**DVDocs Overview**

DVDocs is a Java API with a web service wrapper which can be used to generate documents from VIVO profiles. DVDocs converts a VIVO RDF/XML profile to a standard XML format using XSPARQL then processes this with Apache FOP to generate a document. Two FO document templates are provided in the package to produce curricula vitae and NIH biosketches. Documents can be generated in pdf or rtf format.

**API Architecture**

The Java API, depicted in figure 1, consists of two main components: XSPARQL and Apache FOP. XSPARQL, a tool for converting RDF to XML, is used to convert VIVO profiles provided in RDF/XML format into the XML format used by DVDocs for generating documents.

The API uses two publicly available Java libraries: XSPARQL from the Digital Enterprise Research Institute (see <http://xsparql.deri.org/>) and Apache FOP from the Apache Software Foundation (see <http://xmlgraphics.apache.org/fop/>). XSPARQL is used to convert RDF/XML data to standard nested XML data. Apache FOP is used to generate documents.

The API supports generating documents from VIVO RDF/XML profiles or directly from a DVDoc formatted XML. If RDF/XML is provided it will be converted to XML with XSPARQL then passed to Apache FOP to be rendered into a document. Alternatively a profile in the DVDocs XML format can be passed directly to the Apache FOP layer. The document type can be specified as an NIH biosketch or curriculum vitae (both supported natively), or a custom document template can be provided.

Figure : DVDocs API

XSPARQL

NIH Biosketch FO Template  
(nih-biosketch.fo.xsl)

Curriculum Vitae FO Template  
(curriculum-vitae.fo.xsl)

XSPARQL VIVO to DV Queries  
(vivo2dv.xs)

Apache FOP

**Converting VIVO RDF/XML to DV XML**

When a profile is provided in VIVO RDF/XML format it is converted to XML using XSPARQL. The XSPARQL query used to do the conversion is located within the source code at   
digital-vitaDOCS/src/main/java/document-templates/vivo2dv.xs.

**A note about completeness:** Currently two pieces of information required by NIH biosketches are not supported by this conversion because the information is not available in VIVO profiles. The first piece, narrative required in the Personal Statement section of NIH biosketches, is not included. Also, a flag for publications specifying whether the publications are “most relevant to the current application” is missing. If this information is eventually available via an RDF/XML representation of a VIVO profile, support will need to be added to the XSPARQL query to add it to the Digital Vita document format XML file. The information will need to be added according to the DVDocs XML schema (digital-vitaDOCS/schema/dv-docs.xsd) as shown here:

<personalStatements>  
 <statement>  
 A statement describing why the applicant is suited to the role  
 described in the application.  
 </statement>  
</personalStatements>

<personsPublications>

...   
 <relevantToCurrentApplication>true</relevantToCurrentApplication>  
<personsPublications>

**Using the API**

There are four methods exposed via the Java object DVDocs which can be used to generate a document. These methods support generation of documents given four sets of options. These options are:

* Input Format
  + VIVO RDF/XSL
  + Digital Vita XML document format – Profile data in XML format conforming to the DVDocs XML schema (digital-vitaDOCS/schema/dv-docs.xsd).
* Document Type
  + Curriculum Vitae – The CV template included with the API will be used to generate the document.
  + NIH biosketch – The NIH biosketch template included with the API will be used to generate the document.
  + Custom defined – The document will be generated used a provided custom XSL-FO template.
* File Format
  + Portable document format (pdf).
  + Rich text format (rtf).
* Publication Citation Format
  + Vancouver – A standard citation reference style commonly used for manuscripts submitted to medical journals, also known as Index Medicus format.
  + APA – A citation reference style defined by the American Psychological Association; often used by schools of Nursing.

The four document generation methods are:

* DVDocs.generateDocumentFromVivo – Generate a document using a predefined document format given a profile in VIVO RDF/XSL.
* DVDocs.generateCustomDocumentFromVivo – Generate a document using a custom document template given a profile in VIVO RDF/XSL.
* DVDocs.generateDocument – Generate a document with a predefined document format given a profile in Digital Vita XML.
* DVDocs.generateCustomDocument – Generate a document using a custom document template given a profile in Digital Vita XML.

Further information about these methods is provided in the description of the DVDocs object in the accompanying Javadocs (available at digital-vitaDOCS/documentation/javadoc/index.html after the package has been built).

The API is provided as a jar file and source code. To build the jar file from the source code see the file build-instructions.txt.

**Creating custom document formats**

DVDocs supports the creation of custom documents via the DVDocs.generateCustomDocumentFromVivo and DVDocs.generateCustomDocument methods. These two methods both take as one of their arguments a reference to an XSL-FO template file that is used to generate the document. See the documentation at <http://xmlgraphics.apache.org/fop/> for further information on creating XSL-FO templates. Also, the standard templates provided with the DVDocs API can be used as examples. The standard templates are stored in the DVDocs source code in digital-vitaDOCS/src/main/java/document-templates.

**The web service**

The RESTful web service wrapper to the DVDocs API supports generation of curricula vitae using Vancouver style publication citations from a VIVO profile. Documents can be generated in either pdf or rtf format. The data source for the VIVO profile is specified by providing a server name and person identifier as parameters to the web service call. Optionally pdf or rtf format can also be specified as a parameter, defaulting to pdf if it isn’t specified. The web service takes the form of:

http://server/dv-docs/vivo-doc-gen/vivoServerName/personId?filetype=type

Additional documentation about the DVDocs web service is provided in the accompanying Javadocs (available at digital-vitaDOCS-WS/documentation/javadoc/index.html after the package has been built).

The web service is provided as source code and a war file which can be deployed to a Tomcat application server. To build the war file from the source code see the file build-instructions.txt.