Re-using VIVO data in other websites

Question:

We're generating definite interest from Communications Officers and Web Administrators across departments for pulling data out of VIVO related to faculty publications, etc. to lower the burden of acquisition of this information in maintaining these external, public-facing websites. I'm aware that other VIVO sites are doing this, Cornell possibly being a leading example. From a practical standpoint, can anyone help us understand what the mechanisms are for doing this and what environments/expertise need to be in place on the part of the web administrators for querying VIVO and returning results that can be repurposed for their use?

Responses

There are several mechanisms available with different degrees of flexibility and maturity, and more options are becoming available on a regular basis as the semantic web and linked open data communities grow.

Most techniques currently rely on having a SPARQL endpoint for VIVO. Cornell has used Sesame for some time now, and UF supports [Fuseki.

- RDF to JSON paths are very useful
  - Eric Meeks is starting (1/2013) to look at http://json-ld.org and https://github.com/tristan/jsonld-java
  - There's are two implementations of the linked data API from Google Code
    - in Java, by Epimorphics: Elda
    - in PHP: Puelia
  - see also John Fereira's semantic services below
- PHP tools used on the University of Florida CTSI people pages (e.g., http://www.ctsi.ufl.edu/people/clinical-research-unit-directors/
- Python scripts running SPARQL queries and generating HTML pages on the University of Florida CTSI pages [http://www.ctsi.ufl.edu/institutes/,], http://www.ctsi.ufl.edu/core-labs/
- Miles Worthington has developed the RdfImporter module for Drupal ([This approach supports the http://impact.cals.cornell.edu site, which uses other Drupal modules for features such as map views.
- John Fereira has developed semantic services (https://semanticservice.svn.sourceforge.net/svnroot/semanticservice).]) that are used in production at Cornell for sites such as [and http://www.classics.cornell.edu]. These services parse the results of SPARQL queries in Java and then offer XML and JSON outputs familiar to most webmasters. The queries and Java objects that interpret them have to be created and maintained but the services have a administrative UI, an HTML preview, and handle XML escaping and repeated values well.
- the Duke VIVO Widgets - grant project mini grant is developing tools in Javascript to enable web masters or end users to pull standard content such as recent publications or grants into other websites much as they would a video or image from an Internet service