



Training on Next Generation Open Source Discovery Tool VuFind at BRAC University

5-6 January 2011

- **Installation of VuFind Step by Step**
- **Integration with E-Journals**
- **Integration with ILS (Koha)**
- **Integration with Institutional Repository (DSpace)**



Ayesha Abed Library

BRAC University

Dhaka, Bangladesh

<http://library.bracu.ac.bd/vufind>

- **Installation of VuFind Step by Step**
- **Integration with E-Journals**
- **Integration with ILS (Koha)**
- **Integration with Institutional Repository (DSpace)**

Installation of VuFind Step by Step

Software: VuFind 1.0.1

OS: Debian 6.0 (Squeeze)/Ubuntu 10.04

1. Update: `sudo apt-get update`

2. Install Apache HTTP Server:

```
sudo apt-get -y install apache2
sudo a2enmod rewrite
sudo /etc/init.d/apache2 force-reload
```

3. Installing PHP:

```
sudo apt-get -y install php5 php5-dev php-pear php5-ldap php5-mysql php5-xsl php5
pspell php5-gd
```

4. Installing Sun Java-6:

Add following two sources at '/etc/apt/sources.list' file:

```
deb http://ftp.us.debian.org/debian squeeze main contrib non-free
deb-src http://ftp.us.debian.org/debian squeeze main contrib non-free
```

Now install Java and associated packages:

```
apt-get install sun-java6-bin sun-java6-fonts sun-java6-jdk sun-java6-jre
```

5. Downloading the latest version of VuFind (1.0.1):

Install subversion: `sudo apt-get install subversion`

Download VuFind: `svn export --force`

```
https://vufind.svn.sourceforge.net/svnroot/vufind/releases/VuFind-1.0.1
/tmp/vf_download
```

Move it to default location: `sudo mv /tmp/vf_download /usr/local/vufind`

Apache needs to have write access to some of the VuFind subdirectories – these commands set that up:

```
sudo chown www-data:www-data /usr/local/vufind/web/interface/compile
sudo chown www-data:www-data /usr/local/vufind/web/interface/cache
sudo chown www-data:www-data /usr/local/vufind/web/images/covers/*
```

6. Linking VuFind to Apache:

Apache needs to have some extra VuFind settings loaded. Run this command to make Apache aware of VuFind's configuration file:

```
sudo ln -s /usr/local/vufind/httpd-vufind.conf /etc/apache2/conf.d/vufind
```

Apache needs to be restarted so the changes can take effect:

```
sudo /etc/init.d/apache2 reload
```

7. Installing VuFind:

Switch to installation directory: `cd /usr/local/vufind/` Edit the file 'install' and scroll down to the last portion of the 'install' file where Smarty template engine is being installed, and append this: `"-O Smarty-2.6.19.tar.gz"` to the line of downloading Smarty package. Here, note that this version of VuFind uses Smarty version 2.6.19, upper versions are found incompatible.

```
# Install Smarty into PHP Include Directory
PHPDIR=`pear config-get php_dir`
SMARTYDIR="$PHPDIR/Smarty"
if [ ! -d $SMARTYDIR ]
then
    wget http://www.smarty.net/do_download.php?download_file=Smarty-2.6.19.tar.gz
    -O Smarty-2.6.19.tar.gz
    tar -zxf Smarty-2.6.19.tar.gz
    mkdir $SMARTYDIR
    mv Smarty-2.6.19/libs/* $SMARTYDIR
    rm Smarty-2.6.19.tar.gz
    rm -rf Smarty-2.6.19
fi
```

Now install VuFind running the bash script: `sh ./install` [Assuming you are in `/usr/local/vufind/` directory]

Scripts for running and updating VuFind need to be made executable so they can run from the command line:

```
sudo chmod +x /usr/local/vufind/vufind.sh
```

```
sudo chmod +x /usr/local/vufind/import-marc.sh
```

8. Set Up Environment Variables:

Some environment variables need to be set so that VuFind-related scripts can find Java and VuFind itself. If you plan on running VuFind under a specific user account, you should set these only for that user. If you want to make the settings global for all accounts (the easiest, but not necessarily the best, approach), just run this code to add the necessary lines to the `/etc/profile` file:

```
sudo sh -c 'echo export PATH=$PATH:\"/usr/lib/jvm/java-6-sun/bin\" >> /etc/profile'
sudo sh -c 'echo export JAVA_HOME=\"/usr/lib/jvm/java-6-sun\" >> /etc/profile'
sudo sh -c 'echo export JRE_HOME=\"/usr/lib/jvm/java-6-sun/jre\" >> /etc/profile'
sudo sh -c 'echo export VUFIND_HOME=\"/usr/local/vufind\" >> /etc/profile'
```

After editing /etc/profile, you must reload it for the changes to take effect:
source /etc/profile

9. Configure VuFind:

Edit the /usr/local/vufind/web/conf/config.ini file to personalize VuFind for your library. The configuration file contains comments that should help you along the way.

Start VuFind in installation directory (i.e. /usr/local/vufind/): sh ./vufind start
Browse: <http://hostname/vufind/>

10. Troubleshooting some installation problems

Even you've completed the installation, some problems may arise. If something went wrong, you may see a blank page. To resolve this follow these steps:

- a. Edit 'httpd-vufind.conf' file and uncomment these lines if you wish to show all errors on the screen.
php_value display_errors 1
php_value error_reporting 2047
- b. Set 'debug = false' to 'debug = true' in ../web/conf/config.ini file

Now browse <http://hostname/vufind/> to trace down the errors. Most of the people face this error: Warning: require_once(Smarty/Smarty.class.php): failed to open stream: No such file or directory in /usr/local/**vufind**/web/sys/Interface.php on line 21 Fatal error: require_once(): Failed opening required 'Smarty/Smarty.class.php' (include_path='.:usr/share/php:usr/share/pear') in /usr/local/**vufind**/web/sys/Interface.php on line 21...

If this is your case then the Smarty library is not installed. Follow the last few steps of installation process:

```
tar -zxf Smarty-2.6.19.tar.gz
mv Smarty-2.6.19/libs/* /usr/share/php/Smarty/
```

Integration with E-Journals

1. We've converted all formats i.e. .xls, csv into MARC21 format using **MarcEdit** to upload into VuFind.
2. Make a copy of ./import-marc.sh, like ./import-marc-ejrnل.sh and edit this line to add another properties file:
PROPERTIES_FILE="\$VUFIND_HOME/import/import_ejrnل.properties"
3. Make a copy of "./import/import.properties" to "./import/import_ejrnل.properties" and edit the field "solr.indexer.properties = marc.properties, marc_local_ejrnل.properties"
4. Make a copy of "./import/marc_local.properties" to "./import/marc_local_ejrnل.properties" and add/uncomment following contents:

```
collection = "Catalog"  
institution = "BRAC University"  
building = "Ayesha Abed Library"  
recordtype = "ejournal"  
title = 245ab, first  
title_short = 245ab, first  
publisher = 260b  
format = script(format.bsh), getFormat, format_map.properties
```

Here, recordtype = "ejournal" defines which record driver should be triggered to handle Ejournal's marc data type. So, create another record driver at ./web/RecordDrivers/EjournalRecord.php to add this class definition:

```
class EjournalRecord extends MarcRecord {  
    public function getCoreMetadata()  
    {  
        global $configArray;  
        global $interface;  
  
        // Assign required variables (some of these are also used by templates for  
        // tabs, since every tab can assume that the core data is already assigned):  
        $this->assignTagList();  
        $interface->assign('isbn', $this->getCleanISBN()); // needed for covers  
        $interface->assign('recordFormat', $this->getFormats());  
        $interface->assign('recordLanguage', $this->getLanguages());  
  
        // These variables are only used by the core template, and they are prefixed  
        // with "core" to prevent conflicts with other variable names.
```

```

$interface->assign('coreShortTitle', $this->getShortTitle());
$interface->assign('coreSubtitle', $this->getSubtitle());
$interface->assign('coreTitleStatement', $this->getTitleStatement());
$interface->assign('coreTitleSection', $this->getTitleSection());
$interface->assign('coreNextTitles', $this->getNewerTitles());
$interface->assign('corePrevTitles', $this->getPreviousTitles());
$interface->assign('corePublications', $this->getPublicationDetails());
$interface->assign('coreEdition', $this->getEdition());
$interface->assign('coreSeries', $this->getSeries());
$interface->assign('coreSubjects', $this->getAllSubjectHeadings());

// Added from getExtendedMetadata ()
$interface->assign('extendedAccess', $this->getAccessRestrictions());
$interface->assign('extendedNotes', $this->getGeneralNotes());
$interface->assign('extendedDateSpan', $this->getDateSpan());
$interface->assign('extendedISBNs', $this->getISBNs());
$interface->assign('extendedISSNs', $this->getISSNs());
$interface->assign('extendedFrequency', $this->getPublicationFrequency());

// Only display OpenURL link if the option is turned on and we have
// an ISSN. We may eventually want to make this rule more flexible,
// but for now the ISSN restriction is designed to be consistent with
// the way we display items on the search results list.
$hasOpenURL = ($this->openURLActive('record') && $this->getCleanISSN());
if ($hasOpenURL) {
    $interface->assign('coreOpenURL', $this->getOpenURL());
}

// Only load URLs if we have no OpenURL or we are configured to allow
// URLs and OpenURLs to coexist:
if (!isset($configArray['OpenURL']['replace_other_urls']) ||
    !$configArray['OpenURL']['replace_other_urls'] || !$hasOpenURL) {
    $interface->assign('coreURLs', $this->getURLs());
}

// The secondary author array may contain a corporate or primary author;
// let's be sure we filter out duplicate values.
$mainAuthor = $this->getPrimaryAuthor();
$corpAuthor = $this->getCorporateAuthor();
$secondaryAuthors = $this->getSecondaryAuthors();
$duplicates = array();
if (!empty($mainAuthor)) {
    $duplicates[] = $mainAuthor;
}
if (!empty($corpAuthor)) {
    $duplicates[] = $corpAuthor;
}

```

```

}
if (!empty($duplicates)) {
    $secondaryAuthors = array_diff($secondaryAuthors, $duplicates);
}
$interface->assign('coreMainAuthor', $mainAuthor);
$interface->assign('coreCorporateAuthor', $corpAuthor);
$interface->assign('coreContributors', $secondaryAuthors);

// Assign only the first piece of summary data for the core; we'll get the
// rest as part of the extended data.
$summary = $this->getSummary();
$summary = count($summary) > 0 ? $summary[0] : null;
$interface->assign('coreSummary', $summary);

// Send back the template name:
return 'RecordDrivers/Index/core.tpl';
}

public function getExtendedMetadata()
{
    global $interface;

    $interface->assign('extendedSummary', $this->getSummary());
    $interface->assign('extendedRelated', $this->getRelationshipNotes());
    $interface->assign('extendedPhysical', $this->getPhysicalDescriptions());
    $interface->assign('extendedPlayTime', $this->getPlayingTimes());
    $interface->assign('extendedSystem', $this->getSystemDetails());
    $interface->assign('extendedAudience', $this->getTargetAudienceNotes());
    $interface->assign('extendedAwards', $this->getAwards());
    $interface->assign('extendedCredits', $this->getProductionCredits());
    $interface->assign('extendedBibliography', $this->getBibliographyNotes());
    $interface->assign('extendedFindingAids', $this->getFindingAids());

    return 'RecordDrivers/Index/extended.tpl';
}

public function getSearchResult () {
    global $interface;

    $template = parent :: getSearchResult ();
    $interface -> assign ('summAjaxStatus', false);
    $interface -> assign ('summPublisher', $this->getPublishers());
    $interface -> assign ('summNotes', $this->getGeneralNotes());
    return $template;
}
}

```


This class has been extended from existing MarcRecord class and overrides three methods. We are moving up some extended data from 'Description' tab to the record details view. getExtendedMetada () holds the Description tab's data and getCoreMetadata () holds the data of record view. Here, "\$interface -> assign ('summAjaxStatus', false)" defines not to show the "Call Number" and "Location" for particular types of marc records used for ejournals.

The main parent class is 'IndexRecord.php', modify the getSearchResult () method of IndexRecord.php:

```
public function getSearchResult () {  
    .....  
    .....  
    $interface -> assign ('summAjaxStatus', false);  
    return 'RecordDrivers/Index/result.tpl';  
}
```

5. Now edit ./web/interface/themes/default/RecordDrivers/Index/core.tpl file to display the extended records. Basically, just pick those logics from extended.tpl file.

6. Open the file ../import/index_scripts/format.bsh in your favorite editor, this file determines the format type. Find this piece of code that actually at the beginning, and move this to the end of the script:

```
// check if there's an h in the 245  
if (title != null) {  
    if (title.getSubfield('h') != null){  
        if (title.getSubfield('h').getData().toLowerCase().contains("[electronic  
resource]")) {  
            result.add("Electronic");  
            return result;  
        }  
    }  
}
```

Just before this one:

```
// Nothing worked!  
if (result.isEmpty()) {  
    result.add("Unknown");  
}
```

The format 'Electronic' will be shown when no other specific format matches for a particular electronic resource.

7. The format type is stored at the LEADER field. Modify the following block of codes like follows:

```

// check the Leader at position 7
leaderBit = leader.charAt(7);
switch (Character.toUpperCase(leaderBit)) {
    // Monograph
    case 'M':
        if (formatCode == 'C') {
            result.add("eBook");
        } else {
            result.add("Book");
        }
        break;
    // Serial
    case 'S':
        // Look in 008 to determine what type of Continuing Resource
        formatCode = fixedField.getData().toUpperCase().charAt(21);
        switch (formatCode) {
            case 'N':
                result.add("Newspaper");
                break;
            case 'P':
                eJrnl = fixedField.getData().toUpperCase().charAt(23);
                switch (eJrnl) {
                    case 'O':
                        result.add("eJournal");
                        break;
                    default:
                        result.add("Journal");
                        break;
                }
                break;
            default:
                result.add("Serial");
                break;
        }
    }
}
if (!result.isEmpty()) {
    return result;
}

```

8. Now edit `../import/translation_maps/format_map.properties` file and add a new format type: `eJournal = eJournal`

9. To show the associated icon, edit the file: `"../web/interface/themes/[your chosen theme]/css/styles.css"` and the CSS class called `ebook`:

```
.ebook {
```

```
.....  
}
```

To use this for your new eJournal class, just add .ejournal to the declaration:

```
.ebook, .ejournal {  
.....  
}
```

10. Now make necessary changes in ../web/interface/themes/[your theme]/RecordDrivers/Index/result.tpl file to show the result in search page.

11. Import marc data: sh ./import-marc-ejrnl.sh <your/path/to/marc>

Integration with ILS (Koha)

1. Check the following setting in ../web/conf/config.ini file under these sections, modify if necessary:

```
[Site]  
path          = /vufind  
url           = http://library.bracu.ac.bd/vufind  
local        = /usr/local/vufind/web  
email        = yourid@domain.com  
title        = "Library Catalog"  
theme        = default  
language     = en ; default -- more options available in [Languages] below.  
locale       = en_US  
timezone     = "Asia/Dhaka"  
defaultModule      = Search  
defaultRecordTab  = Holdings
```

```
[Catalog]  
driver      = Koha
```

```
[Authentication]  
method     = ILS
```

```
[Database]  
database    = mysql://mysqlusername:mysqlpassword@hostname/vufind  
schema_location = /usr/local/vufind/web/conf  
class_location = /usr/local/vufind/web/services/MyResearch/lib
```

```
require_prefix =
class_prefix =
debug          = 0
```

[Content]

```
; You can select from Syndetics, LibraryThing, Amazon, OpenLibrary or Google Books
coverimages      = Google      ;We are using google images
authors          = Wikipedia
```

[Proxy]

```
;Leave this section if you are not behind a proxy
host = your.proxy.server
port = 8080
```

2. Copy the driver Koha.php (available at:

<https://vufind.svn.sourceforge.net/svnroot/vufind/trunk/web/Drivers/Koha.php>) to
../web/Drivers/ directory.

3. Also copy the 'Koha.ini' (available at:

<https://vufind.svn.sourceforge.net/svnroot/vufind/trunk/web/conf/Koha.ini>) to your
/usr/local/vufind/web/conf/ directory.

4. Uncomment/add following lines at ../import/marc_local.properties file:

```
id = 999c, first
collection = "Catalog"
institution = "BRAC University"
building = "Ayesha Abed Library"
```

5. In our current configuration, 'checked out', 'not for loan' etc that are not available for circulating, are shown as 'Unavailable (click title for details)' in the vufind's main search page. Users will see per item's details after clicking on the item's title. To show this, edit ../web/interface/themes/[your theme]/Search/list-list.tpl file.

6. In our library, a user can place a hold only on 'checked out' items. Edit ../web/interface/themes/[your theme]/Record/view-holdings.tpl file to meet this specific requirement.

7. Export *.marc data from Koha

8. Import marc data: sh ./import-marc.sh <your/path/to/marc>

Integration with Institutional Repository (DSpace)

1. The associated tools for harvesting XML metadata from DSpace are available at current development trunk: <https://vufind.svn.sourceforge.net/svnroot/vufind/trunk/> To include this tool do followings:

- a. Replace the current ../harvest directory with that one from development trunk.
- b. Also copy other files/directories from ../import/ directory of development trunk to your ../import directory:

```
lib
xsl
import-xsl.php
dspace.properties
ojs.properties
vudl.properties
```

2. Edit the ../import/dspace.properties file:

```
[Parameters]
institution = "My University"
collection = "Digital Repositories"
```

3. Open ../import/xsl/dspace.xsl file in an editor and check/edit as followings:

```
<field name="format">eResource</field>
```

```
<!-- RECORDTYPE -->
<field name="recordtype">dspace</field>
```

```
<!-- FORMAT -->
<!-- populating the format field with dc.type instead, see TYPE below.
if you like, you can uncomment this to add a hard-coded format
in addition to the dynamic ones extracted from the record.
```

```
-->
<field name="format">eResource</field>
```

If you want all the DSpace data under a single format above (eResource), then comment out this section:

```
<!-- TYPE -->
<!-- No more dynamic type, all are under single type
<xsl:if test="//dc:type">
<field name="format">
    <xsl:value-of select="//dc:type" />
</field>
</xsl:if>
-->
```

This is used for harvester to dynamically extract the associated format of each data.

4. Since the record type has been defined as 'dspace', it'll look for 'DspaceRecord.php' record driver in ../web/RecordDriver/ directory. Create a DspaceRecord.php and add following contents:

```
<?php
require_once 'File/MARC.php';
require_once 'RecordDrivers/IndexRecord.php';

class DspaceRecord extends IndexRecord {

    public function getSearchResult () {
        global $interface;

        $template = parent :: getSearchResult ();
        $interface -> assign ('summAjaxStatus', false);
        //Don't show Callnumber and Location
        $interface -> assign ('summDate', false);
        //Don't show date
        $interface -> assign ('summPublisher', $this->getPublishers());
        //Show publisher name
        $interface -> assign ('summNotes', false);
        //Preventing to show any general note
        return $template;
    }
}
?>
```

5. Edit ../harvest/oai.ini file to add following contents:

```
[DSpace]
url = http://123.49.46.157:8180/oai/request
metadataPrefix = oai_dc
idSearch[] = "^/oai:dspace.bracu.ac.bd:/"
idReplace[] = "ir-"
idSearch[] = "^/"
idReplace[] = "-"
injectDate = "datestamp"
injectId = "identifier"
dateGranularity = auto
harvestedIdLog = harvest.log
```

6. Now you can harvest metadata from DSpace. Change to ../harvest directory and run 'php harvest_oai.php'. It'll create a folder as the section name (i.e. DSpace) at oai.ini file

and save all *.xml files inside that directory. To import a single data run 'php ./import-xsl.php ../harvest/Dspace/<filename>.xml ./dspace.properties. If you want to import all the data then switch to ../harvest/ directory and run 'sh ./batch-import-xsl.sh /Dspace ../import/dspace.properties' This will import all the data inside the 'Dspace' directory and put all processed files under 'processed' directory of 'Dspace'. If can delete those if you wish. For more information, go to: http://vufind.org/wiki/other_than_marc. Remember to start vufind before importing xml data.