

# DSR to VIVO XSLT example

## Header

The standard BSD license and authors header.

```
<!--
Copyright (c) 2010 Christopher Haines, James Pence, Dale Scheppler, Nicholas Skaggs, Stephen V. Williams.
All rights reserved. This program and the accompanying materials
are made available under the terms of the new BSD license
which accompanies this distribution, and is available at
http://www.opensource.org/licenses/bsd-license.html

Contributors:
    Christopher Haines, Dale Scheppler, Nicholas Skaggs, Stephen V. Williams - initial API and implementation
    James Pence
-->
<!-- Header information for the Style Sheet
    The style sheet requires xmlns for each prefix you use in constructing
    the new elements
-->
```

## Name Spaces

The xsl uses and puts the proper name spaces into the output file.

```
<xsl:stylesheet version="2.0"
  xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:core="http://vivoweb.org/ontology/core#"
  xmlns:score='http://vivoweb.org/ontology/score#'
  xmlns:rdfs='http://www.w3.org/2000/01/rdf-schema#'
  xmlns:ufVivo='http://vivo.ufl.edu/ontology/vivo-uf1/'
  xmlns:db-dbo.vwVIVO='jdbc:jtds:sqlserver://10.241.46.60:1433/DSR/fields/dbo.vwVIVO/'
  xmlns:db-dbo.vwContracts='jdbc:jtds:sqlserver://10.241.46.60:1433/DSR/fields/dbo.vwContracts/'
  xmlns:db-dbo.vwProjectTeam='jdbc:jtds:sqlserver://10.241.46.60:1433/DSR/fields/dbo.vwProjectTeam/'
  xmlns:db-dbo.vwProjects='jdbc:jtds:sqlserver://10.241.46.60:1433/DSR/fields/dbo.vwProjects/'>

  <xsl:output method="xml" indent="yes"/>
  <xsl:variable name="baseURI">http://vivoweb.org/harvest/uf1/dsr/</xsl:variable>

  <xsl:template match="rdf:RDF">
    <rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
      xmlns:core="http://vivoweb.org/ontology/core#"
      xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
      xmlns:ufVivo='http://vivo.ufl.edu/ontology/vivo-uf1/'
      xmlns:score='http://vivoweb.org/ontology/score#'
      xmlns:db-dbo.vwVIVO='jdbc:jtds:sqlserver://10.241.46.60:1433/DSR/fields/dbo.vwVIVO/'
      xmlns:db-dbo.vwContracts='jdbc:jtds:sqlserver://10.241.46.60:1433/DSR/fields/dbo.
vwContracts/'
      xmlns:db-dbo.vwProjectTeam='jdbc:jtds:sqlserver://10.241.46.60:1433/DSR/fields/dbo.
vwProjectTeam/'
      xmlns:db-dbo.vwProjects='jdbc:jtds:sqlserver://10.241.46.60:1433/DSR/fields/dbo.
vwProjects/' >
      <xsl:apply-templates select="rdf:Description" />
    </rdf:RDF>
  </xsl:template>
```

## Template Determination

This section distills some key information and determines which template to use. The templates are geared towards translating the information for two of the views provided. the information in those two views is already sorted into contracts(grants) and project team(people). The separation of the views facilitates classification of the data.

```

<xsl:template match="rdf:Description">
  <xsl:variable name="this" select="." />
  <xsl:analyze-string select="..@xml:base" regex="^.*/([^/]+)?$">
    <xsl:matching-substring>
      <xsl:variable name="table" select="regex-group(1)" />
      <xsl:variable name="rdfid" select="$this/@rdf:ID" />

      <xsl:analyze-string select="$rdfid" regex="^id_-(.?.*)(_-.+)?$" />
      <xsl:matching-substring>
        This is where the types of data are translated differently-->
        <xsl:choose>
          <xsl:when test="$table = 'dbo.vwContracts'">
            <xsl:call-template name="t_vwContracts">
              <xsl:with-param name="grantid" select="
normalize-space( regex-group(1) )" />
              <xsl:with-param name="this"
select="$this" />
            </xsl:call-template>
          </xsl:when>
          <xsl:when test="$table = 'dbo.vwProjectTeam'">
            <xsl:call-template name="t_vwProjectTeam">
              <xsl:with-param name="this"
select="$this" />
            </xsl:call-template>
          </xsl:when>
        </xsl:choose>
      </xsl:matching-substring>
    </xsl:analyze-string>
  </xsl:matching-substring>
</xsl:analyze-string>
</xsl:template>

```

## Contracts

This template is the contacts template and deals with most of the data specific to a grant. Including links to the PI and CoPI entries.

```

<xsl:template name="t_vwContracts">
  <xsl:param name='grantid' />
  <xsl:param name='this' />
<!--      Creating a Grant-->
  <rdf:Description rdf:about="{ $baseURI }grant/grant{ $grantid }">
    <ufVivo:harvestedBy>DSR-Harvester</ufVivo:harvestedBy>
    <ufVivo:dateHarvested><xsl:value-of select="current-date()" /></ufVivo:dateHarvested>
    <rdf:type rdf:resource="http://vivoweb.org/ontology/core#Grant" />
    <rdf:type rdf:resource="http://vivoweb.org/ontology/core#Relationship" />
    <rdf:type rdf:resource="http://vivoweb.org/ontology/core#Agreement" />
    <ufVivo:psContractNumber><xsl:value-of select="$grantid" /></ufVivo:psContractNumber>
    <rdfs:label><xsl:value-of select="$this/db-dbo.vwContracts:Title" /></rdfs:label>
    <core:administeredBy>
<!--      Creating a department to match with or a stub if no match-->
      <rdf:Description rdf:about="{ $baseURI }org/org{ $this/db-dbo.vwContracts:
ContractDeptID }">
        <ufVivo:harvestedBy>DSR-Harvester</ufVivo:harvestedBy>
        <ufVivo:dateHarvested><xsl:value-of select="current-date()" /></ufVivo:
dateHarvested>
        <ufVivo:deptID><xsl:value-of select="$this/db-dbo.vwContracts:
ContractDeptID" /></ufVivo:deptID>
        <core:administers rdf:resource="{ $baseURI }grant/grant{ $grantid }" />
        <rdf:type rdf:resource="http://xmlns.com/foaf/0.1/Organization" />
      </rdf:Description>
    </core:administeredBy>
    <core:totalAwardAmount><xsl:value-of select="$this/db-dbo.vwContracts:TotalAwarded" /><
/core:totalAwardAmount>
    <core:sponsorAwardId><xsl:value-of select="$this/db-dbo.vwContracts:SponsorID" /></core:
sponsorAwardId>

```

```

        <xsl:choose>
            <xsl:when test="string($this/db-dbo.vwContracts:FlowThruSponsor) = ''">
                <core:grantAwardedBy>
                    <rdf:Description rdf:about="{ $baseURI }sponsor/sponsor{$this/db-
dbo.vwContracts:SponsorID}For{$grantid}">
                        <ufVivo:harvestedBy>DSR-Harvester</ufVivo:harvestedBy>
                        <ufVivo:dateHarvested><xsl:value-of select="current-
date()" /></ufVivo:dateHarvested>
                        <rdfs:label><xsl:value-of select="$this/db-dbo.
vwContracts:Sponsor"/></rdfs:label>
                        <core:awardsGrant rdf:resource="{ $baseURI }grant/grant
{$grantid}"/>
                        <rdf:type rdf:resource="http://xmlns.com/foaf/0.1
/Organization"/>
                        <core:sponsorAwardId><xsl:value-of select="$this/db-dbo.
vwContracts:SponsorID"/></core:sponsorAwardId>
                    </rdf:Description>
                </core:grantAwardedBy>
            </xsl:when>
            <xsl:otherwise>
                <core:grantSubcontractedThrough>
                    <rdf:Description rdf:about="{ $baseURI }sponsor/sponsor{$this/db-
dbo.vwContracts:SponsorID}For{$grantid}">
                        <ufVivo:harvestedBy>DSR-Harvester</ufVivo:harvestedBy>
                        <ufVivo:dateHarvested><xsl:value-of select="current-
date()" /></ufVivo:dateHarvested>
                        <rdfs:label><xsl:value-of select="$this/db-dbo.
vwContracts:Sponsor"/></rdfs:label>
                        <core:subcontractsGrant rdf:resource="{ $baseURI }grant
/grant{$grantid}"/>
                        <rdf:type rdf:resource="http://xmlns.com/foaf/0.1
/Organization"/>
                        <core:sponsorAwardId><xsl:value-of select="$this/db-dbo.
vwContracts:SponsorID"/></core:sponsorAwardId>
                    </rdf:Description>
                </core:grantSubcontractedThrough>
                <core:grantAwardedBy>
                    <rdf:Description rdf:about="{ $baseURI }sponsor/sponsor{$this/db-
dbo.vwContracts:FlowThruSponsorID}For{$grantid}">
                        <ufVivo:harvestedBy>DSR-Harvester</ufVivo:harvestedBy>
                        <ufVivo:dateHarvested><xsl:value-of select="current-
date()" /></ufVivo:dateHarvested>
                        <rdfs:label><xsl:value-of select="$this/db-dbo.
vwContracts:FlowThruSponsor"/></rdfs:label>
                        <core:awardsGrant rdf:resource="{ $baseURI }grant/grant
{$grantid}"/>
                        <rdf:type rdf:resource="http://xmlns.com/foaf/0.1
/Organization"/>
                        <core:sponsorAwardId><xsl:value-of select="$this/db-dbo.
vwContracts:SponsorID"/></core:sponsorAwardId>
                    </rdf:Description>
                </core:grantAwardedBy>
            </xsl:otherwise>
        </xsl:choose>

        <core:startDate rdf:datatype="http://www.w3.org/2001/XMLSchema#date">
            <xsl:analyze-string select="$this/db-dbo.vwContracts:BEGIN_DT" regex="^(....-..-
..).*?$">
                <xsl:matching-substring>
                    <xsl:value-of select="regex-group(1)"/>
                </xsl:matching-substring>
            </xsl:analyze-string>
        </core:startDate>
        <core:endDate rdf:datatype="http://www.w3.org/2001/XMLSchema#date">
            <xsl:analyze-string select="$this/db-dbo.vwContracts:END_DT" regex="^(....-..-
..).*?$">
                <xsl:matching-substring>
                    <xsl:value-of select="regex-group(1)"/>
                </xsl:matching-substring>
            </xsl:analyze-string>
        </core:endDate>

```

```

        </rdf:Description>
    </xsl:template>

```

"

## Project Team

The creation of PI and Co-PI stubs for matching. If they do match with the existing data they should get removed.

```

    <xsl:template name="t_vwProjectTeam">
        <xsl:param name='this' />
        <xsl:variable name='grantid' select='$this/db-dbo.vwProjectTeam:ContractNumber' />
        <xsl:choose>
            <xsl:when test="$this/db-dbo.vwProjectTeam:isPI = 'Y'">
<!--
                Creating the PI-->
                <rdf:Description rdf:about="{ $baseURI }piRole/inGrant{$grantid}For{$this/db-dbo.
vwProjectTeam:InvestigatorID}">
                    <ufVivo:harvestedBy>DSR-Harvester</ufVivo:harvestedBy>
                    <ufVivo:dateHarvested><xsl:value-of select="current-date()" /></ufVivo:
dateHarvested>
                    <rdf:type rdf:resource="http://vivoweb.org/ontology
/core#PrincipalInvestigatorRole"/>
                    <rdf:type rdf:resource="http://vivoweb.org/ontology
/core#InvestigatorRole"/>
                    <rdf:type rdf:resource="http://vivoweb.org/ontology/core#ResearcherRole"
/>
                    <core:roleIn>
                        <rdf:Description rdf:about="{ $baseURI }grant/grant{$grantid}">
                            <ufVivo:harvestedBy>DSR-Harvester</ufVivo:harvestedBy>
                            <ufVivo:dateHarvested><xsl:value-of select="current-
date()" /></ufVivo:dateHarvested>
                            <core:relatedRole rdf:resource="{ $baseURI }piRole/inGrant
{$grantid}For{$this/db-dbo.vwProjectTeam:InvestigatorID}" />
                            </rdf:Description>
                        </core:roleIn>
                        <core:principalInvestigatorRoleOf>
                            <rdf:Description rdf:about="{ $baseURI }person/person{$this/db-
dbo.vwProjectTeam:InvestigatorID}">
                                <ufVivo:harvestedBy>DSR-Harvester</ufVivo:harvestedBy>
                                <ufVivo:dateHarvested><xsl:value-of select="current-
date()" /></ufVivo:dateHarvested>
                                <rdfs:label><xsl:value-of select="$this/db-dbo.
vwProjectTeam:Investigator" /></rdfs:label>
                                <rdf:type rdf:resource="http://vivoweb.org/harvester
/excludeEntity" />
                                <ufVivo:ufid><xsl:value-of select="$this/db-dbo.
vwProjectTeam:InvestigatorID" /></ufVivo:ufid>
                                <rdf:type rdf:resource="http://xmlns.com/foaf/0.1
/Person"/>
                                <core:hasPrincipalInvestigatorRole rdf:resource="
{ $baseURI }piRole/inGrant{$grantid}For{$this/db-dbo.vwProjectTeam:InvestigatorID}" />
                                </rdf:Description>
                            </core:principalInvestigatorRoleOf>
                        </rdf:Description>
                    </xsl:when>
                    <xsl:otherwise>
<!--
                        Creating the Co-PI-->
                        <rdf:Description rdf:about="{ $baseURI }coPiRole/inGrant{$grantid}For{$this/db-
dbo.vwProjectTeam:InvestigatorID}">
                            <ufVivo:harvestedBy>DSR-Harvester</ufVivo:harvestedBy>
                            <ufVivo:dateHarvested><xsl:value-of select="current-date()" /></ufVivo:
dateHarvested>
                            <rdf:type rdf:resource="http://vivoweb.org/ontology
/core#CoPrincipalInvestigatorRole"/>
                            <rdf:type rdf:resource="http://vivoweb.org/ontology
/core#InvestigatorRole"/>
                            <rdf:type rdf:resource="http://vivoweb.org/ontology/core#ResearcherRole"
/>

```

```

<core:roleIn>
  <rdf:Description rdf:about="{ $baseURI }grant/grant{ $grantid }">
    <ufVivo:harvestedBy>DSR-Harvester</ufVivo:harvestedBy>
    <ufVivo:dateHarvested><xsl:value-of select="current-
date()" /></ufVivo:dateHarvested>
    <core:relatedRole rdf:resource="{ $baseURI }coPiRole
/inGrant{ $grantid }For{ $this/db-dbo.vwProjectTeam:InvestigatorID }"/>
  </rdf:Description>
</core:roleIn>
<core:co-PrincipalInvestigatorRoleOf>
  <rdf:Description rdf:about="{ $baseURI }person/person{ $this/db-
dbo.vwProjectTeam:InvestigatorID }">
    <ufVivo:harvestedBy>DSR-Harvester</ufVivo:harvestedBy>
    <ufVivo:dateHarvested><xsl:value-of select="current-
date()" /></ufVivo:dateHarvested>
    <rdfs:label><xsl:value-of select="$this/db-dbo.
vwProjectTeam:Investigator"/></rdfs:label>
    <rdf:type rdf:resource="http://vivoweb.org/harvester
/excludeEntity" />
    <ufVivo:ufid><xsl:value-of select="$this/db-dbo.
vwProjectTeam:InvestigatorID"/></ufVivo:ufid>
    <rdf:type rdf:resource="http://xmlns.com/foaf/0.1
/Person"/>
    <core:hasCo-PrincipalInvestigatorRole rdf:resource="
{ $baseURI }coPiRole/inGrant{ $grantid }For{ $this/db-dbo.vwProjectTeam:InvestigatorID }"/>
  </rdf:Description>
</core:co-PrincipalInvestigatorRoleOf>
</rdf:Description>
</xsl:otherwise>
</xsl:choose>
</xsl:template>
</xsl:stylesheet>

```