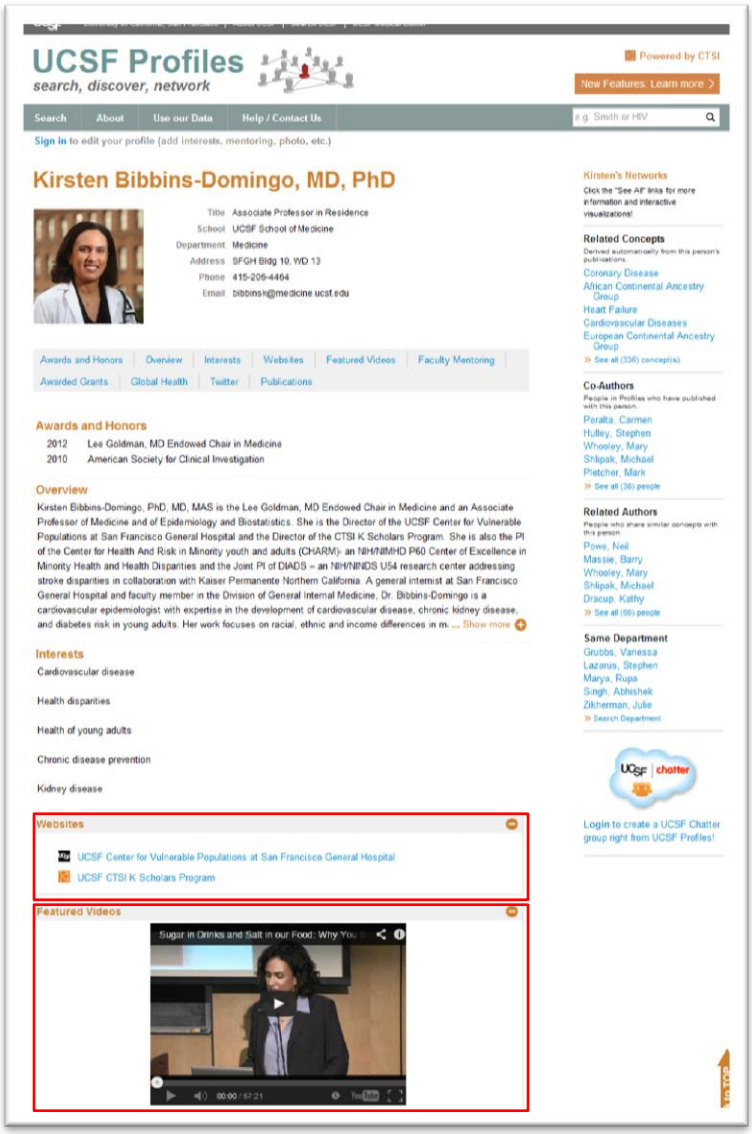
Using Linked Data to Extend OpenSocial in Research Networking

Eric Meeks, Brian Turner, Anirvan Chatterjee, Leslie Yuan,
University of California, San Francisco, CA

Background:

Research Networking refers to the adoption of LinkedIn style web sites that are used in biomedical research to showcase researchers and their expertise. With these sites, a wealth of information can be found supporting use cases that range from helping researchers find collaborators based on specific expertise, to someone from the general public finding a doctor based upon their work and background. At UCSF, we brought OpenSocial into research networking years ago by extending two of the prominent open source tools used to power research networking web sites, namely VIVO[1] and Harvard Catalyst Profiles[2]. More recently, we have extended our OpenSocial integration by adding in Linked Data support. In practice, this required us to extend Apache Shindig with Linked Data-to-JSON conversion capabilities. While varying solutions for exposing Linked Data as JSON can be found, today there exists an emerging standard known as JSON-LD[3] which we now support in our integration. We refer to our work as Open Research Networking Gadgets, or ORNG.



OpenSocial applications as seen in production highlighted in red
From UCSF Profiles: http://profiles.ucsf.edu

Approach:

UCSF added OpenSocial capabilities to our research networking tool prior to the widespread adoption of Linked Data within our field. We developed a collection of OpenSocial applications that relied heavily on

[1] http://vivoweb.org
[2] http://profiles.catalyst.harvard.edu
[3] http://json-ld.org

the “osapi” feature for data access. This allowed our applications to access relevant information about our researchers (both the VIEWER who may be logged into the site, as well as the OWNER which would be the person whose profile page you were viewing) as defined by the OpenSocial “Person” specification. However, the OpenSocial “Person” specification does not include data targeted to our domain such as authored publications, awarded grants, patents, and other items meaningful to biomedical research. In practice, we found that different institutions needed access to this data within their OpenSocial applications, and they would create proprietary solutions to meet these needs. Wake Forest developed one means to access a researchers publications from within an OpenSocial application, whereas UCSF had developed a different solution to the same problem at an earlier time. As a result, you could not port OpenSocial applications developed by Wake Forest into UCSF without some form of code change.

Today with Linked Data, this problem can be solved. Linked Data has become a recommended standard for research networking and it is now supported by a number of the web sites in our field. This includes both VIVO and Profiles, as well as proprietary web sites such as LOKI at Iowa University. At the sites running these tools researcher data is available in a format known as RDF/XML. More specifically, our researcher data conforms to an ontology known as the VIVO ontology (an ontology is a type of data model for Linked Data). The VIVO ontology in turn is derived from the well know FOAF[4] (Friend of a Friend) ontology. RDF/XML is a serialization format for the semantic data in our research networking tools. Other serialization formats for Linked Data exist, but RDF/XML is the most highly supported. Unfortunately RDF/XML is unwieldy to work with in Javascript, and this makes it difficult for support in OpenSocial applications.

We were now faced with an interesting problem and opportunity. If we continue to work with the OpenSocial definition of a “Person”, we will have to heavily extend that definition to make the “Person” look more like a researcher. The RDF/XML Linked Data offered a potential solution to this problem in that it conforms to an ontology specific to our domain, but the data is in a challenging format and access to the data from within an OpenSocial application (who is the VIEWER and who is the OWNER) is not obvious or standard. Formally integrating JSON-LD into our OpenSocial platform provided us with the solution we needed.

Technical aspects:

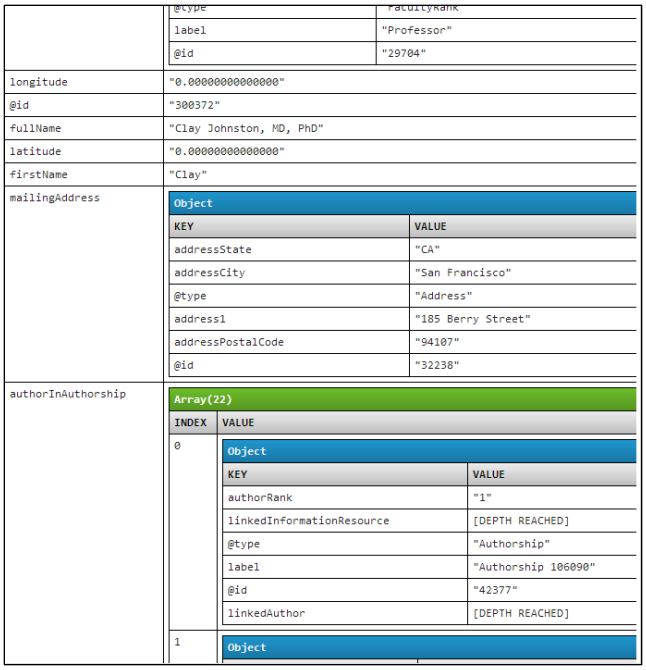
Open source solutions for converting RDF/XML into JSON-LD exists in many different languages, including Java. We were able to integrate JSON-LD capabilities into Apache Shindig and create an “rdf” feature that extends osapi and allows for easy access to the researcher data as JSON. In particular, we now have the following methods:

osapi.rdf.getViewer(…)	Access to the VIEWER as JSON-LD
osapi.rdf.getOwner(…)	Access to the OWNER as JSON-LD
osapi.rdf.getRDF(URI, …)	Access to any Linked Data (internal or external) as JSON-LD via URI

[4] http://xmlns.com/foaf/spec/

Note that with JSON-LD, the metadata describing the ontologies and versions of the returned data is included in the payload, so the returned data is self-described.

An interesting solution came out of our JSON-LD extension with regards to complex data structures. As a serialized format, JSON does not intrinsically support complex data with circular references. The rich data we have on our researches does, however, contain many circular references. As one simple example: researches have publications they have written, publications in turn are formal objects which list their corresponding authors. Circular references are common and desired with Linked Data, and the solution with RDF/XML to serialize this data is by "flattening" out all relationships to avoid infinite loops. With Linked Data every object has a URI, and in the serialized form all object-to-object relationships are represented as object-to-URI relations. But as a developer, having to deal with a large set of Javascript objects that contain URI's that you have to constantly dereference in order to access is cumbersome. Fortunately, with JSON-LD it is easy to rebuild the object-to-object graph in a secondary "deserialization" step that is agnostic of the ontology, and thus works for any Linked Data.



Detail of deserialized Javascript for a researcher as seen through "Pretty Print"

A second note is related to the concept of page Owner. With Linked Data, all objects have a URI that is essentially their home page. The concept of Owner within OpenSocial currently refers to a Person, but with Linked Data other objects will also have home pages, and they could potentially be the "Owner" to an OpenSocial application. The osapi.rdf methods created for ORNG allow for this broader definition, and also allow for any arbitrary types of relationships to be explored between objects and people.

Next Steps:

Our JSON-LD work with OpenSocial is available as open source in the UCSF branch of the Profiles Research Networking software available on GitHub[5]. This code will be merged in to the master branch and released as Profiles 2.0 in August 2013. Next we will add the JSON-LD extension into the VIVO source, which is managed by DuraSpace[6]. Wake Forest and Baylor (as well as UCSF) have deployed earlier versions of our OpenSocial extension of Profiles into production and built a few OpenSouce applications for our community. This OpenSocial extension was never a part of the official Profiles product, but that will change with the Profiles 2.0 release. We are also working with Boston University to build a JSON-LD/OpenSocial application for integrating ORCID[7] into our research networking tools.

[5] http://gitub.com/ProfilesRNS
[6] http://www.duraspace.com
[7] http://orcid.org/

This application needs access to a large amount of researcher specific data, so access to the researchers Linked Data is crucial.

Discussion:

Linked Data is a perfect complement to OpenSocial because it overcomes the issues of data impedance found across our different social networking sites. The data that describes people, their properties, other objects and the myriad relationships between them varies from one site to the next, and it is a constantly moving target even amongst fields like biomedical research that have adopted strict data standards. Linked Data is the best solution to date for making this data available in a future proof machine readable format, and with JSON-LD this data can be accessed by “lightweight” applications such as those defined by the OpenSocial standard. Our work at UCSF in combining Linked Data with OpenSocial is currently being applied within the field of biomedical research, but the technology supporting ORNG is agnostic to our domain. As Linked Data expands to cover more industries and domains, we expect to find a broader application of the type of approach we have made to bring our data into hosted applications.