Harvard Scholarship Indexing Project (HSIP) – Progress Report

Introduction

The purpose of this project is to create a metadata repository for Harvard faculty scholarship. We will accomplish this by adding "push" APIs to the Harvard Faculty Finder (HFF) website to enable the owners of publication databases across Harvard to send metadata about those publications to HFF in a standardized way. The existing "pull" APIs in HFF enable users to extract that metadata in a variety of ways. Our work on this project so far has been to define the functionality of the push APIs and format of the data to be sent to those APIs.

Background

HFF is a new website, which provides, for the first time, a University-wide view of Harvard faculty and scholarship in order to help students, faculty, and others identify Harvard faculty according to their research and teaching expertise. It is currently available to the Harvard community at http://facultyfinder.harvard.edu. HFF is not a faculty profiles tool and does not attempt to aggregate comprehensive faculty information. Instead, HFF indexes and links existing sources of public information to enable cross-school faculty search and browse, including topic search. A key feature of HFF is that it is a Semantic Web application, which uses the Resource Description Framework (RDF) to represent information in a standard data exchange format. HFF makes this data available to other computer systems through several types of read-only ("pull") application programming interfaces (APIs). Thus, HFF not only collects data from multiple sources, it serves as a single central location where users can access those sources of data in a standardized way. Documentation for the APIs is online at http://api.facultyfinder.harvard.edu.

In order for HFF to store publication data in RDF format, it requires an ontology that defines the properties that can be represented and how people and publications can be linked together. For this, HFF uses the VIVO ontology, which was developed by a consortium of universities funded by a federal grant to describe the people and scholarly activities at academic institutions. VIVO, in turn, is based on other commonly used ontologies, such as Friend of a Friend (FOAF) for people and the Bibliographic Ontology (BIBO) for publications. VIVO includes three distinct types of entities: a Person, an InformationResource (e.g., a publication), and an Authorship that links a Person to an InformationResource. This structure enables multiple people to be linked to the same publication and multiple publications to be linked to the same person. The Authorship entity can include its own metadata, such as the person's authorship position (e.g., first author).

In RDF, each Person, InformationResource, and Authorship is assigned a unique URI. When RDF is presented in XML format, an "rdf:Description" tag contains the properties for a given URI. Figure 1 is an example of an RDF/XML document that uses three rdf:Description tags to describe a person linked to a single publication. (Many properties are removed from this example for clarity.) HFF collects publication data from several sources, including Thomson Reuters' Web of Science, Harvard's open access manuscript repository DASH, the Harvard OnLine Library System (HOLLIS), and databases provided by individual Harvard schools. To keep track of the source of a publication and its ID within that source, HFF extends the VIVO ontology to include additional classes and properties. The example in Figure 1 illustrates this with the use of a "dashId" property.

Figure 1. An RDF/XML representation of a person linked to a publication in HFF.

```
<rdf:RDF>
  <rdf:Description rdf:about="http://facultyfinder.harvard.edu/profile/1177154">
    <rdf:type rdf:resource="http://xmlns.com/foaf/0.1/Person" />
    <rdf:type rdf:resource="http://vivoweb.org/ontology/core#FacultyMember" />
    <rdfs:label>Weber, Griffin</rdfs:label>
    <foaf:firstName>Griffin</foaf:firstName>
    <foaf:lastName>Weber</foaf:lastName>
    <vivo:authorInAuthorship rdf:resource="http://facultyfinder.harvard.edu/profile/241680" />
  </rdf:Description>
  <rdf:Description rdf:about="http://facultyfinder.harvard.edu/profile/241680">
    <rdf:type rdf:resource="http://vivoweb.org/ontology/core#Authorship" />
    <vivo:linkedAuthor rdf:resource="http://facultyfinder.harvard.edu/profile/1177154" />
    <vivo:linkedInformationResource rdf:resource="http://facultyfinder.harvard.edu/profile/508909" />
  </rdf:Description>
  <rdf:Description rdf:about="http://facultyfinder.harvard.edu/profile/508909">
    <rdf:type rdf:resource="http://vivoweb.org/ontology/core#InformationResource" />
    <rdf:type rdf:resource="http://purl.org/ontology/bibo/Document" />
    <rdf:type rdf:resource="http://purl.org/ontology/bibo/Article" />
    <rdf:type rdf:resource="http://purl.org/ontology/bibo/AcademicArticle" />
    <rdf:type rdf:resource="http://profiles.catalyst.harvard.edu/ontology/catalyst#DashArticle" />
    <catalyst:dashId>oai:dash.harvard.edu:1/2031671</catalyst:dashId>
    <rdfs:label>Representation in stochastic search for phylogenetictree reconstruction</rdfs:label>
    <prns:informationResourceReference>Weber, Griffin, Lucila Ohno-Machado, and Stuart M. Shieber.
        Representation in stochastic search for phylogenetictree reconstruction. Journal of Biomedical
        Informatics 39.1 (2006): 43-50. Copyright World Scientific Publishing Company.
        http://www.sciencedirect.com/science/journal/15320464.</prns:informationResourceReference>
  </rdf:Description>
</rdf:RDF>
```

Push API Functionality

We have decided that the default push API will essentially work in the exact opposite way as the pull API. When requesting publication data from HFF, it is returned in RDF/XML format as illustrated in Figure 1. Users will be able to use the same RDF/XML format to send publication data to the push API in order to link those publications to faculty in HFF.

Although Figure 1 describes a single authorship, one RDF/XML message sent to the push API can include any number of people, publications, and authorship links. This will make it easier for a user to send an entire publication archive to HFF in one bulk process.

It will not be necessary to include in the RDF/XML message data that already exists in HFF. For example, every Harvard faculty member is already defined in HFF using RDF. Therefore, the rdf:Description tag corresponding to the faculty member can be removed as long as the rdf:Description tag corresponding to the Authorship has the correct URI for the faculty member. Thus, the push API will look for publication and authorship descriptions in the RDF/XML and ignore other types of entities including people. Similarly, if the ID of a publication matches one that already exists in HFF, then a duplicate publication record will not be created.

Because a publication and authorship are assigned URIs only after they are loaded into HFF, the RDF/XML for the push API can use the RDF concept of “blank nodes” that have an rdf:nodeID as a temporary unique ID within the context of that RDF/XML message. Figure 2 illustrates this. (Again, many properties are removed for clarity in Figure 2.)

Figure 2. An example RDF/XML message sent to the HFF push API.

```
<rdf:RDF>
  <rdf:Description rdf:nodeID="A1">
    <rdf:type rdf:resource="http://vivoweb.org/ontology/core#Authorship" />
    <vivo:linkedAuthor rdf:resource="http://facultyfinder.harvard.edu/profile/1177154" />
    <vivo:linkedInformationResource rdf:nodeID="P1" />
  </rdf:Description>
  <rdf:Description rdf:nodeID="P1">
    <rdf:type rdf:resource="http://profiles.catalyst.harvard.edu/ontology/catalyst#DashArticle" />
    <catalyst:dashId>oai:dash.harvard.edu:1/2031671</catalyst:dashId>
    <rdfs:label>Representation in stochastic search for phylogenetictree reconstruction</rdfs:label>
    <prns:informationResourceReference>Weber, Griffin, Lucila Ohno-Machado, and Stuart M. Shieber.
        Representation in stochastic search for phylogenetictree reconstruction. Journal of Biomedical
        Informatics 39.1 (2006): 43-50. Copyright World Scientific Publishing Company.
        http://www.sciencedirect.com/science/journal/15320464.</prns:informationResourceReference>
  </rdf:Description>
</rdf:RDF>
```

The push API will return to the user a summary report in XML format on whether the request message could be parsed correctly and details on whether records were successfully added to HFF or not.

Although RDF/XML and the VIVO ontology are rapidly gaining adoption in academic universities, we recognize that format is still unfamiliar to many people. Therefore, we will also develop push APIs that support the more commonly used Research Information Systems (RIS) and EndNote XML file formats. These APIs will basically be proxies that transform the RIS or EndNote XML into VIVO RDF/XML and send it to the native HFF push API. This model can be extended in the future to support other types of formats by similarly creating proxies that transform publication and authorship data into VIVO RDF/XML.

Next Steps

The next part of this project will involve designing the software components that will be needed to parse the VIVO RDF/XML data and incorporate it into the existing publication data feeds for HFF. This will be followed by the implementation of the software and the development of proxies that can transform RIS and EndNode XML into VIVO RDF/XML.