, K. Dolinski, S. Dwight, J. Eppig *et al.*, "Gene Ontology: tool for the unification of biology," *Nature genetics*, vol. 25, no. 1, pp. 25–29, 2000.

[2] Gomadam,K. and Ranabahu, A. and Sheth, A., "SA-REST: Semantic Annotation of Web Resources," http://www.w3.org/Submission/SA-REST/.

[3] A. Sheth, K. Gomadam, and J. Lathem, "SA-REST: Semantically interoperable and easier-to-use services and mashups," *Internet Computing, IEEE*, vol. 11, no. 6, pp. 91–94, 2007.

[4] K. Gomadam, A. Ranabahu, M. Nagarajan, A. Sheth, and K. Verma, "A faceted classification based approach to search and rank web apis," in *Web Services, 2008. ICWS'08. IEEE International Conference on*. IEEE, 2008, pp. 177–184.

[5] K. Rutherford, J. Parkhill, J. Crook, T. Horsnell, P. Rice, M. Rajandream, and B. Barrell, "Artemis: sequence visualization and annotation," *Bioinformatics*, vol. 16, no. 10, p. 944, 2000.

[6] J. Paralic, T. Sabol, and M. Mach, "First trials in Webocracy," *Electronic Government*, pp. 69–74, 2003.

[7] H.-M. Mller, E. E. Kenny, and P. W. Sternberg, "Textpresso: An ontology-based information retrieval and extraction system for biological literature," *PLoS Biol*, vol. 2, no. 11, p. e309, 09 2004.

[8] H. Zhao and P. Doshi, "Towards automated restful web service composition," in *Web Services, 2009. ICWS 2009. IEEE International Conference on*. IEEE, 2009, pp. 189–196.

[9] J. Lathem, "SA-REST: Adding Semantics to REST-Based Web Services," *Master's Thesis, University of Georgia, Athens, Georgia*, 2007.