

# Software Release Notes for 5.0

Jeffery A. Triggs, Sho Nakagama, & Isaiah Beard

August 17, 2009

## 1 Introduction

This document supersedes relevant sections of the one entitled “Software Release Procedures for 1.1” dated February 8, 2006. It reflects our our migration to a new platform, including Fedora 3.0, PHP 5, and MySQL 5.

The following sections describe: 1) the base software platform, which should already be in place before any release of the project software; 2) the different projects to be released along with their functions.

## 2 Basic Software Platform Requirements

This section deals with the underlying software platforms necessary for the 5.0 release. The software is developed on Linux at the SCC and released on Solaris at Library Systems. The basic software packages are compiled and loaded separately on each of these platforms, and should remain stable throughout the 5.x release cycle.

### 2.1 amberfish 1.6.4

#### 2.1.1 xerces 2.7

### 2.2 Apache

Server version: Apache/2.0.59

Server compiled with....

```
-D APACHE_MPM_DIR="server/mpm/prefork"
```

```
-D APR_HAS_SENDFILE
-D APR_HAS_MMAP
-D APR_HAVE_IPV6 (IPv4-mapped addresses enabled)
-D APR_USE_SYSVSEM_SERIALIZE
-D APR_USE_PTHREAD_SERIALIZE
-D SINGLE_LISTEN_UNSERIALIZED_ACCEPT
-D APR_HAS_OTHER_CHILD
-D AP_HAVE_RELIABLE_PIPED_LOGS
-D HTTPD_ROOT="/usr/local/apache2"
-D SUEXEC_BIN="/usr/local/apache2/bin/suexec"
-D DEFAULT_PIDLOG="logs/httpd.pid"
-D DEFAULT_SCOREBOARD="logs/apache_runtime_status"
-D DEFAULT_LOCKFILE="logs/accept.lock"
-D DEFAULT_ERRORLOG="logs/error_log"
-D AP_TYPES_CONFIG_FILE="conf/mime.types"
-D SERVER_CONFIG_FILE="conf/httpd.conf"

./configure
--enable-ssl
--enable-so
--with-mpm=prefork
--enable-mods-shared=most
--with-openssl=/usr
```

## 2.3 CNRI Handles

The CNRI handle client is run typically on ingest. The location of the handle.jar file is defined in the incs file. On lefty64 it is currently /usr/local/lib/handle.jar.

## 2.4 Configuration Include Files

The special directory “/mellon/includes” contains a set of files used to configure the system resources and other system-specific variables used by dlr/EDIT, RU-core programs, and other programs working with the created Fedora repository.

```
classes (directory of PHP classes discussed below)
incs (editable language-neutral includes file base)
incs.php (PHP includes file)
```

```
incs.pl (Perl includes file)
incs.sh (shell includes file)
makeincs.pl (generates the language-specific includes files)
perlmodules (directory of Perl modules discussed below)
```

The incs file, maintained by the local system administrator, has four main sections:

```
## pathnames and special tmp file locations
## user names and passwords
## special parameters
## needed utilities listed alphabetically
```

The new section of “special parameters” is used currently for two settings, e.g.: `$darwintimeout = 3`; (the length a video link is left visible on the darwin server expressed as multiples of the duration of the video, or 60 seconds for “0”) and `$shibauth = "no"`; (“yes” or “no” indicating whether or not Shibboleth authentication is being used).

## 2.5 Crons

We will need to run four cron scripts. The amberfish indexing is done by two scripts run as a nightly cron: `$filebase/dlr/EDIT/getfedorasql.php`, which reads data from the current Fedora database and populates the fedsearch database with the title, collection membership, and other information about the Fedora objects, and `$filebase/dlr/EDIT/doamberindexsql.php`, which reads the fedsearch database and creates or updates modular indexes as needed based on collection membership. There are two other cron scripts that need to be run nightly. The first is `rnsignatures.sh` in `$filebase/dlr/SIGS`. The second is `gsitemap.sh` in `$filebase/dlr/EDIT`, which recreates `$serverbase/dlr/sitemap.xml` based on the the current objects associated with collections that have been marked for export to outside search engines. There should be a `robots.txt` file in `$serverbase/dlr` with the following line:

```
Sitemap: http://\$serverbase/dlr/sitemap.xml
```

This allows “autodiscovery” of the `sitemap.xml` file by other search engines.

## 2.6 CVS 1.12.12

## 2.7 DjVu DocumentExpress 4.1

## 2.8 DjVuLibre 3.5.17

## 2.9 Fedora 3.0

The current version of Fedora at the SCC is 3.0. Our 5.0 release will involve the move to Fedora 3.0 among other major changes. As in previous releases, Fedora runs under a manually created “fedora” user and group. Fedora 3.0 has the following software requirements:

```
J2SDK 1.6
JAVA\_HOME and FEDORA\_HOME environment variables
```

The /mellon/FEDORA-ENV script sets the following values for the Fedora environment:

```
JAVA\_ROOT=/mellon/java
JAVA\_HOME=/mellon/java
JAVA\_BINDIR=/mellon/java/bin
FEDORA\_HOME=/mellon
FEDORA\_JAVA\_HOME=/mellon/java
JDK\_HOME=/mellon/java/jvm/java
JRE\_HOME=/mellon/java/jvm/java/jre
CATALINA\_HOME=/mellon/tomcat
```

### 2.9.1 Customizations of Fedora 3.0 for the 5.0 Release

There are a number of customizations of the Fedora configuration that we need for the 5.0 release. We have included a file called r50-fedora.tar in the \$CVSROOT directory on lefty64, /mellon/cvsroot. This tar file contains copies of the current lefty64 configuration file, a file of only those elements of the default configuration requiring customization, our current expanded fedora users file, and a directory tree with the repository XACML policies that we have customized for lefty64. Note: the lefty64 password has been disguised in these files.

```
ls -l r50-fedora
total 76
```

```
-rw-r--r-- 1 8267 2009-05-21 15:55 changes-fedora.fcfg
drwxr-xr-x 3 4096 2009-05-21 15:55 example-repository-policies
-rw-r--r-- 1 50085 2009-05-21 15:50 fedora.fcfg
-rw-r--r-- 1 1836 2009-05-21 15:50 fedora-users.xml
```

## 2.10 ghostscript 8.15.3 (2006-04-19)

## 2.11 ImageMagick 6.2.0

## 2.12 Java 1.6

## 2.13 javabridge

```
autoconf-2.53
automake-1.9
libtool-1.5.22 (libtool-1.5.20 minimum)
```

```
source FEDORA-ENV
phpize
./configure
make
make install
```

## 2.14 LAME 64bits version 3.98.2

## 2.15 mp3wrap 0.5

## 2.16 MySQL Ver 14.12 Distrib 5.0.26, for suse-linux-gnu (x86\_64)

Note: anything above MySQL 5 will work, as long as it is MySQL-Max with innodb support. The version for Solaris is 5.0.77.

## 2.17 Perl 5.8.8 built for x86\_64-linux-thread-multi

Perl should have the following modules installed<sup>1</sup>:

---

<sup>1</sup>Note: the SOAP::Lite module used in dlr/EDIT is version 0.60 on lefty and mss3 and version 0.69 on lefty64.

```
CGI ($CGI::revision = '$Id: CGI.pm,
v 1.194 2005/12/06 22:12:56 lstein Exp $';
$CGI::VERSION='3.15';)
CGI::Carp ($VERSION = '1.04';)
Data::Dump ($VERSION = '2.121_08';)
HTTP::Request ($Id: Request.pm,v 1.40)
LWP::Simple ($Id: Simple.pm,v 1.41)
LWP::UserAgent ($Id: UserAgent.pm,v 2.33)
MIME::Base64 ($Id: Base64.pm,v 3.11)
SOAP::Lite ($Id: Lite.pm,v 1.43, $VERSION = '0.69';)
Storable ($VERSION = '2.15';)
XML::Parser (require XML::Parser::Expat; $VERSION = '2.34';)
XML::Parser::Expat ($VERSION = "2.34" ;)
XML::XPath ($VERSION = '1.13';)
XML::XPath::XMLParser ($Id: XMLParser.pm,v 1.49)
```

There is a directory of Perl modules (/mellon/includes/perlmodules) containing stubs for handling the complete set of Fedora 3.0 API-M and API-A services.

## 2.18 PHP 5.2.6

The PHP build options include

```
PHP Version => 5.2.6
Configure Command => './configure'
'--with-libdir=lib64'
'--with-apxs2=/usr/local/apache2/bin/apxs'
'--with-openssl=/usr'
'--with-zlib'
'--enable-exif'
'--with-gettext'
'--with-ldap=/usr'
'--with-mcrypt'
'--with-mysql=/usr'
'--enable-soap'
'--enable-sockets'
'--with-xmlrpc'
'--with-xsl=/usr'
'--enable-zip'
```

```
'--with-pear'  
'--enable-ftp'  
'--enable-mbstring'  
'--with-mimemagic'  
'--with-mysqli'  
'--with-curl=/usr'
```

There is a directory of PHP classes (/mellon/includes/classes) containing classes for handling pdfMaker, mysql, and the collection hierarchy. The custom PHP ini configure options include

- max\_execution\_time 300
- max\_input\_time 300
- post\_max\_filesize 24M
- upload\_max\_filesize 512M

Some of these system wide defaults need to be over-ridden for the WMS directory. Therefor the following lines should be added to the Apache httpd.conf file:

```
<Directory /home/httpd/html/dwms>  
    php_value max_execution_time 1800  
    php_value max_input_time 1200  
    php_value post_max_size 512M  
</Directory>
```

The following Parameter Settings are in effect on our system:

```
allow_url_include Off  
register_argv/argc On
```

## 2.19 SSH

The 5.0 release will require SSH for interaction between the machine hosting Fedora and the machine hosting the darwin server used for streaming videos. The following section describes how SSH should be set up.

Requirements: OpenSSH

STEP 1: Generate a local RSA key. This is done by logging into a terminal session as the user running the fedora processes (www-user? fedora?), and issuing the following command:

```
# ssh-keygen -t rsa
```

You will be asked for a path to place the generated keys. The default path that is offered (typically `~/home/$USER/.ssh`) is sufficient.

You will be asked for a password. JUST PRESS ENTER, for no password. This will generate a passwordless encryption key.

Two key files, a public a private key, will be generated in your `/.ssh` subdirectory called `id_rsa` and `id_rsa.pub`. These files should be kept in this location.

STEP 2: The public key needs to be placed on the darwin streaming server. A copy of the `id_rsa.pub` file must now be given to the system administrator. The sysadmin will `sftp` in to the Darwin server as the `fedora` user, and upload the provided `/.ssh/id_rsa.pub` file to the appropriate home directory on darwin

```
# sftp fedora@darwin.scc-net.rutgers.edu
```

```
[User is authenticated using an existing, approved rsa key]
```

```
> put id\_rsa.pub  
> quit
```

STEP 3: Sysadmin logs into a terminal session on Darwin. The new key is appended to an existing rsa keyring:

```
# cat ~/id\_rsa.pub >> ~/.ssh/authorized\_keys
```

Upon completion of these steps, the FEDORA-authorized user on the repository server (`mss3`) will be permitted to issue commands, and transfer files on the Darwin streaming server.

Notes:

1. File transfers and commands given will remain encrypted while in transit
2. Only authorized "users" that present a private RSA key that corresponds to the public key on the server will be granted access. Unauthorized users without an RSA key or presented an invalid/revoked RSA key will be prompted to enter a password. This is a red herring: the password has been scrambled for this user on the darwin server.

## 2.20 UNIX Utilities

The following UNIX utilities are used. Their pathname locations are set in the `/mellon/includes/incs` file. The versions listed reflect the current utilities on lefty64.



```
awk (GNU Awk 3.1.5)
cat (cat (GNU coreutils) 6.4)
chmod (chmod (GNU coreutils) 6.4)
cp (cp (GNU coreutils) 6.4)
diff (diff (GNU diffutils) 2.8.7)
echo (echo (GNU coreutils) 6.4)
egrep (egrep (GNU grep) 2.5.1)
find (GNU find version 4.2.28)
Features enabled:
D_TYPE O_NOFOLLOW(enabled)
LEAF_OPTIMISATION)
head (head (GNU coreutils) 6.4)
ls (ls (GNU coreutils) 6.4)
lynx (Lynx Version 2.8.6rel.4 (15 Nov 2006)
libwww-FM 2.14,
SSL-MM 1.4.1,
OpenSSL 0.9.8d,
ncurses 5.5.20060513(wide))
mv (mv (GNU coreutils) 6.4)
pwd (pwd (GNU coreutils) 6.4)
rm (rm (GNU coreutils) 6.4)
scp (OpenSSH_4.4p1, OpenSSL 0.9.8d 28 Sep 2006)
sed (GNU sed version 4.1.5)
shasum (shasum (GNU coreutils) 6.4)
sort (sort (GNU coreutils) 6.4)
ssh (OpenSSH_4.4p1, OpenSSL 0.9.8d 28 Sep 2006)
tail (tail (GNU coreutils) 6.4)
tar (tar (GNU tar) 1.15.1)
uniq (uniq (GNU coreutils) 6.4)
vis (delivered by us)
wc (wc (GNU coreutils) 6.4)
xargs (GNU xargs version 4.2.28)
```

Note: vis must be compiled from the vis.c program in \$filebase/dlr/EDIT. The command is:

```
gcc -o vis vis.c
```

While we have usually left the vis binary in \$filebase/dlr/EDIT and recompiled it with new releases, it is referenced with the \$VIS variable in incs and could be

stored in a less volatile location such as `/usr/local/bin`.

## 2.21 Repository-wide XACML Policies

Repository-wide XACML Policies, which should not be confused with XACML policies for individual objects, are part of our basic configuration of Fedora beginning with the 5.0 release using Fedora 3.0. Our Fedora administrator will deliver a set of sample policies controlling access to the Fedora APIs. These will need to be re-configured to work with a given Fedora repository. Note: backups of these sample XACML policies as well as the samples delivered with the Fedora distribution should not be kept in the Fedora policies directories (defined in the `fedora.fcfg` file in Fedora's config directory).

## 2.22 xsltproc

Using `libxml 20630`, `libxslt 10124` and `libexslt 813` `xsltproc` was compiled against `libxml 20630`, `libxslt 10124` and `libexslt 813` `libxslt 10117` was compiled against `libxml 20630` `libexslt 813` was compiled against `libxml 20630`

# 3 Identification of Projects

## 3.1 dlr/EDIT/notification

This is the main Fedora interface. The `dlr` directory contains the public search functions for various portals, e.g., `NJDH` and `RUcore`, with view only Fedora links. The `EDIT` directory is password protected and contains the management functions, including ingesting objects, indexing, searching, object editing, management of the collections database, handle creation, object validation, alerting services, and a new statistics reporting tool. It requires the following software platform elements: `PHP`, `Perl`, `amberfish`, `xsltproc`, `MySQL`, and various `UNIX` utilities. It uses the authentication database and `auth.class.php` located in the `doc_root` directory. The notification directory contains alerting and statistics reporting functions.

### **3.2 disseminators**

This is a group of Fedora disseminators that perform functions such as creating tables of contents for books and other multi-divisional objects, listing objects in collections, and outputting flash viewers and SMIL files for Quicktime movies. It uses PHP and Perl and makes use of the PHP XPath class located in `doc_root/phpxpath`.

### **3.3 WMS**

This is the main interface for creating the metadata needed by Fedora objects, creating presentation datastreams from archival files in a “pipeline”, and creating the external XML objects used for ingestion into Fedora. It uses MySQL, PHP, Perl, the DjVu utilities, ImageMagick, the ghostscript utilities, lame, sha1sum, tar, and other UNIX utilities.