

# Upgrading a DSpace Installation

## DSpace System Documentation: Upgrading a DSpace Installation

This section describes how to upgrade a DSpace installation from one version to the next. Details of the differences between the functionality of each version are given in the [Version History](#) section.

### Test Your Upgrade Process

In order to minimize downtime, it is always recommended to first perform a DSpace upgrade using a Development or Test server. You should note any problems you may have encountered (and also how to resolve them) before attempting to upgrade your Production server. It also gives you a chance to "practice" at the upgrade. Practice makes perfect, and minimizes problems and downtime. Additionally, if you are using a version control system, such as subversion or git, to manage your locally developed features or modifications, then you can do all of your upgrades in your local version control system on your Development server and commit the changes. That way your Production server can just checkout your well tested and upgraded code.

If you are upgrading across multiple versions

You should perform all of the steps of each upgrade between the version from which you are starting and the version to which you are upgrading. You do not need to install each intervening version, but you do need to carry out all of the configuration changes and additions, and all of the database updates, for each one. For example, when upgrading from 1.5.x to 1.7.x, you need to perform the configuration & database upgrade steps detailed in [#Upgrading from 1.5.x to 1.6.x](#) followed by those detailed in [#Upgrading from 1.6.x to 1.7.x](#).

- [DSpace System Documentation: Upgrading a DSpace Installation](#)
  - [Upgrading from 1.7 to 1.7.x](#)
  - [Upgrading from 1.6.x to 1.7.x](#)
  - [Upgrading from 1.6 to 1.6.x](#)
  - [Upgrading from 1.5.x to 1.6.x](#)
  - [Upgrading From 1.5 or 1.5.1 to 1.5.2](#)
  - [Upgrading From 1.4.2 to 1.5](#)
  - [Upgrading From 1.4.1 to 1.4.2](#)
  - [Upgrading From 1.4 to 1.4.x](#)
  - [Upgrading From 1.3.2 to 1.4.x](#)
  - [Upgrading From 1.3.1 to 1.3.2](#)
  - [Upgrading From 1.2.x to 1.3.x](#)
  - [Upgrading From 1.2.1 to 1.2.2](#)
  - [Upgrading From 1.2 to 1.2.1](#)
  - [Upgrading From 1.1 \(or 1.1.1\) to 1.2](#)
  - [Upgrading From 1.1 to 1.1.1](#)
  - [Upgrading From 1.0.1 to 1.1](#)

## Upgrading from 1.7 to 1.7.x

In the notes below [dspace] refers to the install directory for your existing DSpace installation, and [dspace-source] to the source directory for DSpace 1.7.1. Whenever you see these path references, be sure to replace them with the actual path names on your local system. Additionally, be sure to backup your configs, source code modifications, and database before doing a step that could destroy your instance.

1. **Download DSpace 1.7.1** Either download DSpace 1.7.1 from [DSpace.org](#) or check it out directly from the [SVN code repository](#). If you downloaded DSpace do not unpack it on top of your existing installation. Refer to [Installation Instructions, Step 3](#) for unpacking directives.
2. **Apply any customizations.** If you have made any local customizations to your DSpace installation they will need to be migrated over to the new DSpace. These are housed in one of the following places:
  - JSPUI modifications: [dspace-source]/dspace/modules/jspui/src/main/webapp/
  - XMLUI modifications: [dspace-source]/dspace/modules/xmlui/src/main/webapp/
  - Config modifications: [dspace]/config
3. **Build DSpace.** Run the following commands to compile DSpace:

```
cd [dspace-source]/dspace/  
mvn -U clean package
```

You will find the result in [dspace-source]/dspace/target/dspace-[version]-build.dir. Inside this directory is the compiled binary distribution of DSpace. Before rebuilding DSpace ('package'), the above command will clean out any previously compiled code ('clean') and ensure that your local DSpace JAR files are updated from the remote maven repository.

4. **Stop Tomcat.** Take down your servlet container. For Tomcat, use the \$CATALINA/shutdown.sh script. (Many Unix-based installations will have a startup/shutdown script in the /etc/init.d or /etc/rc.d directories.)
5. **Backup Your DSpace.** Make a complete backup of your system, including:
  - A snapshot of the database. To have a "snapshot" of the PostgreSQL database use Postgres' [pg\\_dump](#) command.
  - The asset store ([dspace]/assetstore by default, and any other assetstores configured in the [dspace]/config/dspace.cfg "assetstore.dir" and "assetstore.dir.#" settings)
  - Your configuration files and customizations to DSpace (including any customized scripts).
6. **Update DSpace.** Update the DSpace installed directory with the new code and libraries. Issue the following commands:

```
cd [dspace-source]/dspace/target/dspace-[version]-build.dir  
ant -Dconfig=[dspace]/config/dspace.cfg update
```

7. **Generate Browse and Search Indexes.** Though there are not any database changes between 1.7 and 1.7.1 release, it makes good policy to rebuild your search and browse indexes when upgrading to a new release. To do this, run the following command from your DSpace install directory (as the dspace user):

```
[dspace]/bin/dspace index-init
```

8. **Deploy Web Applications.** Copy the web applications files from your [dspace]/webapps directory to the subdirectory of your servlet container (e.g. tomcat):

```
cp -R [dspace]/webapps/* [tomcat]/webapps/
```

9. **Restart servlet container.** Now restart your Tomcat/Jetty/Resin server program and test out the upgrade.

## Upgrading from 1.6.x to 1.7.x

In the notes below [dspace] refers to the install directory for your existing DSpace installation, and [dspace-source] to the source directory for DSpace 1.7.x. Whenever you see these path references, be sure to replace them with the actual path names on your local system.

Before upgrading you need to check you are using the current recommended minimum versions of Java (1.6), Maven (2.0.8 or above) and ant (1.7 or above). For more details, see the current listing of [Prerequisite Software](#)

1. **Backup Your DSpace.** First, and foremost, make a complete backup of your system, including:
  - A snapshot of the database. *To have a "snapshot" of the PostgreSQL database, you need to shut it down during the backup. You should also have your regular PostgreSQL Backup output (using Postgres' [pg\\_dump](#) command).*
  - The asset store ([dspace]/assetstore by default)
  - Your configuration files and customizations to DSpace (including any customized scripts).
2. **Download DSpace 1.7.x** Retrieve the new DSpace 1.7.x source code either as a download from [DSpace.org](#) or check it out directly from the [SVN code repository](#). If you downloaded DSpace do not unpack it on top of your existing installation. Refer to [Installation Instructions, Step 3](#) for unpacking directives.
3. **Stop Tomcat.** Take down your servlet container. For Tomcat, use the \$CATALINA\_HOME/bin/shutdown.sh script. (Many Unix-based installations will have a startup/shutdown script in the /etc/init.d or /etc/rc.d directories).
4. **Apply any customizations.** If you have made any local customizations to your DSpace installation they will need to be migrated over to the new DSpace. These are normally housed in one of the following places:
  - JSPUI modifications: [dspace-source]/dspace/modules/jspui/src/main/webapp/
  - XMLUI modifications: [dspace-source]/dspace/modules/xmlui/src/main/webapp/
5. **Update Configuration Files.** Some parameters have changed and some are new. You can either attempt to make these changes in your current 1.6.x dspace.cfg file, or you can start with a new 1.7 dspace.cfg and re-modify it as needed. Configuration changes are noted below:
  - **\*CORRECTION\*** There was a missing hyphen "-" in the property key for mail character set:

```
# Set the default mail character set. This may be over ridden by providing a line
# inside the email template "charset: <encoding>", otherwise this default is used.
#mail.charset = UTF-8
```

- **\*CORRECTION\*** This was moved from the end of the solr configuration section to just under Logging Configurations:

```
# If enabled, the logging and the solr statistics system will look for
# an X-Forward header. If it finds it, it will use this for the user IP Address
# useProxies = true
```

- **\*CHANGE\*** The MediaFilter is now able to process Power Point Text Extractor

```

#Names of the enabled MediaFilter or FormatFilter plugins
filter.plugins = PDF Text Extractor, HTML Text Extractor, \
                PowerPoint Text Extractor, \
                Word Text Extractor, JPEG Thumbnail
# [To enable Branded Preview]: remove last line above, and uncomment 2 lines below
#                Word Text Extractor, JPEG Thumbnail, \
#                Branded Preview JPEG

#Assign 'human-understandable' names to each filter
plugin.named.org.dspace.app.mediafilter.FormatFilter = \
    org.dspace.app.mediafilter.PDFFilter = PDF Text Extractor, \
    org.dspace.app.mediafilter.HTMLFilter = HTML Text Extractor, \
    org.dspace.app.mediafilter.WordFilter = Word Text Extractor, \
    org.dspace.app.mediafilter.PowerPointFilter = PowerPoint Text Extractor, \
    org.dspace.app.mediafilter.JPEGFilter = JPEG Thumbnail, \
    org.dspace.app.mediafilter.BrandedPreviewJPEGFilter = Branded Preview JPEG

#Configure each filter's input format(s)
filter.org.dspace.app.mediafilter.PDFFilter.inputFormats = Adobe PDF
filter.org.dspace.app.mediafilter.HTMLFilter.inputFormats = HTML, Text
filter.org.dspace.app.mediafilter.WordFilter.inputFormats = Microsoft Word
filter.org.dspace.app.mediafilter.PowerPointFilter.inputFormats = Microsoft Powerpoint, Microsoft
Powerpoint XML
filter.org.dspace.app.mediafilter.JPEGFilter.inputFormats = BMP, GIF, JPEG, image/png
filter.org.dspace.app.mediafilter.BrandedPreviewJPEGFilter.inputFormats = BMP, GIF, JPEG, image/png

```

- **\*CHANGE\*** The Crosswalk Plugin Configuration has changed with additional lines. Edit your file accordingly:

```

# Crosswalk Plugin Configuration:
#   The purpose of Crosswalks is to translate an external metadata format to/from
#   the DSpace Internal Metadata format (DIM) or the DSpace Database.
#   Crosswalks are often used by one or more Packager plugins (see below).
plugin.named.org.dspace.content.crosswalk.IngestionCrosswalk = \
    org.dspace.content.crosswalk.AIPDIMCrosswalk = DIM, \
    org.dspace.content.crosswalk.AIPTechMDCrosswalk = AIP-TECHMD, \
    org.dspace.content.crosswalk.PREMISCrosswalk = PREMIS, \
    org.dspace.content.crosswalk.OREIngestionCrosswalk = ore, \
    org.dspace.content.crosswalk.NullIngestionCrosswalk = NIL, \
    org.dspace.content.crosswalk.OAIDCIngestionCrosswalk = dc, \
    org.dspace.content.crosswalk.DIMIngestionCrosswalk = dim, \
    org.dspace.content.crosswalk.METSRightsCrosswalk = METSRIGHTS, \
    org.dspace.content.crosswalk.RoleCrosswalk = DSPACE-ROLES

plugin.selfnamed.org.dspace.content.crosswalk.IngestionCrosswalk = \
    org.dspace.content.crosswalk.XSLTIngestionCrosswalk, \
    org.dspace.content.crosswalk.QDCCrosswalk

plugin.named.org.dspace.content.crosswalk.StreamIngestionCrosswalk = \
    org.dspace.content.crosswalk.NullStreamIngestionCrosswalk = NULLSTREAM, \
    org.dspace.content.crosswalk.CreativeCommonsRDFStreamIngestionCrosswalk = DSPACE_CCRDF, \
    org.dspace.content.crosswalk.LicenseStreamIngestionCrosswalk = DSPACE_DEPLICENSE

plugin.named.org.dspace.content.crosswalk.DisseminationCrosswalk = \
    org.dspace.content.crosswalk.AIPDIMCrosswalk = DIM, \
    org.dspace.content.crosswalk.AIPTechMDCrosswalk = AIP-TECHMD, \
    org.dspace.content.crosswalk.SimpleDCDisseminationCrosswalk = DC, \
    org.dspace.content.crosswalk.SimpleDCDisseminationCrosswalk = dc, \
    org.dspace.content.crosswalk.PREMISCrosswalk = PREMIS, \
    org.dspace.content.crosswalk.METSDisseminationCrosswalk = METS, \
    org.dspace.content.crosswalk.METSDisseminationCrosswalk = mets, \
    org.dspace.content.crosswalk.METSRightsCrosswalk = METSRIGHTS, \
    org.dspace.content.crosswalk.OREDisseminationCrosswalk = ore, \
    org.dspace.content.crosswalk.DIMDisseminationCrosswalk = dim, \
    org.dspace.content.crosswalk.RoleCrosswalk = DSPACE-ROLES

```

- **\*NEW\***

```

plugin.named.org.dspace.content.crosswalk.StreamDisseminationCrosswalk = \
  org.dspace.content.crosswalk.CreativeCommonsRDFStreamDisseminationCrosswalk = DSPACE_CCRDF, \
  org.dspace.content.crosswalk.CreativeCommonsTextStreamDisseminationCrosswalk = DSPACE_CCTEXT, \
  org.dspace.content.crosswalk.LicenseStreamDisseminationCrosswalk = DSPACE_DEPLICENSE

```

- **\*CHANGE\*** The Packager Plugin Configuration has changed considerably. Carefully revise your configuration file:

```

Packager Plugin Configuration:
#   Configures the ingest and dissemination packages that DSpace supports.
#   These Ingestor and Disseminator classes support a specific package file format
#   (e.g. METS) which DSpace understands how to import/export. Each Packager
#   plugin often will use one (or more) Crosswalk plugins to translate metadata (see above).
plugin.named.org.dspace.content.packager.PackageDisseminator = \
  org.dspace.content.packager.DSpaceAIPDisseminator = AIP, \
  org.dspace.content.packager.DSpaceMETSDisseminator = METS, \
  org.dspace.content.packager.RoleDisseminator = DSPACE-ROLES

plugin.named.org.dspace.content.packager.PackageIngestor = \
  org.dspace.content.packager.DSpaceAIPIngestor = AIP, \
  org.dspace.content.packager.PDFPackager = Adobe PDF, PDF, \
  org.dspace.content.packager.DSpaceMETSIngestor = METS, \
  org.dspace.content.packager.RoleIngestor = DSPACE-ROLES

```

- **\*CHANGE\*** The Mets Ingestor configuration has change and been updated. Carefully edit:

```

#### METS ingestor configuration:
# These settings configure how DSpace will ingest a METS-based package

# Configures the METS-specific package ingesters (defined above)
# 'default' settings are specified by 'default' key

# Default Option to save METS manifest in the item: (default is false)
mets.default.ingest.preserveManifest = false

# Default Option to make use of collection templates when using the METS ingestor (default is false)
mets.default.ingest.useCollectionTemplate = false

# Default crosswalk mappings
# Maps a METS 'mdtype' value to a DSpace crosswalk for processing.
# When the 'mdtype' value is same as the name of a crosswalk, that crosswalk
# will be called automatically (e.g. mdtype='PREMIS' calls the crosswalk named
# 'PREMIS', unless specified differently in below mapping)
# Format is 'mets.default.ingest.crosswalk.<mdType> = <DSpace-crosswalk-name>'
mets.default.ingest.crosswalk.DC = QDC
mets.default.ingest.crosswalk.DSpaceDepositLicense = DSPACE_DEPLICENSE
mets.default.ingest.crosswalk.Creative Commons = DSPACE_CCRDF
mets.default.ingest.crosswalk.CreativeCommonsRDF = DSPACE_CCRDF
mets.default.ingest.crosswalk.CreativeCommonsText = NULLSTREAM

# Locally cached copies of METS schema documents to save time on ingest. This
# will often speed up validation & ingest significantly. Before enabling
# these settings, you must manually cache all METS schemas in
# [dspace]/config/schemas/ (does not exist by default). Most schema documents
# can be found on the http://www.loc.gov/ website.
# Enable the below settings to pull these *.xsd files from your local cache.
# (Setting format: mets.xsd.<abbreviation> = <namespace> <local-file-name>)
#mets.xsd.mets = http://www.loc.gov/METS/ mets.xsd
#mets.xsd.xlink = http://www.w3.org/1999/xlink xlink.xsd
#mets.xsd.mods = http://www.loc.gov/mods/v3 mods.xsd
#mets.xsd.xml = http://www.w3.org/XML/1998/namespace xml.xsd
#mets.xsd.dc = http://purl.org/dc/elements/1.1/ dc.xsd
#mets.xsd.dcterms = http://purl.org/dc/terms/ dcterms.xsd
#mets.xsd.premis = http://www.loc.gov/standards/premis PREMIS.xsd
#mets.xsd.premisObject = http://www.loc.gov/standards/premis PREMIS-Object.xsd
#mets.xsd.premisEvent = http://www.loc.gov/standards/premis PREMIS-Event.xsd

```

```

#mets.xsd.premisAgent = http://www.loc.gov/standards/premis PREMIS-Agent.xsd
#mets.xsd.premisRights = http://www.loc.gov/standards/premis PREMIS-Rights.xsd

#### AIP Ingestor & Disseminator Configuration
# These settings configure how DSpace will ingest/export its own
# AIP (Archival Information Package) format for backups and restores
# (Please note, as the DSpace AIP format is also METS based, it will also
# use many of the 'METS ingestor configuration' settings directly above)

# AIP-specific ingestion crosswalk mappings
# (overrides 'mets.default.ingest.crosswalk' settings)
# Format is 'mets.dspaceAIP.ingest.crosswalk.<mdType> = <DSpace-crosswalk-name>'
mets.dspaceAIP.ingest.crosswalk.DSpaceDepositLicense = NULLSTREAM
mets.dspaceAIP.ingest.crosswalk.CreativeCommonsRDF = NULLSTREAM
mets.dspaceAIP.ingest.crosswalk.CreativeCommonsText = NULLSTREAM

# Create EPerson if necessary for Submitter when ingesting AIP (default=false)
# (by default, EPerson creation is already handled by 'DSpace-ROLES' Crosswalk)
#mets.dspaceAIP.ingest.createSubmitter = false

## AIP-specific Disseminator settings
# These settings allow you to customize which metadata formats are exported in AIPs

# Technical metadata in AIP (exported to METS <techMD> section)
# Format is <label-for-METS>:<DSpace-crosswalk-name> [, ...] (label is optional)
# If unspecified, defaults to "PREMIS"
aip.disseminate.techMD = PREMIS, DSPACE-ROLES

# Source metadata in AIP (exported to METS <sourceMD> section)
# Format is <label-for-METS>:<DSpace-crosswalk-name> [, ...] (label is optional)
# If unspecified, defaults to "AIP-TECHMD"
aip.disseminate.sourceMD = AIP-TECHMD

# Preservation metadata in AIP (exported to METS <digipovMD> section)
# Format is <label-for-METS>:<DSpace-crosswalk-name> [, ...] (label is optional)
# If unspecified, defaults to nothing in <digiprovMD> section
#aip.disseminate.digiprovMD =

# Rights metadata in AIP (exported to METS <rightsMD> section)
# Format is <label-for-METS>:<DSpace-crosswalk-name> [, ...] (label is optional)
# If unspecified, default to adding all Licenses (CC and Deposit licenses),
# as well as METSRights information
aip.disseminate.rightsMD = DSpaceDepositLicense:DSpace_DEPLICENSE, \
    CreativeCommonsRDF:DSpace_CCRDF, CreativeCommonsText:DSpace_CCTEXT, METSRIGHTS

# Descriptive metadata in AIP (exported to METS <dmdSec> section)
# Format is <label-for-METS>:<DSpace-crosswalk-name> [, ...] (label is optional)
# If unspecified, defaults to "MODS, DIM"
aip.disseminate.dmd = MODS, DIM

```

- **\*NEW\*** A new property has been added to control the discovery index for the Event System Configuration:

```

# consumer to maintain the discovery index
event.consumer.discovery.class = org.dspace.discovery.IndexEventConsumer
event.consumer.discovery.filters =
Community|Collection|Item|Bundle+Add|Create|Modify|Modify_Metadata|Delete|Remove

```

- **\*NEW\*** License bundle display is now configurable. You are able to either display or suppress.

```

# whether to display the contents of the licence bundle (often just the deposit
# licence in standard DSpace installation
webui.licence_bundle.show = false

```

- **\*CORRECTION\*** Thumbnail generation. The width and height of generated thumbnails had a missing equal sign.

```
# maximum width and height of generated thumbnails
thumbnail.maxwidth = 80
thumbnail.maxheight = 80
```

- **\*CORRECTION and ADDITION\*** Authority Control Settings have changed. Formerly called ChoiceAuthority, it is now referred to as DCInputAuthority.

```
## The DCInputAuthority plugin is automatically configured with every
## value-pairs element in input-forms.xml, namely:
##   common_identifiers, common_types, common_iso_languages
##
## The DSpaceControlledVocabulary plugin is automatically configured
## with every *.xml file in [dspace]/config/controlled-vocabularies,
## and creates a plugin instance for each, using base filename as the name.
## eg: nsi, srsc.
## Each DSpaceControlledVocabulary plugin comes with three configuration options:
# vocabulary.plugin._plugin_.hierarchy.store = <true|false>    # default: true
# vocabulary.plugin._plugin_.hierarchy.suggest = <true|false>  # default: true
# vocabulary.plugin._plugin_.delimiter = "<string>"            # default: "::"
##
## An example using "srsc" can be found later in this section

#plugin.selfnamed.org.dspace.content.authority.ChoiceAuthority = \
# org.dspace.content.authority.DCInputAuthority, \
# org.dspace.content.authority.DSpaceControlledVocabulary
```

- **\*NEW\*** Controls autocomplete for authority control

```
## demo: subject code autocomplete, using srsc as authority
## (DSpaceControlledVocabulary plugin must be enabled)
#choices.plugin.dc.subject = srsc
#choices.presentation.dc.subject = select
#vocabulary.plugin.srsc.hierarchy.store = true
#vocabulary.plugin.srsc.hierarchy.suggest = true
#vocabulary.plugin.srsc.delimiter = "::"
```

- **\*NEW\*** You are now able to order your bitstreams by sequence id or file name.

```
#### Ordering of bitstreams ####

## Specify the ordering that bitstreams are listed.
##
## Bitstream field to sort on. Values: sequence_id or name. Default: sequence_id
#webui.bitstream.order.field = "sequence_id"

## Direction of sorting order. Values: DESC or ASC. Default: ASC
#webui.bitstream.order.direction = ASC
```

- **\*NEW\*** DSpace now includes a metadata mapping feature that makes repository content discoverable by Google Scholar:

```
##### Google Scholar Metadata Configuration #####
google-metadata.config = ${dspace.dir}/config/crosswalks/google-metadata.properties
google-metadata.enable = true
```

- **\*NEW\*** XMLUI is now able to concatenate CSS, JS and JSON files:

```
# Enabling this property will concatenate CSS, JS and JSON files where possible.
# CSS files can be concatenated if multiple CSS files with the same media attribute
# are used in the same page. Links to the CSS files are automatically referring to the
# concatenated resulting CSS file.
# The theme sitemap should be updated to use the ConcatenationReader for all js, css and json
# files before enabling this property.
#xmlui.theme.enableConcatenation = false

# Enabling this property will minify CSS, JS and JSON files where possible.
# The theme sitemap should be updated to use the ConcatenationReader for all js, css and json
# files before enabling this property.
#xmlui.theme.enableMinification = false
```

- **\*NEW\*** XMLUI Mirage Theme. This is a new theme with it's own configuration:

```
### Setings for Item lists in Mirage theme ###
# What should the emphasis be in the display of item lists?
# Possible values : 'file', 'metadata'. If your repository is
# used mainly for scientific papers 'metadata' is probably the
# best way. If you have a lot of images and other files 'file'
# will be the best starting point
# (metadata is the default value if this option is not specified)
#xmlui.theme.mirage.item-list.emphasis = file
```

- **\*NEW\*** OAI Response default change.

```
# DSpace by default uses 100 records as the limit for the oai responses.
# This can be altered by enabling the oai.response.max-records parameter
# and setting the desired amount of results.
oai.response.max-records = 100
```

- **\*CHANGE\*** EPDCX property key has been renamed.

```
# Define the metadata type EPDCX (EPrints DC XML)
# to be handled by the SWORD crosswalk configuration
#
mets.default.ingest.crosswalk.EPDCX = SWORD
```

- **\*NEW\*** New SOLR Statistic Property keys:

```
# Timeout for the resolver in the dns lookup
# Time in milliseconds, defaults to 200 for backward compatibility
# Your systems default is usually set in /etc/resolv.conf and varies
# between 2 to 5 seconds, to high a value might result in solr exhausting
# your connection pool
solr.resolver.timeout = 200

----

# Enable/disable logging of spiders in solr statistics.
# If false, and IP matches an address in solr.spiderips.urls, event is not logged.
# If true, event will be logged with the 'isBot' field set to true
# (see solr.statistics.query.filter.* for query filter options)
# Default value is true.
#solr.statistics.logBots = true
```

**6. Build DSpace.** Run the following commands to compile DSpace.:

```
cd [dspace-source]/dspace/  
mvn -U clean package
```

You will find the result in `[dspace-source]/dspace/target/dspace-[version]-build.dir`. Inside this directory is the compiled binary distribution of DSpace. Before rebuilding DSpace, the above command will clean out any previously compiled code ('clean') and ensure that your local DSpace JAR files are updated from the remote maven code repository.

7. **Update DSpace.** Update the DSpace installed directory with the new code and libraries. Issue the following commands:

```
cd [dspace-source]/dspace/target/dspace-[version]-build.dir  
ant -Dconfig=[dspace]/config/dspace.cfg update
```

8. **Update the Database.** You will need to run the 1.6.x to 1.7.x database upgrade script.

- For PostgreSQL:

```
psql -U [dspace-user] -f [dspace-source]/dspace/etc/postgres/database_schema_16-17.sql [database  
name]
```

(Your database name is by default 'dspace'). Example:

```
psql -U dspace -f [dspace-source]/dspace/etc/postgres/database_schema_16-17.sql dspace
```

- For Oracle: Execute the upgrade script, e.g. with sqlplus, recording the output:
  - a. Start SQL\*Plus with `sqlplus [connect args]`
  - b. Record the output: `SQL> spool 'upgrade.lst'`
  - c. Run the upgrade script `SQL> @[dspace-source]/dspace/etc/oracle/database_schema_16-17.sql`
  - d. Turn off recording of output: `SQL> spool off`

9. **Generate Browse and Search Indexes.** It's always good policy to rebuild your search and browse indexes when upgrading to a new release. To do this, run the following command from your DSpace install directory (as the 'dspace' user):

```
[dspace]/bin/dspace index-init
```

10. **Deploy Web Applications.** If your servlet container (e.g. Tomcat) is not configured to look for new web applications in your `[dspace]/webapps` directory, then you will need to copy the web applications files into the appropriate subdirectory of your servlet container. For example:

```
cp -R [dspace]/webapps/* [tomcat]/webapps/
```

11. **Restart servlet container.** Now restart your Tomcat/Jetty/Resin server program and test out the upgrade.
12. **Add a new crontab entry,** or add to your system's scheduler, the following, run as the DSpace user, to enable routine maintenance of your SOLR indexes. If you do not run this command daily, it is likely your production instances of DSpace will exhaust the available memory in your servlet container

```
[dspace]/bin/dspace stats-util -o
```

## Upgrading from 1.6 to 1.6.x

In the notes below `[dspace]` refers to the install directory for your existing DSpace installation, and `[dspace-source]` to the source directory for DSpace 1.6.1. Whenever you see these path references, be sure to replace them with the actual path names on your local system.

1. **Backup Your DSpace.** First, and foremost, make a complete backup of your system, including:
  - A snapshot of the database. *To have a "snapshot" of the PostgreSQL database, you need to shut it down during the backup. You should also have your regular PostgreSQL Backup output (using `pg_dump` commands).*
  - The asset store (`[dspace]/assetstore` by default)
  - Your configuration files and customizations to DSpace (including any customized scripts).
2. **Download DSpace 1.6.2** Retrieve the new DSpace 1.6.2 source code either as a download from [DSpace.org](https://dspace.org) or check it out directly from the [SVN code repository](#). If you downloaded DSpace do not unpack it on top of your existing installation. Refer to [Installation Instructions, Step 3](#) for unpacking directives.
3. **Stop Tomcat.** Take down your servlet container. For Tomcat, use the `$CATALINA/shutdown.sh` script. (Many installations will have a startup /shutdown script in the `/etc/init.d` or `/etc/rc.d` directories.
4. **Apply any customizations.** If you have made any local customizations to your DSpace installation they will need to be migrated over to the new DSpace. These are housed in one of the following places:
  - JSPUI modifications: `[dspace-source]/dspace/modules/jspui/src/main/webapp/`
  - XMLUI modifications: `[dspace-source]/dspace/modules/xmlui/src/main/webapp/`
5. **Update Configuration Files.** There are no additions to this release. So you do not have to update the configuration files.
6. **Build DSpace.** Run the following commands to compile DSpace.:



```
cd [dspace-source]/dspace/  
mvn -U clean package
```

You will find the result in `[dspace-source]/dspace/target/dspace-[version]-build.dir`. Inside this directory is the compiled binary distribution of DSpace. Before rebuilding DSpace, the above command will clean out any previously compiled code ('clean') and ensure that your local DSpace JAR files are updated from the remote maven repository.

7. **Update DSpace.** Update the DSpace installed directory with the new code and libraries. Issue the following commands:

```
cd [dspace-source]/dspace/target/dspace-[version]-build.dir  
ant -Dconfig=[dspace]/config/dspace.cfg update
```

8. **Run Registry Format Update for CC License.** Creative Commons licenses have been assigned the wrong mime-type in past versions of DSpace. Even if you are not currently using CC Licenses, you should update your Bitstream Format Registry to include a new entry with the proper mime-type. To update your registry, run the following command: `dspace/bin/dspace registry-loader -bitstream [dspace]/etc/updates/15-16/new-bitstream-formats.xml`
9. **Update the Database.** If you are using Creative Commons Licenses in your DSpace submission process, you will need to run the 1.5.x to 1.6.x database upgrade script again. In 1.6.0 the improper mime-type was being assigned to all CC Licenses. This has now been resolved, and rerunning the upgrade script will now assign the proper mime-type to all existing CC Licenses in your DSpace installation. NOTE: You will receive messages that most of the script additions already exist. This is normal, and nothing to be worried about.
  - For PostgreSQL: `psql -U [dspace-user] -f [dspace-source]/dspace/etc/postgres/database_schema_15-16.sql [database name]` (Your database name is by default 'dspace'). Example:

```
psql -U dspace -f  
/dspace-1.6-1-src-release/dspace/etc/postgres/database_schema_15-16.sql dspace
```

*(The line break above is cosmetic. Please place your command in one line.)*

- For Oracle: Execute the upgrade script, e.g. with sqlplus, recording the output:
  - a. Start SQL\*Plus with `sqlplus [connect args]`
  - b. Record the output: `SQL> spool 'upgrade.lst'`
  - c. Run the upgrade script `SQL> @[dspace-source]/dspace/etc/oracle/database_schema_15-16.sql`
  - d. Turn off recording of output: `SQL> spool off`
  - e. Please note: The final few statements WILL FAIL. That is because you have run some queries and use the results to construct the statements to remove the constraints, manually, Oracle doesn't have any easy way to automate this (unless you know PL/SQL). So, look for the comment line beginning:

```
"--You need to remove the already in place constraints"  
and follow the instructions in the actual SQL file.  
Refer to the contents of the spool file "upgrade.lst" for  
the output of the queries you'll need.
```

10. **Generate Browse and Search Indexes.** Though there are not any database changes in the 1.6 to 1.6.1 release, it makes good policy to rebuild your search and browse indexes when upgrading to a new release. To do this, run the following command from your DSpace install directory (as the dspace user): `[dspace]/bin/dspace index-init`
11. **Deploy Web Applications.** Copy the web applications files from your `[dspace]/webapps` directory to the subdirectory of your servlet container (e.g. tomcat): `cp -R [dspace]/webapps/* [tomcat]/webapps/`
12. **Restart servlet.** Now restart your Tomcat/Jetty/Resin server program and test out the upgrade.

## Upgrading from 1.5.x to 1.6.x

In the notes below `[dspace]` refers to the install directory for your existing DSpace installation, and `[dspace-source]` to the source directory for DSpace 1.6. Whenever you see these path references, be sure to replace them with the actual path names on your local system.

1. **Backup Your DSpace.** First, and foremost, make a complete backup of your system, including:
  - A snapshot of the database. \_To have a "snapshot" of the PostgreSQL database, you need to shut it down during the backup. You should also have your regular PostgreSQL Backup output (using `pg_dump` commands). \_
  - The asset store (`[dspace]/assetstore` by default)
  - Your configuration files and customizations to DSpace (including any customized scripts).
2. **Download DSpace 1.6.x** Retrieve the new DSpace 1.6.x source code either as a download from [DSpace.org](https://dspace.org) or check it out directly from the [SVN code repository](#). If you downloaded DSpace do not unpack it on top of your existing installation. Refer to [Installation Instructions, Step 3](#) for unpacking directives.
3. **Stop Tomcat.** Take down your servlet container. For Tomcat, use the `$CATALINA/shutdown.sh` script. (Many installations will have a startup /shutdown script in the `/etc/init.d` or `/etc/rc.d` directories.
4. **Apply any customizations.** If you have made any local customizations to your DSpace installation they will need to be migrated over to the new DSpace. These are housed in one of the following places:
  - JSPUI modifications: `[dspace-source]/dspace/modules/jspui/src/main/webapp/`
  - XMLUI modifications: `[dspace-source]/dspace/modules/xmlui/src/main/webapp/`
5. **Update Configuration Files.** Some of the parameters have change and some are new. Changes will be noted below:
  - **\*\*CHANGE\*\*** The base url and oai urls property keys are set differently

```
# DSpace host name - should match base URL. Do not include port number
dspace.hostname = localhost

# DSpace base host URL. Include port number etc.
dspace.baseUrl = http://localhost:8080

# DSpace base URL. Include port number etc., but NOT trailing slash
# Change to xmlui if you wish to use the xmlui as the default, or remove
# "/jspui" and set webapp of your choice as the "ROOT" webapp in
# the servlet engine.
dspace.url = ${dspace.baseUrl}/xmlui

# The base URL of the OAI webapp (do not include /request).
dspace.oai.url = ${dspace.baseUrl}/oai
```

- **\*\*NEW\*\*** New email options (Add these at the end of the "Email Settings" sub-section):

```
# A comma separated list of hostnames that are allowed to refer browsers to
# email forms. Default behavior is to accept referrals only from
# dspace.hostname
#mail.allowed.referrers = localhost

# Pass extra settings to the Java mail library. Comma separated, equals sign
# between the key and the value.
#mail.extraproperties = mail.smtp.socketFactory.port=465, \
#                       mail.smtp.socketFactory.class=javax.net.ssl.SSLSocketFactory, \
#                       mail.smtp.socketFactory.fallback=false

# An option is added to disable the mailserver. By default, this property is
# set to false. By setting mail.server.disabled = true, DSpace will not send
# out emails. It will instead log the subject of the email which should have
# been sent. This is especially useful for development and test environments
# where production data is used when testing functionality.
#mail.server.disabled = false
```

- **\*\*NEW\*\*** New Authorization levels and parameters. See the [Configuration](#) documentation, "Delegation Administration" section for further information.

```

##### Authorization system configuration - Delegate ADMIN #####

# COMMUNITY ADMIN configuration
# subcommunities and collections
#core.authorization.community-admin.create-subelement = true
#core.authorization.community-admin.delete-subelement = true
# his community
#core.authorization.community-admin.policies = true
#core.authorization.community-admin.admin-group = true
# collections in his community
#core.authorization.community-admin.collection.policies = true
#core.authorization.community-admin.collection.template-item = true
#core.authorization.community-admin.collection.submitters = true
#core.authorization.community-admin.collection.workflows = true
#core.authorization.community-admin.collection.admin-group = true
# item owned by collections in his community
#core.authorization.community-admin.item.delete = true
#core.authorization.community-admin.item.withdraw = true
#core.authorization.community-admin.item.reinstantiate = true
#core.authorization.community-admin.item.policies = true
# also bundle...
#core.authorization.community-admin.item.create-bitstream = true
#core.authorization.community-admin.item.delete-bitstream = true
#core.authorization.community-admin.item-admin.cc-license = true

# COLLECTION ADMIN
#core.authorization.collection-admin.policies = true
#core.authorization.collection-admin.template-item = true
#core.authorization.collection-admin.submitters = true
#core.authorization.collection-admin.workflows = true
#core.authorization.collection-admin.admin-group = true
# item owned by his collection
#core.authorization.collection-admin.item.delete = true
#core.authorization.collection-admin.item.withdraw = true
#core.authorization.collection-admin.item.reinstantiate = true
#core.authorization.collection-admin.item.policies = true
# also bundle...
#core.authorization.collection-admin.item.create-bitstream = true
#core.authorization.collection-admin.item.delete-bitstream = true
#core.authorization.collection-admin.item-admin.cc-license = true

# ITEM ADMIN
#core.authorization.item-admin.policies = true
# also bundle...
#core.authorization.item-admin.create-bitstream = true
#core.authorization.item-admin.delete-bitstream = true
#core.authorization.item-admin.cc-license = true

```

- **\*\*CHANGE\*\*** METS ingester has been revised. (Modify In "Crosswalk and Packager Plugin Settings")

```
# Option to make use of collection templates when using the METS ingester (default is false)
mets.submission.useCollectionTemplate = false

# Crosswalk Plugins:
plugin.named.org.dspace.content.crosswalk.IngestionCrosswalk = \
  org.dspace.content.crosswalk.PREMISCrosswalk = PREMIS \
  org.dspace.content.crosswalk.OREIngestionCrosswalk = ore \
  org.dspace.content.crosswalk.NullIngestionCrosswalk = NIL \
  org.dspace.content.crosswalk.QDCCrosswalk = qdc \
  org.dspace.content.crosswalk.OAIDCIngestionCrosswalk = dc \
  org.dspace.content.crosswalk.DIMIngestionCrosswalk = dim

plugin.selfnamed.org.dspace.content.crosswalk.IngestionCrosswalk = \
  org.dspace.content.crosswalk.XSLTIngestionCrosswalk

plugin.named.org.dspace.content.crosswalk.DisseminationCrosswalk = \
  org.dspace.content.crosswalk.SimpleDCDisseminationCrosswalk = DC \
  org.dspace.content.crosswalk.SimpleDCDisseminationCrosswalk = dc \
  org.dspace.content.crosswalk.PREMISCrosswalk = PREMIS \
  org.dspace.content.crosswalk.METSDisseminationCrosswalk = METS \
  org.dspace.content.crosswalk.METSDisseminationCrosswalk = mets \
  org.dspace.content.crosswalk.OREDisseminationCrosswalk = ore \
  org.dspace.content.crosswalk.QDCCrosswalk = qdc \
  org.dspace.content.crosswalk.DIMDisseminationCrosswalk = dim
```

- **\*\*CHANGE\*\*** Event Settings have had the following revision with the addition of 'harvester' (modify in "Event System Configuration"):

```
#### Event System Configuration ####

# default synchronous dispatcher (same behavior as traditional DSpace)
event.dispatcher.default.class = org.dspace.event.BasicDispatcher
event.dispatcher.default.consumers = search, browse, eperson, harvester
```

also:

```
# consumer to clean up harvesting data
event.consumer.harvester.class = org.dspace.harvest.HarvestConsumer
event.consumer.harvester.filters = Item+Delete
```

- **\*\*NEW\*\*** New option for the Embargo of Thesis and Dissertations.

```
#### Embargo Settings ####
# DC metadata field to hold the user-supplied embargo terms
embargo.field.terms = SCHEMA.ELEMENT.QUALIFIER

# DC metadata field to hold computed "lift date" of embargo
embargo.field.lift = SCHEMA.ELEMENT.QUALIFIER

# string in terms field to indicate indefinite embargo
embargo.terms.open = forever

# implementation of embargo setter plugin--replace with local implementation if
# applicable
plugin.single.org.dspace.embargo.EmbargoSetter = \
  org.dspace.embargo.DefaultEmbargoSetter

# implementation of embargo lifter plugin--replace with local implementation if
# applicable
plugin.single.org.dspace.embargo.EmbargoLifter = \
  org.dspace.embargo.DefaultEmbargoLifter
```

- **\*\*NEW\*\*** New option for using the Batch Editing capabilities. See [Batch Metadata Editing Configuration](#) and also [System Administration : Batch Metadata Editing](#)

```

### Bulk metadata editor settings ###
# The delimiter used to separate values within a single field (defaults to a double pipe ||)
# bulkedit.valueseparator = ||

# The delimiter used to separate fields (defaults to a comma for CSV)
# bulkedit.fieldseparator = ,

# A hard limit of the number of items allowed to be edited in one go in the UI
# (does not apply to the command line version)
# bulkedit.gui-item-limit = 20

# Metadata elements to exclude when exporting via the user interfaces, or when
# using the command line version and not using the -a (all) option.
# bulkedit.ignore-on-export = dc.date.accessioned, dc.date.available, \
#                               dc.date.updated, dc.description.provenance

```

- **\*\*NEW\*\*** Ability to hide metadata fields is now available. (Look for "*JSPUI & XMLUI Configurations*" Section)

```

##### Hide Item Metadata Fields #####
# Fields named here are hidden in the following places UNLESS the
# logged-in user is an Administrator:
# 1. XMLUI metadata XML view, and Item splash pages (long and short views).
# 2. JSPUI Item splash pages
# 3. OAI-PMH server, "oai_dc" format.
# (NOTE: Other formats are _not_ affected.)
# To designate a field as hidden, add a property here in the form:
#   metadata.hide.SCHEMA.ELEMENT.QUALIFIER = true
#
# This default configuration hides the dc.description.provenance field,
# since that usually contains email addresses which ought to be kept
# private and is mainly of interest to administrators:
metadata.hide.dc.description.provenance = true

```

- **\*\*NEW\*\*** Choice Control and Authority Control options are available (Look for "*JSPUI & XMLUI Configurations*" Section):

```

## example of authority-controlled browse category--see authority control config
#webui.browse.index.5 = lcAuthor:metadataAuthority:dc.contributor.author:authority

```

And also:

```

##### Authority Control Settings #####

#plugin.named.org.dspace.content.authority.ChoiceAuthority = \
# org.dspace.content.authority.SampleAuthority = Sample, \
# org.dspace.content.authority.LCNameAuthority = LCNameAuthority, \
# org.dspace.content.authority.SHERPARoMEOPublisher = SRPublisher, \
# org.dspace.content.authority.SHERPARoMEOJournalTitle = SRJournalTitle

## This ChoiceAuthority plugin is automatically configured with every
## value-pairs element in input-forms.xml, namely:
##   common_identifiers, common_types, common_iso_languages
#plugin.selfnamed.org.dspace.content.authority.ChoiceAuthority = \
# org.dspace.content.authority.DCInputAuthority

## configure LC Names plugin
#lcname.url = http://alcme.oclc.org/srw/search/lcnaf

## configure SHERPA/RoMEO authority plugin
#sherpa.romeo.url = http://www.sherpa.ac.uk/romeo/api24.php

##
## This sets the default lowest confidence level at which a metadata value is included
## in an authority-controlled browse (and search) index. It is a symbolic
## keyword, one of the following values (listed in descending order):
##   accepted
##   uncertain
##   ambiguous
##   notfound
##   failed
##   rejected
##   novalue
##   unset
## See manual or org.dspace.content.authority.Choices source for descriptions.
authority.minconfidence = ambiguous

## demo: use LC plugin for author
#choices.plugin.dc.contributor.author = LCNameAuthority
#choices.presentation.dc.contributor.author = lookup
#authority.controlled.dc.contributor.author = true
##
## This sets the lowest confidence level at which a metadata value is included
## in an authority-controlled browse (and search) index. It is a symbolic
## keyword from the same set as for the default "authority.minconfidence"
#authority.minconfidence.dc.contributor.author = accepted

## Demo: publisher name lookup through SHERPA/RoMEO:
#choices.plugin.dc.publisher = SRPublisher
#choices.presentation.dc.publisher = suggest

## demo: journal title lookup, with ISSN as authority
#choices.plugin.dc.title.alternative = SRJournalTitle
#choices.presentation.dc.title.alternative = suggest
#authority.controlled.dc.title.alternative = true

## demo: use choice authority (without authority-control) to restrict dc.type on EditItemMetadata
## page
# choices.plugin.dc.type = common_types
# choices.presentation.dc.type = select

## demo: same idea for dc.language.iso
# choices.plugin.dc.language.iso = common_iso_languages
# choices.presentation.dc.language.iso = select

# Change number of choices shown in the select in Choices lookup popup
#xmlui.lookup.select.size = 12

```

- **\*\*REPLACE\*\*** RSS Feeds now support Atom 1.0. *Replace its previous configuration with the one below:*

```

#### Syndication Feed (RSS) Settings #####

# enable syndication feeds - links display on community and collection home pages
# (This setting is not used by XMLUI, as you enable feeds in your theme)
webui.feed.enable = false
# number of DSpace items per feed (the most recent submissions)
webui.feed.items = 4
# maximum number of feeds in memory cache
# value of 0 will disable caching
webui.feed.cache.size = 100
# number of hours to keep cached feeds before checking currency
# value of 0 will force a check with each request
webui.feed.cache.age = 48
# which syndication formats to offer
# use one or more (comma-separated) values from list:
# rss_0.90, rss_0.91, rss_0.92, rss_0.93, rss_0.94, rss_1.0, rss_2.0
webui.feed.formats = rss_1.0,rss_2.0,atom_1.0
# URLs returned by the feed will point at the global handle server (e.g. http://hdl.handle.net
/123456789/1)
# Set to true to use local server URLs (i.e. http://myserver.myorg/handle/123456789/1)
webui.feed.localresolve = false

# Customize each single-value field displayed in the
# feed information for each item. Each of
# the below fields takes a *single* metadata field
#
# The form is <schema prefix>.<element>[.<qualifier>|. *]
webui.feed.item.title = dc.title
webui.feed.item.date = dc.date.issued

# Customize the metadata fields to show in the feed for each item's description.
# Elements will be displayed in the order that they are specified here.
#
# The form is <schema prefix>.<element>[.<qualifier>|. *][(date)], ...
#
# Similar to the item display UI, the name of the field for display
# in the feed will be drawn from the current UI dictionary,
# using the key:
# "metadata.<field>"
#
# e.g.      "metadata.dc.title"
#           "metadata.dc.contributor.author"
#           "metadata.dc.date.issued"
webui.feed.item.description = dc.title, dc.contributor.author, \
                             dc.contributor.editor, dc.description.abstract, \
                             dc.description
# name of field to use for authors (Atom only) - repeatable
webui.feed.item.author = dc.contributor.author

# Customize the extra namespaced DC elements added to the item (RSS) or entry
# (Atom) element. These let you include individual metadata values in a
# structured format for easy extraction by the recipient, instead of (or in
# addition to) appending these values to the Description field.
## dc:creator value(s)
#webui.feed.item.dc.creator = dc.contributor.author
## dc:date value (may be contradicted by webui.feed.item.date)
#webui.feed.item.dc.date = dc.date.issued
## dc:description (e.g. for a distinct field that is ONLY the abstract)
#webui.feed.item.dc.description = dc.description.abstract

# Customize the image icon included with the site-wide feeds:
# Must be an absolute URL, e.g.
## webui.feed.logo.url = ${dspace.url}/themes/mysite/images/mysite-logo.png

```

- **\*\*NEW\*\*** Opensearch Feature is new to DSpace

```
#### OpenSearch Settings ####
# NB: for result data formatting, OpenSearch uses Syndication Feed Settings
# so even if Syndication Feeds are not enabled, they must be configured
# enable open search
websvc.opensearch.enable = false
# context for html request URLs - change only for non-standard servlet mapping
websvc.opensearch.uicontext = simple-search
# context for RSS/Atom request URLs - change only for non-standard servlet mapping
websvc.opensearch.svccontext = open-search/
# present autodiscovery link in every page head
websvc.opensearch.autolink = true
# number of hours to retain results before recalculating
websvc.opensearch.validity = 48
# short name used in browsers for search service
# should be 16 or fewer characters
websvc.opensearch.shortname = DSpace
# longer (up to 48 characters) name
websvc.opensearch.longname = ${dspace.name}
# brief service description
websvc.opensearch.description = ${dspace.name} DSpace repository
# location of favicon for service, if any must be 16X16 pixels
websvc.opensearch.faviconurl = http://www.dspace.org/images/favicon.ico
# sample query - should return results
websvc.opensearch.samplequery = photosynthesis
# tags used to describe search service
websvc.opensearch.tags = IR DSpace
# result formats offered - use 1 or more comma-separated from: html,atom,rss
# NB: html is required for autodiscovery in browsers to function,
# and must be the first in the list if present
websvc.opensearch.formats = html,atom,rss
```

- **\*\*NEW\*** Exposure of METS metadata can be now hidden. (See "*OAI-PMH SPECIFIC CONFIGURATIONS*" in the *dspace.cfg* file)

```
# When exposing METS/MODS via OAI-PMH all metadata that can be mapped to MODS
# is exported. This includes description.provenance which can contain personal
# email addresses and other information not intended for public consumption. To
# hide this information set the following property to true
oai.mets.hide-provenance = true
```

- **\*\*NEW\*** SWORD has added the following to accept MIME/types. (See "*SWORD Specific Configurations*" Section)

```
# A comma separated list of MIME types that SWORD will accept
sword.accepts = application/zip
```

- **\*\*NEW\*** New OAI Harvesting Configuration settings are now available. (See "*OAI Harvesting Configurations*"

```
#-----#
#-----OAI HARVESTING CONFIGURATIONS-----#
#-----#
# These configs are only used by the OAI-ORE related functions #
#-----#

### Harvester settings

# Crosswalk settings; the {name} value must correspond to a declared ingestion crosswalk
# harvester.oai.metadataformats.{name} = {namespace},{optional display name}
harvester.oai.metadataformats.dc = http://www.openarchives.org/OAI/2.0/oai_dc/, Simple Dublin Core
harvester.oai.metadataformats.qdc = http://purl.org/dc/terms/, Qualified Dublin Core
harvester.oai.metadataformats.dim = http://www.dspace.org/xmlns/dspace/dim, DSpace Intermediate
Metadata

# This field works in much the same way as harvester.oai.metadataformats.PluginName
# The {name} must correspond to a declared ingestion crosswalk, while the
# {namespace} must be supported by the target OAI-PMH provider when harvesting content.
# harvester.oai.oreSerializationFormat.{name} = {namespace}
```



```

# Determines whether the harvester scheduling process should be started
# automatically when the DSpace webapp is deployed.
# default: false
harvester.autoStart=false

# Amount of time subtracted from the from argument of the PMH request to account
# for the time taken to negotiate a connection. Measured in seconds. Default value is 120.
#harvester.timePadding = 120

# How frequently the harvest scheduler checks the remote provider for updates,
# measured in minutes. The default value is 12 hours (or 720 minutes)
#harvester.harvestFrequency = 720

# The heartbeat is the frequency at which the harvest scheduler queries the local
# database to determine if any collections are due for a harvest cycle (based on
# the harvestFrequency) value. The scheduler is optimized to then sleep until the
# next collection is actually ready to be harvested. The minHeartbeat and
# maxHeartbeat are the lower and upper bounds on this timeframe. Measured in seconds.
# Default minHeartbeat is 30. Default maxHeartbeat is 3600.
#harvester.minHeartbeat = 30
#harvester.maxHeartbeat = 3600

# How many harvest process threads the scheduler can spool up at once. Default value is 3.
#harvester.maxThreads = 3

# How much time passes before a harvest thread is terminated. The termination process
# waits for the current item to complete ingest and saves progress made up to that point.
# Measured in hours. Default value is 24.
#harvester.threadTimeout = 24

# When harvesting an item that contains an unknown schema or field within a schema what
# should the harvester do? Either add a new registry item for the field or schema, ignore
# the specific field or schema (importing everything else about the item), or fail with
# an error. The default value if undefined is: fail.
# Possible values: 'fail', 'add', or 'ignore'
harvester.unknownField = add
harvester.unknownSchema = fail

# The webapp responsible for minting the URIs for ORE Resource Maps.
# If using oai, the dspace.oai.uri config value must be set.
# The URIs generated for ORE ReMs follow the following convention for both cases.
# format: [baseURI]/metadata/handle/[theHandle]/ore.xml
# Default value is oai
#ore.authoritative.source = oai

# A harvest process will attempt to scan the metadata of the incoming items
# (dc.identifier.uri field, to be exact) to see if it looks like a handle.
# If so, it matches the pattern against the values of this parameter.
# If there is a match the new item is assigned the handle from the metadata value
# instead of minting a new one. Default value: hdl.handle.net
#harvester.acceptedHandleServer = hdl.handle.net, handle.myu.edu

# Pattern to reject as an invalid handle prefix (known test string, for example)
# when attempting to find the handle of harvested items. If there is a match with
# this config parameter, a new handle will be minted instead. Default value: 123456789.
#harvester.rejectedHandlePrefix = 123456789, myTestHandle

```

- **\*\*NEW\*\*** SOLR Statistics Configurations. For a little more detailed information regarding the configuration, please refer to [DSpace SOLR Statistics Configuration](#) ; or, for installation procedures, refer to [Advanced Installation: Dspace Statistics](#).

```
#-----#
#-----SOLR STATISTICS CONFIGURATIONS-----#
#-----#
# These configs are only used by the SOLR interface/webapp to #
# track usage statistics. #
#-----#

##### Usage Logging #####
solr.log.server = ${dspace.baseUrl}/solr/statistics
solr.spidersfile = ${dspace.dir}/config/spiders.txt
solr.dbfile = ${dspace.dir}/config/GeoLiteCity.dat
useProxies = true

statistics.item.authorization.admin=true
```

6. **Build DSpace.** Run the following commands to compile DSpace.:

```
cd [dspace-source]/dspace/
mvn -U clean package
```

You will find the result in `[dspace-source]/dspace/target/dspace-[version]-build.dir`. Inside this directory is the compiled binary distribution of DSpace. Before rebuilding DSpace, the above command will clean out any previously compiled code ('clean') and ensure that your local DSpace JAR files are updated from the remote maven repository.

7. **Update the database.** The database schema needs to be updated to accommodate changes to the database. SQL files contain the relevant updates are provided. Please note that if you have made any local customizations to the database schema, you should consult these updates and make sure they will work for you.
- For PostgreSQL: `psql -U [dspace-user] -f [dspace-source]/dspace/etc/postgres/database_schema_15-16.sql [database name]` (Your database name is by default 'dspace'). Example: `psql -U dspace -f /dspace-1.6-1-src-release/dspace/etc/postgres/database_schema_15-16.sql dspace`
  - For Oracle: Execute the upgrade script, e.g. with sqlplus, recording the output:
    - Start SQL\*Plus with `sqlplus [connect args]`
    - Record the output: `SQL> spool 'upgrade.lst'`
    - Run the upgrade script `SQL> @[dspace-source]/dspace/etc/oracle/database_schema_15-16.sql`
    - Turn off recording of output: `SQL> spool off`
    - Please note: The final few statements WILL FAIL. That is because you have run some queries and use the results to construct the statements to remove the constraints, manually, Oracle doesn't have any easy way to automate this (unless you know PL/SQL). So, look for the comment line beginning:

```
"--You need to remove the already in place constraints"
and follow the instructions in the actual SQL file.
Refer to the contents of the spool file "upgrade.lst" for
the output of the queries you'll need.
```

8. **Update DSpace.** Update the DSpace installed directory with the new code and libraries. Issue the following commands:

```
cd [dspace-source]/dspace/target/dspace-[version]-build.dir
ant -Dconfig=[dspace]/config/dspace.cfg update
```

9. **Update Registry for the CC License.** If you use the CC License, an incorrect mime-type type is being assigned. You will need to run the following step: `_dspace/bin/dspace registry-loader -bitstream [dspace]/etc/upgrades/15-16/new-bitstream-formats.xml _`
10. **Generate Browse and Search Indexes.** It makes good policy to rebuild your search and browse indexes when upgrading to a new release. Almost every release has database changes and indexes can be affected by this. In the DSpace 1.6 release there is Authority Control features and those will need the indexes to be regenerated. To do this, run the following command from your DSpace install directory (as the dspace user): `[dspace]/bin/dspace index-init`
11. **Deploy Web Applications.** Copy the web applications files from your `[dspace]/webapps` directory to the subdirectory of your servlet container (e.g. tomcat): `cp -R [dspace]/webapps/* [tomcat]/webapps/`
12. **Restart servlet.** Now restart your Tomcat/Jetty/Resin server program and test out the upgrade.
13. **Rolling Log Appender Upgrade.** You will want to upgrade your logs to the new format to use the SOLR Statistics now included with DSpace. While the commands for this are found in Chapter 8, here is the steps needed to be performed.

```
[dspace]/bin/dspace stats-log-converter -i input file name -o output file name -m (if you have more than
one dspace.log file)
[dspace]/bin/dspace stats-log-importer -i input file name (probably the output name from above) -m
```

The user is highly recommend to see the [System Administration : DSpace Log Converter](#) documentation.

## Upgrading From 1.5 or 1.5.1 to 1.5.2

The changes in DSpace 1.5.2 do not include any database schema upgrades, and the upgrade should be straightforward.

In the notes below `[dspace]` refers to the install directory for your existing DSpace installation, and `[dspace-source]` to the source directory for DSpace 1.5. Whenever you see these path references, be sure to replace them with the actual path names on your local system.

1. **Backup your DSpace** First and foremost, make a complete backup of your system, including:
  - A snapshot of the database
  - The asset store (`[dspace]/assetstore` by default)
  - Your configuration files and customizations to DSpace
  - Your statistics scripts (`[dspace]/bin/stat*`) which contain customizable dates
2. **Download DSpace 1.5.2** Get the new DSpace 1.5.2 source code either as a download from [DSpace.org](http://DSpace.org) or check it out directly from the [SVN code repository](#). If you downloaded DSpace do not unpack it on top of your existing installation.
3. **Build DSpace** Run the following commands to compile DSpace.

```
cd [dspace-source]/dspace/  
mvn package
```

You will find the result in `[dspace-source]/dspace/target/dspace-1.5.2-build.dir/`; inside this directory is the compiled binary distribution of DSpace.

4. **Stop Tomcat** Take down your servlet container, for Tomcat use the `bin/shutdown.sh` script.
5. **Apply any customizations** If you have made any local customizations to your DSpace installation they will need to be migrated over to the new DSpace. Commonly these modifications are made to "JSP" pages located inside the `[dspace 1.4.2]/jsp/local` directory. These should be moved `[dspace-source]/dspace/modules/jspui/src/main/webapp/` in the new build structure. See Customizing the JSP Pages for more information.
6. **Update DSpace** Update the DSpace installed directory with new code and libraries. Inside the `[dspace-source]/dspace/target/dspace-1.5-build.dir/` directory run:

```
cd [dspace-source]/dspace/target/dspace-1.5-build.dir/  
ant -Dconfig=[dspace]/config/dspace.cfg update
```

7. **Update configuration files** This ant target preserves existing files in `[dspace]/config/` and will copy any new configuration files in place. If an existing file prevents copying the new file in place, the new file will have the suffix `_.new`, for example `[dspace]/local/dspace.cfg.new`. Note: there is also a configuration option `-Doverwrite=true` which will instead copy the conflicting target files to `*.old` suffixes and overwrite target file then with the new file (essentially the opposite) this is beneficial for developers and those who use the `[dspace-source]/dspace/config` to maintain their changes.

```
cd [dspace-source]/dspace/target/dspace-1.5-build.dir/  
ant -Dconfig=[dspace]/config/dspace.cfg update_configs
```

You must then verify that you've merged and differenced in the `[dspace]/config/*.new` files into your configuration. Some of the new parameters you should look out for in `dspace.cfg` include:

- New option to restrict the expose of private items. The following needs to be added to `dspace.cfg`:

```
#### Restricted item visibility settings ####  
# By default RSS feeds, OAI-PMH and subscription emails will include ALL items  
# regardless of permissions set on them.  
#  
# If you wish to only expose items through these channels where the ANONYMOUS  
# user is granted READ permission, then set the following options to false  
#harvest.includerestricted.rss = true  
#harvest.includerestricted.oai = true  
#harvest.includerestricted.subscription = true
```

- Special groups for LDAP and password authentication.

```
##### Password users group #####

# If required, a group name can be given here, and all users who log in
# using the DSpace password system will automatically become members of
# this group. This is useful if you want a group made up of all internal
# authenticated users.
#password.login.specialgroup = group-name

##### LDAP users group #####

# If required, a group name can be given here, and all users who log in
# to LDAP will automatically become members of this group. This is useful
# if you want a group made up of all internal authenticated users.
#ldap.login.specialgroup = group-name
```

- new option for case insensitivity in browse tables.

```
# By default, the display of metadata in the browse indexes is case sensitive
# So, you will get separate entries for the terms
#
#   Olive oil
#   olive oil
#
# However, clicking through from either of these will result in the same set of items
# (ie. any item that contains either representation in the correct field).
#
# Uncommenting the option below will make the metadata items case-insensitive. This will
# result in a single entry in the example above. However the value displayed may be either 'Olive
# oil'
# or 'olive oil' - depending on what representation was present in the first item indexed.
#
# If you care about the display of the metadata in the browse index - well, you'll have to go and
# fix the metadata in your items.
#
# webui.browse.metadata.case-insensitive = true
```

- New usage event handler for collecting statistics:

```
### Usage event settings ###
# The usage event handler to call. The default is the "passive" handler, which ignores events.
# plugin.single.org.dspace.app.statistics.AbstractUsageEvent = \
#   org.dspace.app.statistics.PassiveUsageEvent
```

- The location where sitemaps are stored is now configurable.

```
##### Sitemap settings #####
# the directory where the generated sitemaps are stored
sitemap.dir = ${dspace.dir}/sitemaps
```

- MARC 21 ordering should now be used as default. Unless you have it set already, or you have it set to a different value, the following should be set:

```
plugin.named.org.dspace.sort.OrderFormatDelegate = org.dspace.sort.OrderFormatTitleMarc21=title
```

- Hierarchical LDAP support.

```

##### Hierarchical LDAP Settings #####
# If your users are spread out across a hierarchical tree on your
# LDAP server, you will need to use the following stackable authentication
# class:
#   plugin.sequence.org.dspace.authenticate.AuthenticationMethod = \
#       org.dspace.authenticate.LDAPHierarchicalAuthentication
#
# You can optionally specify the search scope. If anonymous access is not
# enabled on your LDAP server, you will need to specify the full DN and
# password of a user that is allowed to bind in order to search for the
# users.

# This is the search scope value for the LDAP search during
# autoregistering. This will depend on your LDAP server setup.
# This value must be one of the following integers corresponding
# to the following values:
# object scope : 0
# one level scope : 1
# subtree scope : 2
#ldap.search_scope = 2

# The full DN and password of a user allowed to connect to the LDAP server
# and search for the DN of the user trying to log in. If these are not specified,
# the initial bind will be performed anonymously.
#ldap.search.user = cn=admin,ou=people,o=myu.edu
#ldap.search.password = password

# If your LDAP server does not hold an email address for a user, you can use
# the following field to specify your email domain. This value is appended
# to the netid in order to make an email address. E.g. a netid of 'user' and
# ldap.netid_email_domain as '@example.com' would set the email of the user
# to be 'user@example.com'
#ldap.netid_email_domain = @example.com

```

- Shibboleth authentication support.

```

#### Shibboleth Authentication Configuration Settings ####
# Check https://mams.melcoe.mq.edu.au/zope/mams/pubs/Installation/dspace15/view
# for installation detail.
#
#         org.dspace.authenticate.ShibAuthentication
#
# DSpace requires email as user's credential. There are 2 ways of providing
# email to DSpace:
# 1) by explicitly specifying to the user which attribute (header)
#     carries the email address.
# 2) by turning on the user-email-using-tomcat=true which means
#     the software will try to acquire the user's email from Tomcat
# The first option takes PRECEDENCE when specified. Both options can
# be enabled to allow fallback.

# this option below specifies that the email comes from the mentioned header.
# The value is CASE-Sensitive.
authentication.shib.email-header = MAIL

# optional. Specify the header that carries user's first name
# this is going to be used for creation of new-user
authentication.shib.firstname-header = SHIB-EP-GIVENNAME

# optional. Specify the header that carries user's last name
# this is used for creation of new user
authentication.shib.lastname-header = SHIB-EP-SURNAME

# this option below forces the software to acquire the email from Tomcat.
authentication.shib.email-use-tomcat-remote-user = true

# should we allow new users to be registered automatically
# if the IdP provides sufficient info (and user not exists in DSpace)
authentication.shib.autoregister = true

# this header here specifies which attribute that is responsible
# for providing user's roles to DSpace. When not specified, it is
# defaulted to 'Shib-EP-UnscopedAffiliation'. The value is specified
# in AAP.xml (Shib 1.3.x) or attribute-filter.xml (Shib 2.x).
# The value is CASE-Sensitive. The values provided in this
# header are separated by semi-colon or comma.
# authentication.shib.role-header = Shib-EP-UnscopedAffiliation

# when user is fully authN on IdP but would not like to release
# his/her roles to DSpace (for privacy reason?), what should be
# the default roles be given to such users?
# The values are separated by semi-colon or comma
# authentication.shib.default-roles = Staff, Walk-ins

# The following mappings specify role mapping between IdP and Dspace.
# the left side of the entry is IdP's role (prefixed with
# "authentication.shib.role.") which will be mapped to
# the right entry from DSpace. DSpace's group as indicated on the
# right entry has to EXIST in DSpace, otherwise user will be identified
# as 'anonymous'. Multiple values on the right entry should be separated
# by comma. The values are CASE-Sensitive. Heuristic one-to-one mapping
# will be done when the IdP groups entry are not listed below (i.e.
# if "X" group in IdP is not specified here, then it will be mapped
# to "X" group in DSpace if it exists, otherwise it will be mapped
# to simply 'anonymous')
#
# Given sufficient demand, future release could support regex for the mapping
# special characters need to be escaped by \
authentication.shib.role.Senior\ Researcher = Researcher, Staff
authentication.shib.role.Librarian = Administrator

```

- DOI and handle identifiers can now be rendered in the JSPUI.

```

# When using "resolver" in webui.itemdisplay to render identifiers as resolvable
# links, the base URL is taken from <code>webui.resolver.<n>.baseurl</code>
# where <code>webui.resolver.<n>.baseurl</code> matches the urn specified in the metadata value.
# The value is appended to the "baseurl" as is, so the baseurl need to end with slash almost in
any case.
# If no urn is specified in the value it will be displayed as simple text.
#
#webui.resolver.1.urn = doi
#webui.resolver.1.baseurl = http://dx.doi.org/
#webui.resolver.2.urn = hdl
#webui.resolver.2.baseurl = http://hdl.handle.net/
#
# For the doi and hdl urn defaults values are provided, respectively http://dx.doi.org and
# http://hdl.handle.net are used.
#
# If a metadata value with style: "doi", "handle" or "resolver" matches a URL
# already, it is simply rendered as a link with no other manipulation.

```

In configuration sections such as `webui.itemdisplay.default`, values can be changed from (e.g.) `metadata.dc.identifier.doi` to `metadata.doi.dc.identifier.doi`

- The whole of the SWORD configuration has changed. The SWORD section must be removed and replaced with

```

#-----#
#-----SWORD SPECIFIC CONFIGURATIONS-----#
#-----#
# These configs are only used by the SWORD interface #
#-----#

# tell the SWORD METS implementation which package ingester to use
# to install deposited content. This should refer to one of the
# classes configured for:
#
# plugin.named.org.dspace.content.packager.PackageIngester
#
# The value of sword.mets-ingester.package-ingester tells the
# system which named plugin for this interface should be used
# to ingest SWORD METS packages
#
# The default is METS
#
# sword.mets-ingester.package-ingester = METS

# Define the metadata type EPDCX (EPrints DC XML)
# to be handled by the SWORD crosswalk configuration
#
mets.submission.crosswalk.EPDCX = SWORD

# define the stylesheet which will be used by the self-named
# XSLTIngestionCrosswalk class when asked to load the SWORD
# configuration (as specified above). This will use the
# specified stylesheet to crosswalk the incoming SWAP metadata
# to the DIM format for ingestion
#
crosswalk.submission.SWORD.stylesheet = crosswalks/sword-swap-ingest.xsl

# The base URL of the SWORD deposit. This is the URL from
# which DSpace will construct the deposit location urls for
# collections.
#
# The default is {dspace.url}/sword/deposit
#
# In the event that you are not deploying DSpace as the ROOT
# application in the servlet container, this will generate
# incorrect URLs, and you should override the functionality
# by specifying in full as below:
#
# sword.deposit.url = http://www.myu.ac.uk/sword/deposit

```

```

# The base URL of the SWORD service document. This is the
# URL from which DSpace will construct the service document
# location urls for the site, and for individual collections
#
# The default is {dspace.url}/sword/servicedocument
#
# In the event that you are not deploying DSpace as the ROOT
# application in the servlet container, this will generate
# incorrect URLs, and you should override the functionality
# by specifying in full as below:
#
# sword.servicedocument.url = http://www.myu.ac.uk/sword/servicedocument

# The base URL of the SWORD media links. This is the URL
# which DSpace will use to construct the media link urls
# for items which are deposited via sword
#
# The default is {dspace.url}/sword/media-link
#
# In the event that you are not deploying DSpace as the ROOT
# application in the servlet container, this will generate
# incorrect URLs, and you should override the functionality
# by specifying in full as below:
#
# sword.media-link.url = http://www.myu.ac.uk/sword/media-link

# The URL which identifies the sword software which provides
# the sword interface. This is the URL which DSpace will use
# to fill out the atom:generator element of its atom documents.
#
# The default is:
#
# http://www.dspace.org/ns/sword/1.3.1
#
# If you have modified your sword software, you should change
# this URI to identify your own version. If you are using the
# standard dspace-sword module you will not, in general, need
# to change this setting
#
# sword.generator.url = http://www.dspace.org/ns/sword/1.3.1

# The metadata field in which to store the updated date for
# items deposited via SWORD.
#
sword.updated.field = dc.date.updated

# The metadata field in which to store the value of the slug
# header if it is supplied
#
sword.slug.field = dc.identifier.slug

# the accept packaging properties, along with their associated
# quality values where appropriate.
#
# Global settings; these will be used on all DSpace collections
#
sword.accept-packaging.METSDSpaceSIP.identifier =
http://purl.org/net/sword-types/METSDSpaceSIP
sword.accept-packaging.METSDSpaceSIP.q = 1.0

# Collection Specific settings: these will be used on the collections
# with the given handles
#
# sword.accept-packaging.[handle].METSDSpaceSIP.identifier =
http://purl.org/net/sword-types/METSDSpaceSIP
# sword.accept-packaging.[handle].METSDSpaceSIP.q = 1.0

# Should the server offer up items in collections as sword deposit
# targets. This will be effected by placing a URI in the collection
# description which will list all the allowed items for the depositing
# user in that collection on request

```



```

#
# NOTE: this will require an implementation of deposit onto items, which
# will not be forthcoming for a short while
#
sword.expose-items = false

# Should the server offer as the default the list of all Communities
# to a Service Document request. If false, the server will offer
# the list of all collections, which is the default and recommended
# behavior at this stage.
#
# NOTE: a service document for Communities will not offer any viable
# deposit targets, and the client will need to request the list of
# Collections in the target before deposit can continue
#
sword.expose-communities = false

# The maximum upload size of a package through the sword interface,
# in bytes
#
# This will be the combined size of all the files, the metadata and
# any manifest data. It is NOT the same as the maximum size set
# for an individual file upload through the user interface. If not
# set, or set to 0, the sword service will default to no limit.
#
sword.max-upload-size = 0

# Should DSpace store a copy of the original sword deposit package?
#
# NOTE: this will cause the deposit process to run slightly slower,
# and will accelerate the rate at which the repository consumes disk
# space. BUT, it will also mean that the deposited packages are
# recoverable in their original form. It is strongly recommended,
# therefore, to leave this option turned on
#
# When set to "true", this requires that the configuration option
# "upload.temp.dir" above is set to a valid location
#
sword.keep-original-package = true

# The bundle name that SWORD should store incoming packages under if
# sword.keep-original-package is set to true. The default is "SWORD"
# if not value is set
#
# sword.bundle.name = SWORD

# Should the server identify the sword version in deposit response?
#
# It is recommended to leave this enabled.
#
sword.identify-version = true

# Should we support mediated deposit via sword? Enabled, this will
# allow users to deposit content packages on behalf of other users.
#
# See the SWORD specification for a detailed explanation of deposit
# On-Behalf-Of another user
#
sword.on-behalf-of.enable = true

# Configure the plugins to process incoming packages. The form of this
# configuration is as per the Plugin Manager's Named Plugin documentation:
#
# plugin.named.[interface] = [implementation] = [package format identifier] \
#
# Package ingesters should implement the SWORDIngestor interface, and
# will be loaded when a package of the format specified above in:
#
# sword.accept-packaging.[package format].identifier = [package format identifier]
#
# is received.

```

```
#
# In the event that this is a simple file deposit, with no package
# format, then the class named by "SimpleFileIngester" will be loaded
# and executed where appropriate. This case will only occur when a single
# file is being deposited into an existing DSpace Item
#
plugin.named.org.dspace.sword.SWORDIngester = \
  org.dspace.sword.SWORDMETSIngester =
http://purl.org/net/sword-types/METSDSpaceSIP \
  org.dspace.sword.SimpleFileIngester = SimpleFileIngester
```

8. **Restart Tomcat** Restart your servlet container, for Tomcat use the *bin/startup.sh* script.

## Upgrading From 1.4.2 to 1.5

The changes in DSpace 1.5 are significant and wide spread involving database schema upgrades, code restructuring, completely new user and programmatic interfaces, and new build system.

In the notes below *[dspace]* refers to the install directory for your existing DSpace installation, and *[dspace-source]* to the source directory for DSpace 1.5. Whenever you see these path references, be sure to replace them with the actual path names on your local system.

1. **Backup your DSpace** First and foremost, make a complete backup of your system, including:
  - A snapshot of the database
  - The asset store (*[dspace]/assetstore* by default)
  - Your configuration files and customizations to DSpace
  - Your statistics scripts (*[dspace]/bin/stat\**) which contain customizable dates
2. **Download DSpace 1.5.x** Get the new DSpace 1.5 source code either as a download from [SourceForge](#) or check it out directly from the [SVN code repository](#). If you downloaded DSpace do not unpack it on top of your existing installation.
3. **Build DSpace** The build process has radically changed for DSpace 1.5. With this new release the build system has moved to a maven-based system enabling the various projects (JSPUI, XMLUI, OAI, and Core API) into separate projects. See the [Installation](#) section for more information on building DSpace using the new maven-based build system. Run the following commands to compile DSpace.

```
cd [dspace-source]/dspace/;
mvn package
```

You will find the result in *[dspace-source]/dspace/target/dspace-1.5-build.dir/*; inside this directory is the compiled binary distribution of DSpace.

4. **Stop Tomcat** Take down your servlet container, for Tomcat use the *bin/shutdown.sh* script.
5. **Update *dspace.cfg*** Several new parameters need to be added to your *[dspace]/config/dspace.cfg*. While it is advisable to start with a fresh *DSpace 1.5\_dspace.cfg* configuration file, here are the minimum set of parameters that need to be added to an old *DSpace 1.4.2* configuration.

```
##### Stackable Authentication Methods #####
#
# Stack of authentication methods
# (See org.dspace.authenticate.AuthenticationManager)
# Note when upgrading you should remove the parameter:
# plugin.sequence.org.dspace.eperson.AuthenticationMethod
plugin.sequence.org.dspace.authenticate.AuthenticationMethod = \
  org.dspace.authenticate.PasswordAuthentication

##### JSPUI item style plugin #####
#
# Specify which strategy use for select the style for an item
plugin.single.org.dspace.app.webui.util.StyleSelection = \
  org.dspace.app.webui.util.CollectionStyleSelection

##### Browse Configuration #####
#
# The following configuration will mimic the previous
# behavior exhibited by DSpace 1.4.2. For alternative
# configurations see the manual.

# Browse indexes
webui.browse.index.1 = dateissued:item:dateissued
webui.browse.index.2 = author:metadata:dc.contributor.*:text
webui.browse.index.3 = title:item:title
```

```

webui.browse.index.4 = subject:metadata:dc.subject.*:text

# Sorting options
webui.itemlist.sort-option.1 = title:dc.title:title
webui.itemlist.sort-option.2 = dateissued:dc.date.issued:date
webui.itemlist.sort-option.3 =
    dateaccessioned:dc.date.accessioned:date

# Recent submissions
recent.submissions.count = 5

# Itemmapper browse index
itemmap.author.index = author

# Recent submission processor plugins
plugin.sequence.org.dspace.plugin.CommunityHomeProcessor = \
    org.dspace.app.webui.components.RecentCommunitySubmissions
plugin.sequence.org.dspace.plugin.CollectionHomeProcessor = \
    org.dspace.app.webui.components.RecentCollectionSubmissions

#### Content Inline Disposition Threshold ####
#
# Set the max size of a bitstream that can be served inline
# Use -1 to force all bitstream to be served inline
# webui.content_disposition_threshold = -1
webui.content_disposition_threshold = 8388608

#### Event System Configuration ####
#
# default synchronous dispatcher (same behavior as traditional
    DSpace)
event.dispatcher.default.class = org.dspace.event.BasicDispatcher
event.dispatcher.default.consumers = search, browse, eperson

# consumer to maintain the search index
event.consumer.search.class = org.dspace.search.SearchConsumer
event.consumer.search.filters =
    Item|Collection|Community|Bundle+Create|Modify|Modify_Metadata|Delete:
Bundle+Add|Remove

# consumer to maintain the browse index
event.consumer.browse.class = org.dspace.browse.BrowseConsumer
event.consumer.browse.filters =
    Item+Create|Modify|Modify_Metadata:Collection+Add|Remove

# consumer related to EPerson changes
event.consumer.eperson.class = org.dspace.eperson.EPersonConsumer
event.consumer.eperson.filters = EPerson+Create

```

6. **Add 'xmlui.xconf' Manakin configuration** The new Manakin user interface available with DSpace 1.5 requires an extra configuration file that you will need to manually copy it over to your configuration directory.

```

cp [dspace-source]/dspace/config/xmlui.xconf
   [dspace]/config/xmlui.xconf

```

7. **Add 'item-submission.xml' and 'item-submission.dtd' configurable submission configuration** The new configurable submission system that enables an administrator to re-arrange, or add/remove item submission steps requires this configuration file. You need to manually copy it over to your configuration directory.

```

cp [dspace-source]/dspace/config/item-submission.xml
   [dspace]/config/item-submission.xml

cp [dspace-source]/dspace/config/item-submission.dtd
   [dspace]/config/item-submission.dtd

```

8. **Add new 'input-forms.xml' and 'input-forms.dtd' configurable submission configuration** The input-forms.xml now has an included dtd reference to support validation. You'll need to merge in your changes to both file/and or copy them into place.

```
cp [dspace-source]/dspace/config/input-forms.xml
   [dspace]/config/input-forms.xml

cp [dspace-source]/dspace/config/input-forms.dtd
   [dspace]/config/inputforms.dtd
```

9. **Add 'sword-swap-ingest.xml' and 'xhtml-head-item.properties' crosswalk files** New crosswalk files are required to support SWORD and the inclusion of metadata into the head of items.

```
cp [dspace-source]/dspace/config/crosswalks/sword-swap-ingest.xml
   [dspace]/config/crosswalks/sword-swap-ingest.xml

cp
   [dspace-source]/dspace/config/crosswalks/xhtml-head-item.properties
   [dspace]/config/crosswalks/xhtml-head-item.properties
```

10. **Add 'registration\_notify' email files** A new configuration option (*registration.notify = you@your-email.com*) can be set to send a notification email whenever a new user registers to use your DSpace. The email template for this email needs to be copied.

```
cp [dspace-source]/dspace/config/emails/registration_notify
   [dspace]/config/emails/registration_notify
```

11. **Update the database** The database schema needs updating. SQL files contain the relevant updates are provided, note if you have made any local customizations to the database schema you should consult these updates and make sure they will work for you.
- For PostgreSQL *psql -U [dspace-user] -f [dspace-source]/dspace/etc/database\_schema\_14-15.sql [database-name]*
  - For Oracle *[dspace-source]/dspace/etc/oracle/database\_schema\_142-15.sql* contains the commands necessary to upgrade your database schema on oracle.
12. **Apply any customizations** If you have made any local customizations to your DSpace installation they will need to be migrated over to the new DSpace. Commonly these modifications are made to "JSP" pages located inside the *[dspace 1.4.2]/jsp/local* directory. These should be moved *[dspace-source]/dspace/modules/jspui/src/main/webapp/* in the new build structure. See Customizing the JSP Pages for more information.
13. **Update DSpace** Update the DSpace installed directory with new code and libraries. Inside the *[dspace-source]/dspace/target/dspace-1.5-build.dir/* directory run:

```
cd [dspace-source]/dspace/target/dspace-1.5-build.dir/;
ant -Dconfig=[dspace]/config/dspace.cfg update
```

14. **Update the Metadata Registry** New Metadata Registry updates are required to support SWORD.

```
cp [dspace-source]/dspace/config/registries/sword-metadata.xml
   [dspace]/config/registries/sword-metadata.xml;

[dspace]/bin/dsrun org.dspace.administer.MetadataImporter -f
   [dspace]/config/registries/sword-metadata.xml
```

15. **Rebuild browse and search indexes** One of the major new features of DSpace 1.5 is the browse system which necessitates that the indexes be recreated. To do this run the following command from your DSpace installed directory:

```
[dspace]/bin/index-init
```

16. **Update statistics scripts** The statistics scripts have been rewritten for DSpace 1.5. Prior to 1.5 they were written in Perl, but have been rewritten in Java to avoid having to install Perl. First, make a note of the dates you have specified in your statistics scripts for the statistics to run from. You will find these in *[dspace]/bin/stat-initial*, as *\$start\_year* and *\$start\_month*. Note down these values. Copy the new stats scripts:

```
cp [dspace-source]/dspace/bin/stat* [dspace]/bin/
```

Then edit your statistics configuration file with the start details. Add the following to *[dspace]/conf/dstat.cfg* *# the year and month to start creating reports from* *# - year as four digits (e.g. 2005)* *# - month as a number (e.g. January is 1, December is 12)* *start.year = 2005* *start.month = 1* Replace '2005' and '1' as with the values you noted down. *dstat.cfg* also used to contain the hostname and service name as displayed at the top of the statistics. These values are now taken from *dspace.cfg* so you can remove *host.name* and *host.url* from *dstat.cfg* if you wish. The values now used are *dspace.hostname* and *dspace.name* from *dspace.cfg*

17. **Deploy web applications** Copy the web applications files from your *[dspace]/webapps* directory to the subdirectory of your servlet container (e.g. Tomcat):

```
cp [dspace]/webapps/* [tomcat]/webapps/
```

18. **Restart Tomcat** Restart your servlet container, for Tomcat use the *bin/startup.sh* script.

## Upgrading From 1.4.1 to 1.4.2

See Upgrading From 1.4 to 1.4.x; the same instructions apply.

## Upgrading From 1.4 to 1.4.x

The changes in 1.4.x releases are only code and configuration changes so the update is simply a matter of rebuilding the wars and slight changes to your config file.

In the notes below *[dspace]* refers to the install directory for your existing DSpace installation, and *[dspace-1.4.x-source]* to the source directory for DSpace 1.4.x. Whenever you see these path references, be sure to replace them with the actual path names on your local system.

1. Get the new DSpace 1.4.x source code from [the DSpace page on SourceForge](#) and unpack it somewhere. Do not unpack it on top of your existing installation!!
2. Copy the PostgreSQL driver JAR to the source tree. For example:

```
cd [dspace]/lib
cp postgresql.jar [dspace-1.4.x-source]/lib
```

3. **Note:** Licensing conditions for the *handle.jar* file have changed. As a result, the latest version of the *handle.jar* file is not included in this distribution. It is recommended you read the [new license conditions](#) and decide whether you wish to update your installation's *handle.jar*. If you decide to update, you should replace the existing *handle.jar* in *[dspace-1.4.x-source]/lib* with the new version.
4. Take down Tomcat (or whichever servlet container you're using).
5. A new configuration item *webui.html.max-depth-guess* has been added to avoid infinite URL spaces. Add the following to the *dspace.cfg* file:

```
##### Multi-file HTML document/site settings #####
#
# When serving up composite HTML items, how deep can the request be
#   for us to
#   serve up a file with the same name?
#
# e.g. if we receive a request for "foo/bar/index.html"
# and we have a bitstream called just "index.html"
# we will serve up that bitstream for the request if
#   webui.html.max-depth-guess
# is 2 or greater. If webui.html.max-depth-guess is 1 or less, we
#   would not
# serve that bitstream, as the depth of the file is greater.
#
# If webui.html.max-depth-guess is zero, the request filename and
#   path must
# always exactly match the bitstream name. Default value is 3.
#
webui.html.max-depth-guess = 3
```

If *webui.html.max-depth-guess* is not present in *dspace.cfg* the default value is used. If archiving entire web sites or deeply nested HTML documents it is advisable to change the default to a higher value more suitable for these types of materials.

6. Your 'localized' JSPs (those in `jsp/local`) now need to be maintained in the `source` directory. If you have locally modified JSPs in your `[dspace]/jsp/local` directory, you will need to merge the changes in the new 1.4.x versions into your locally modified ones. You can use the `diff` command to compare your JSPs against the 1.4.x versions to do this. You can also check against the [DSpace CVS](#).
7. In `[dspace-1.4.x-source]` run:

```
ant -Dconfig= [dspace]/config/dspace.cfg update
```

8. Copy the `.war` Web application files in `[dspace-1.4.x-source]/build` to the `webapps` sub-directory of your servlet container (e.g. Tomcat). e.g.:

```
cp [dspace-1.4.x-source]/build/*.war
   [tomcat]/webapps
```

If you're using Tomcat, you need to delete the directories corresponding to the old `.war` files. For example, if `dspace.war` is installed in `[tomcat]/webapps/dspace.war`, you should delete the `[tomcat]/webapps/dspace` directory. Otherwise, Tomcat will continue to use the old code in that directory.

9. Restart Tomcat.

## Upgrading From 1.3.2 to 1.4.x

1. First and foremost, **make a complete backup** of your system, including:
  - A snapshot of the database
  - The asset store (`[dspace]/assetstore` by default)
  - Your configuration files and localized JSPs
2. Download the [latest DSpace 1.4.x source bundle](#) and unpack it in a suitable location (*not* over your existing DSpace installation or source tree!)
3. Copy the PostgreSQL driver JAR to the source tree. For example:

```
cd [dspace]/lib
cp postgresql.jar [dspace-1.4.x-source]/lib
```

4. **Note:** Licensing conditions for the `handle.jar` file have changed. As a result, the latest version of the `handle.jar` file is not included in this distribution. It is recommended you read the [new license conditions](#) and decide whether you wish to update your installation's `handle.jar`. If you decide to update, you should replace the existing `handle.jar` in `[dspace-1.4.x-source]/lib` with the new version.
5. Take down Tomcat (or whichever servlet container you're using).
6. Your DSpace configuration will need some updating:
  - In `dspace.cfg`, paste in the following lines for the new stackable authentication feature, the new method for managing Media Filters, and the Checksum Checker.

```
##### Stackable Authentication Methods #####
# Stack of authentication methods
# (See org.dspace.eperson.AuthenticationManager)
plugin.sequence.org.dspace.eperson.AuthenticationMethod = \
    org.dspace.eperson.PasswordAuthentication

##### Example of configuring X.509 authentication
##### (to use it, add org.dspace.eperson.X509Authentication to stack)

## method 1, using keystore
#authentication.x509.keystore.path = /var/local/tomcat/conf/keystore
#authentication.x509.keystore.password = changeit

## method 2, using CA certificate
#authentication.x509.ca.cert = ${dspace.dir}/config/mitClientCA.der

## Create e-persons for unknown names in valid certificates?
#authentication.x509.autoregister = true

##### Media Filter plugins (through PluginManager) #####

plugin.sequence.org.dspace.app.mediafilter.MediaFilter = \
    org.dspace.app.mediafilter.PDFFilter,
    org.dspace.app.mediafilter.HTMLFilter, \
    org.dspace.app.mediafilter.WordFilter,
    org.dspace.app.mediafilter.JPEGFilter
# to enable branded preview: remove last line above, and uncomment 2
```

```

        lines below
#   org.dspace.app.mediafilter.WordFilter,
        org.dspace.app.mediafilter.JPEGFilter, \
#   org.dspace.app.mediafilter.BrandedPreviewJPEGFilter

filter.org.dspace.app.mediafilter.PDFFilter.inputFormats = Adobe PDF
filter.org.dspace.app.mediafilter.HTMLFilter.inputFormats = HTML,
    Text
filter.org.dspace.app.mediafilter.WordFilter.inputFormats = Microsoft
    Word
filter.org.dspace.app.mediafilter.JPEGFilter.inputFormats = GIF,
    JPEG, image/png
filter.org.dspace.app.mediafilter.BrandedPreviewJPEGFilter.inputFormat
s = GIF, JPEG, image/png

#### Settings for Item Preview ####
webui.preview.enabled = false
# max dimensions of the preview image
webui.preview.maxwidth = 600
webui.preview.maxheight = 600
# the brand text
webui.preview.brand = My Institution Name
# an abbreviated form of the above text, this will be used
# when the preview image cannot fit the normal text
webui.preview.brand.abbrev = MyOrg
# the height of the brand
webui.preview.brand.height = 20
# font settings for the brand text
webui.preview.brand.font = SansSerif
webui.preview.brand.fontpoint = 12
#webui.preview.dc = rights

#### Checksum Checker Settings ####
# Default dispatcher in case none specified
plugin.single.org.dspace.checker.BitstreamDispatcher=org.dspace.checke
r.SimpleDispatcher
# Standard interface implementations. You shouldn't need to tinker
    with these.
plugin.single.org.dspace.checker.ReporterDAO=org.dspace.checker.Report
erDAOImpl

# check history retention
checker.retention.default=10y
checker.retention.CHECKSUM_MATCH=8w

```

- If you have customized advanced search fields (*search.index.n* fields, note that you now need to include the schema in the values. Dublin Core is specified as *dc*. So for example, if in 1.3.2 you had:

```
search.index.1 = title:title.alternative
```

That needs to be changed to:

```
search.index.1 = title:dc.title.alternative
```

- If you use LDAP or X509 authentication, you'll need to add *org.dspace.eperson.LDAPAuthentication* or *org.dspace.eperson.X509Authentication* respectively. See also configuring custom authentication code.
  - If you have custom Media Filters, note that these are now configured through *dspace.cfg* (instead of *mediafilter.cfg* which is obsolete.)
  - Also, take a look through the default *dspace.cfg* file supplied with DSpace 1.4.x, as this contains configuration options for various new features you might like to use. In general, these new features default to 'off' and you'll need to add configuration properties as described in the default 1.4.x *dspace.cfg* to activate them.
7. Your 'localized' JSPs (those in *jsp/local*) now need to be maintained in the *source* directory. If you have locally modified JSPs in your *[dspace]/jsp/local* directory, you will need to merge the changes in the new 1.4.x versions into your locally modified ones. You can use the *diff* command to compare your JSPs against the 1.4.x versions to do this. You can also check against the [DSpace CVS](#).
  8. In *[dspace-1.4.x-source]* run:

```
ant -Dconfig= [dspace]/config/dspace.cfg update
```

9. The database schema needs updating. SQL files containing the relevant file are provided. If you've modified the schema locally, you may need to check over this and make alterations.
  - **For PostgreSQL:** *[dspace-1.4.x-source]/etc/database\_schema\_13-14.sql* contains the SQL commands to achieve this for PostgreSQL. To apply the changes, go to the source directory, and run: *psql -f etc/database\_schema\_13-14.sql [DSpace database name] -h localhost*
  - **For Oracle:** *[dspace-1.4.x-source]/etc/oracle/database\_schema\_13-14.sql* should be run on the DSpace database to update the schema.
10. Rebuild the search indexes: *[dspace]/bin/index-all*
11. Copy the *.war* Web application files in *[dspace-1.4-source]/build* to the *webapps* sub-directory of your servlet container (e.g. Tomcat). e.g.:

```
cp [dspace-1.4-source]/build/*.war
   [tomcat]/webapps
```

If you're using Tomcat, you need to delete the directories corresponding to the old *.war* files. For example, if *dspace.war* is installed in *[tomcat]/webapps/dspace.war*, you should delete the *[tomcat]/webapps/dspace* directory. Otherwise, Tomcat will continue to use the old code in that directory.

12. Restart Tomcat.

## Upgrading From 1.3.1 to 1.3.2

The changes in 1.3.2 are only code changes so the update is simply a matter of rebuilding the wars.

In the notes below *[dspace]* refers to the install directory for your existing DSpace installation, and *[dspace-1.3.2-source]* to the source directory for DSpace 1.3.2. Whenever you see these path references, be sure to replace them with the actual path names on your local system.

1. Get the new DSpace 1.3.2 source code from [the DSpace page on SourceForge](#) and unpack it somewhere. Do not unpack it on top of your existing installation!!
2. Copy the PostgreSQL driver JAR to the source tree. For example:

```
cd [dspace]/lib
cp postgresql.jar [dspace-1.3.2-source]/lib
```

3. Take down Tomcat (or whichever servlet container you're using).
4. Your 'localized' JSPs (those in *jsp/local*) now need to be maintained in the *source* directory. If you have locally modified JSPs in your *[dspace]/jsp/local* directory, you will need to merge the changes in the new 1.3.2 versions into your locally modified ones. You can use the *diff* command to compare the 1.3.1 and 1.3.2 versions to do this.
5. In *[dspace-1.3.2-source]* run:

```
ant -Dconfig= [dspace]/config/dspace.cfg update
```

6. Copy the *.war* Web application files in *[dspace-1.3.2-source]/build* to the *webapps* sub-directory of your servlet container (e.g. Tomcat). e.g.:

```
cp [dspace-1.3.2-source]/build/*.war
   [tomcat]/webapps
```

If you're using Tomcat, you need to delete the directories corresponding to the old *.war* files. For example, if *dspace.war* is installed in *[tomcat]/webapps/dspace.war*, you should delete the *[tomcat]/webapps/dspace* directory. Otherwise, Tomcat will continue to use the old code in that directory.

7. Restart Tomcat.

## Upgrading From 1.2.x to 1.3.x

In the notes below *[dspace]* refers to the install directory for your existing DSpace installation, and *[dspace-1.3.x-source]* to the source directory for DSpace 1.3.x. Whenever you see these path references, be sure to replace them with the actual path names on your local system.

1. Step one is, of course, to **back up all your data** before proceeding!! Include all of the contents of *[dspace]* and the PostgreSQL database in your backup.
2. Get the new DSpace 1.3.x source code from [the DSpace page on SourceForge](#) and unpack it somewhere. Do not unpack it on top of your existing installation!!
3. Copy the PostgreSQL driver JAR to the source tree. For example: *cd [dspace]/lib cp postgresql.jar [dspace-1.2.2-source]/lib*
4. Take down Tomcat (or whichever servlet container you're using).
5. Remove the old version of *xerces.jar* from your installation, so it is not inadvertently later used: *rm [dspace]/lib/xerces.jar*



6. Install the new config files by moving *dstat.cfg* and *dstat.map* from *[dspace-1.3.x-source]/config/* to *[dspace]/config*
7. You need to add new parameters to your *[dspace]/dspace.cfg*:

```
##### Statistical Report Configuration Settings #####

# should the stats be publicly available?  should be set to false if
# you only
# want administrators to access the stats, or you do not intend to
# generate
# any
report.public = false

# directory where live reports are stored
report.dir = /dspace/reports/
```

8. Build and install the updated DSpace 1.3.x code. Go to the *[dspace-1.3.x-source]* directory, and run: *ant -Dconfig=[dspace]/config/dspace.cfg update*
9. You'll need to make some changes to the database schema in your PostgreSQL database. *[dspace-1.3.x-source]/etc/database\_schema\_12-13.sql* contains the SQL commands to achieve this. If you've modified the schema locally, you may need to check over this and make alterations. To apply the changes, go to the source directory, and run: *psql -f etc/database\_schema\_12-13.sql [DSpace database name] -h localhost*
10. Customize the stat generating statistics as per the instructions in System Statistical Reports
11. Initialize the statistics using: *[dspace]/bin/stat-initial[dspace]/bin/stat-general[dspace]/bin/stat-report-initial[dspace]/bin/stat-report-general*
12. Rebuild the search indexes: *[dspace]/bin/index-all*
13. Copy the *.war* Web application files in *[dspace-1.3.x-source]/build* to the *webapps* sub-directory of your servlet container (e.g. Tomcat). e.g.: *cp [dspace-1.3.x-source]/build/\*.war [tomcat]/webapps*
14. Restart Tomcat.

## Upgrading From 1.2.1 to 1.2.2

The changes in 1.2.2 are only code and config changes so the update should be fairly simple.

In the notes below *[dspace]* refers to the install directory for your existing DSpace installation, and *[dspace-1.2.2-source]* to the source directory for DSpace 1.2.2. Whenever you see these path references, be sure to replace them with the actual path names on your local system.

1. Get the new DSpace 1.2.2 source code from [the DSpace page on SourceForge](#) and unpack it somewhere. Do not unpack it on top of your existing installation!!
2. Copy the PostgreSQL driver JAR to the source tree. For example:

```
cd [dspace]/lib
cp postgresql.jar [dspace-1.2.2-source]/lib
```

3. Take down Tomcat (or whichever servlet container you're using).
4. Your 'localized' JSPs (those in *jsp/local*) now need to be maintained in the *source* directory. If you have locally modified JSPs in your *[dspace]/jsp/local* directory, you might like to merge the changes in the new 1.2.2 versions into your locally modified ones. You can use the *diff* command to compare the 1.2.1 and 1.2.2 versions to do this. Also see the version history for a list of modified JSPs.
5. You need to add a new parameter to your *[dspace]/dspace.cfg* for configurable fulltext indexing

```
##### Fulltext Indexing settings #####
# Maximum number of terms indexed for a single field in Lucene.
# Default is 10,000 words - often not enough for full-text indexing.
# If you change this, you'll need to re-index for the change
# to take effect on previously added items.
# -1 = unlimited (Integer.MAX_VALUE)
search.maxfieldlength = 10000
```

6. In *[dspace-1.2.2-source]* run:

```
ant -Dconfig= [dspace]/config/dspace.cfg update
```

7. Copy the *.war* Web application files in *[dspace-1.2.2-source]/build* to the *webapps* sub-directory of your servlet container (e.g. Tomcat). e.g.:

```
cp [dspace-1.2.2-source]/build/*.war
   [tomcat]/webapps
```

If you're using Tomcat, you need to delete the directories corresponding to the old *.war* files. For example, if *dspace.war* is installed in *[tomcat]/webapps/dspace.war*, you should delete the *[tomcat]/webapps/dspace* directory. Otherwise, Tomcat will continue to use the old code in that directory.

8. To finalize the install of the new configurable submission forms you need to copy the file *[dspace-1.2.2-source]/config/input-forms.xml* into *[dspace]/config*.
9. Restart Tomcat.

## Upgrading From 1.2 to 1.2.1

The changes in 1.2.1 are only code changes so the update should be fairly simple.

In the notes below *[dspace]* refers to the install directory for your existing DSpace installation, and *[dspace-1.2.1-source]* to the source directory for DSpace 1.2.1. Whenever you see these path references, be sure to replace them with the actual path names on your local system.

1. Get the new DSpace 1.2.1 source code from [the DSpace page on SourceForge](#) and unpack it somewhere. Do not unpack it on top of your existing installation!!
2. Copy the PostgreSQL driver JAR to the source tree. For example:

```
cd [dspace]/lib
cp postgresql.jar [dspace-1.2.1-source]/lib
```

3. Take down Tomcat (or whichever servlet container you're using).
4. Your 'localized' JSPs (those in *jsp/local*) now need to be maintained in the *source* directory. If you have locally modified JSPs in your *[dspace]/jsp/local* directory, you might like to merge the changes in the new 1.2.1 versions into your locally modified ones. You can use the *diff* command to compare the 1.2 and 1.2.1 versions to do this. Also see the version history for a list of modified JSPs.
5. You need to add a few new parameters to your *[dspace]/dspace.cfg* for browse/search and item thumbnails display, and for configurable DC metadata fields to be indexed.

```
# whether to display thumbnails on browse and search results pages
# (1.2+)
webui.browse.thumbnail.show = false

# max dimensions of the browse/search thumbs. Must be <=
#   thumbnail.maxwidth
# and thumbnail.maxheight. Only need to be set if required to be
#   smaller than
# dimension of thumbnails generated by mediafilter (1.2+)
#webui.browse.thumbnail.maxheight = 80
#webui.browse.thumbnail.maxwidth = 80

# whether to display the thumb against each bitstream (1.2+)
webui.item.thumbnail.show = true

# where should clicking on a thumbnail from browse/search take the
#   user
# Only values currently supported are "item" and
#   "bitstream"
#webui.browse.thumbnail.linkbehaviour = item

##### Fields to Index for Search #####

# DC metadata elements.qualifiers to be indexed for search
# format: - search.index.[number] = [search field]:element.qualifier
#           - * used as wildcard

###      changing these will change your search results,      ###
### but will NOT automatically change your search displays ###

search.index.1 = author:contributor.*
search.index.2 = author:creator.*
search.index.3 = title:title.*
search.index.4 = keyword:subject.*
search.index.5 = abstract:description.abstract
```

```
search.index.6 = author:description.statementsofresponsibility
search.index.7 = series:relation.ispartofseries
search.index.8 = abstract:description.tableofcontents
search.index.9 = mime:format.mimetype
search.index.10 = sponsor:description.sponsorship
search.index.11 = id:identifier.*
```

6. In `[dspace-1.2.1-source]` run:

```
ant -Dconfig= [dspace]/config/dspace.cfg update
```

7. Copy the `.war` Web application files in `[dspace-1.2.1-source]/build` to the `webapps` sub-directory of your servlet container (e.g. Tomcat). e.g.:

```
cp [dspace-1.2.1-source]/build/*.war
   [tomcat]/webapps
```

If you're using Tomcat, you need to delete the directories corresponding to the old `.war` files. For example, if `dspace.war` is installed in `[tomcat]/webapps/dspace.war`, you should delete the `[tomcat]/webapps/dspace` directory. Otherwise, Tomcat will continue to use the old code in that directory.

8. Restart Tomcat.

## Upgrading From 1.1 (or 1.1.1) to 1.2

The process for upgrading to 1.2 from either 1.1 or 1.1.1 is the same. If you are running DSpace 1.0 or 1.0.1, you need to follow the instructions for upgrading from 1.0.1 to 1.1 to before following these instructions.

Note also that if you've substantially modified DSpace, these instructions apply to an unmodified 1.1.1 DSpace instance, and you'll need to adapt the process to any modifications you've made.

This document refers to the install directory for your existing DSpace installation as `[dspace]`, and to the source directory for DSpace 1.2 as `[dspace-1.2-source]`. Whenever you see these path references below, be sure to replace them with the actual path names on your local system.

1. Step one is, of course, to **back up all your data** before proceeding!! Include all of the contents of `[dspace]` and the PostgreSQL database in your backup.
2. Get the new DSpace 1.2 source code from [the DSpace page on SourceForge](#) and unpack it somewhere. Do not unpack it on top of your existing installation!!
3. Copy the required Java libraries that we couldn't include in the bundle to the source tree. For example:

```
cd [dspace]/lib
cp activation.jar servlet.jar mail.jar
   [dspace-1.2-source]/lib
```

4. Stop Tomcat (or other servlet container.)
5. It's a good idea to upgrade all of the various third-party tools that DSpace uses to their latest versions:
  - Java (note that now version 1.4.0 or later is *required*)
  - Tomcat (Any version after 4.0 will work; symbolic links are no longer an issue)
  - PostgreSQL (don't forget to build/download an updated JDBC driver .jar file! Also, **back up the database** first.)
  - Ant
6. You need to add the following new parameters to your `[dspace]/dspace.cfg`:

```
##### Media Filter settings #####
# maximum width and height of generated thumbnails
thumbnail.maxwidth = 80
thumbnail.maxheight = 80
```

There are one or two other, optional extra parameters (for controlling the pool of database connections). See the version history for details. If you leave them out, defaults will be used. Also, to avoid future confusion, you might like to **remove** the following property, which is no longer required:

```
config.template.oai-web.xml =
    [dspace]/oai/WEB-INF/web.xml
```

7. The layout of the installation directory (i.e. the structure of the contents of [dspace]) has changed somewhat since 1.1.1. First up, your 'localized' JSPs (those in jsp/local) now need to be maintained in the *source* directory. So make a copy of them now! Once you've done that, you can remove [dspace]/jsp and [dspace]/oai, these are no longer used. (.war Web application archive files are used instead). Also, if you're using the same version of Tomcat as before, you need to **remove the lines from Tomcat's conf/server.xml file that enable symbolic links for DSpace**. These are the <Context> elements you added to get DSpace 1.1.1 working, looking something like this:

```
<Context path="/dspace" docBase="dspace" debug="0" reloadable="true"
  crossContext="true">
  <Resources className="org.apache.naming.resources.FileDirContext"
    allowLinking="true" />
</Context>
```

Be sure to remove the <Context> elements for both the Web UI and the OAI Web applications.

8. Build and install the updated DSpace 1.2 code. Go to the DSpace 1.2 source directory, and run:

```
ant -Dconfig= [dspace]/config/dspace.cfg update
```

9. Copy the new config files in *config* to your installation, e.g.:

```
cp [dspace-1.2-source]/config/news-*
   [dspace-1.2-source]/config/mediafilter.cfg
   [dspace-1.2-source]/config/dc2mods.cfg
   [dspace]/config
```

10. You'll need to make some changes to the database schema in your PostgreSQL database. [dspace-1.2-source]/etc/database\_schema\_11-12.sql contains the SQL commands to achieve this. If you've modified the schema locally, you may need to check over this and make alterations. To apply the changes, go to the source directory, and run:

```
psql -f etc/database_schema_11-12.sql [DSpace database name] -h
      localhost
```

11. A tool supplied with the DSpace 1.2 codebase will then update the actual data in the relational database. Run it using:

```
[dspace]/bin/dsrun
      org.dspace.administer.UpgradellTo12
```

12. Then rebuild the search indexes:

```
[dspace]/bin/index-all
```

13. Delete the existing symlinks from your servlet container's (e.g. Tomcat's) *webapp* sub-directory. Copy the .war Web application files in [dspace-1.2-source]/build to the *webapps* sub-directory of your servlet container (e.g. Tomcat). e.g.:

```
cp [dspace-1.2-source]/build/*.war
   [tomcat]/webapps
```

14. Restart Tomcat.

15. To get image thumbnails generated and full-text extracted for indexing automatically, you need to set up a 'cron' job, for example one like this:

```
# Run the media filter at 02:00 every day
0 2 * * * [dspace]/bin/filter-media
```

You might also wish to run it now to generate thumbnails and index full text for the content already in your system.

16. **Note 1:** This update process has effectively 'touched' all of your items. Although the dates in the Dublin Core metadata won't have changed (accession date and so forth), the 'last modified' date in the database for each will have been changed. This means the e-mail subscription tool may be confused, thinking that all items in the archive have been deposited that day, and could thus send a rather long email to lots of subscribers. So, it is recommended that you **turn off the e-mail subscription feature for the next day**, by commenting out the relevant line in DSpace's cron job, and then re-activating it the next day. Say you performed the update on 08-June-2004 (UTC), and your e-mail subscription cron job runs at 4am (UTC). When the subscription tool runs at 4am on 09-June-2004, it will find that everything in the system has a modification

date in 08-June-2004, and accordingly send out huge emails. So, immediately after the update, you would edit DSpace's 'crontab' and comment out the `/dspace/bin/subs-daily` line. Then, after 4am on 09-June-2004 you'd 'un-comment' it out, so that things proceed normally. Of course this means, any *real* new deposits on 08-June-2004 won't get e-mailed, however if you're updating the system it's likely to be down for some time so this shouldn't be a big problem.

17. **Note 2:** After consultation with the OAI community, various OAI-PMH changes have occurred:
- The OAI-PMH identifiers have changed (they're now of the form `oai:hostname:handle` as opposed to just Handles)
  - The set structure has changed, due to the new sub-communities feature.
  - The default base URL has changed
  - As noted in note 1, every item has been 'touched' and will need re-harvesting. The above means that, if already registered and harvested, you will need to re-register your repository, effectively as a 'new' OAI-PMH data provider. You should also consider posting an announcement to the [OAI implementers e-mail list](#) so that harvesters know to update their systems. Also note that your site may, over the next few days, take quite a big hit from OAI-PMH harvesters. The resumption token support should alleviate this a little, but you might want to temporarily whack up the database connection pool parameters in `[dspace]/config/dspace.cfg`. See the `dspace.cfg` distributed with the source code to see what these parameters are and how to use them. (You need to stop and restart Tomcat after changing them.) I realize this is not ideal; for discussion as to the reasons behind this please see relevant posts to the OAI community: [post one](#), [post two](#), as well as [this post to the dspace-tech mailing list](#). If you really can't live with updating the base URL like this, you can fairly easily have things proceed more-or-less as they are, by doing the following:
  - Change the value of `OAI_ID_PREFIX` at the top of the `org.dspace.app.oai.DSpaceOAICatalog` class to `hdl:`
  - Change the servlet mapping for the `OAIHandler` servlet back to `/` (from `/request`)
  - Rebuild and deploy `_oai.war`. However, note that in this case, all the records will be re-harvested by harvesters anyway, so you still need to brace for the associated DB activity; also note that the set spec changes may not be picked up by some harvesters. It's recommended you read the above-linked mailing list posts to understand why the change was made.
- Now, you should be finished!

## Upgrading From 1.1 to 1.1.1

Fortunately the changes in 1.1.1 are only code changes so the update is fairly simple.

In the notes below `[dspace]` refers to the install directory for your existing DSpace installation, and `[dspace-1.1.1-source]` to the source directory for DSpace 1.1.1. Whenever you see these path references, be sure to replace them with the actual path names on your local system.

1. Take down Tomcat.
2. It would be a good idea to update any of the third-party tools used by DSpace at this point (e.g. PostgreSQL), following the instructions provided with the relevant tools.
3. In `[dspace-1.1.1-source]` run:

```
ant -Dconfig=[dspace]/config/dspace.cfg update
```

4. If you have locally modified JSPs of the following JSPs in your `[dspace]/jsp/local` directory, you might like to merge the changes in the new 1.1.1 versions into your locally modified ones. You can use the `diff` command to compare the 1.1 and 1.1.1 versions to do this. The changes are quite minor.

```
collection-home.jsp
admin/authorize-collection-edit.jsp
admin/authorize-community-edit.jsp
admin/authorize-item-edit.jsp
admin/eperson-edit.jsp
```

5. Restart Tomcat.

## Upgrading From 1.0.1 to 1.1

To upgrade from DSpace 1.0.1 to 1.1, follow the steps below. Your `dspace.cfg` does not need to be changed. In the notes below `[dspace]` refers to the install directory for your existing DSpace installation, and `[dspace-1.1-source]` to the source directory for DSpace 1.1. Whenever you see these path references, be sure to replace them with the actual path names on your local system.

1. Take down Tomcat (or whichever servlet container you're using).
2. We recommend that you upgrade to the latest version of PostgreSQL (7.3.2). Included are some notes to help you do this (postgres-upgrade-notes.txt file). Note you will also have to upgrade Ant to version 1.5 if you do this.
3. Make the necessary changes to the DSpace database. These include a couple of minor schema changes, and some new indexes which should improve performance. Also, the names of a couple of database views have been changed since the old names were so long they were causing problems. First run `psql` to access your database (e.g. `psql -U dspace -W` and then enter the password), and enter these SQL commands:

```
ALTER TABLE bitstream ADD store_number INTEGER;
UPDATE bitstream SET store_number = 0;

ALTER TABLE item ADD last_modified TIMESTAMP;
CREATE INDEX last_modified_idx ON Item(last_modified);
```

```

CREATE INDEX eperson_email_idx ON EPerson(email);
CREATE INDEX item2bundle_item_idx on Item2Bundle(item_id);
CREATE INDEX bundle2bitstream_bundle_idx ON
    Bundle2Bitstream(bundle_id);
CREATE INDEX dcvalue_item_idx on DCValue(item_id);
CREATE INDEX collection2item_collection_idx ON
    Collection2Item(collection_id);
CREATE INDEX resourcepolicy_type_id_idx ON ResourcePolicy
    (resource_type_id,resource_id);
CREATE INDEX epersongroup2eperson_group_idx on
    EPersonGroup2EPerson(eperson_group_id);
CREATE INDEX handle_handle_idx ON Handle(handle);
CREATE INDEX sort_author_idx on ItemsByAuthor(sort_author);
CREATE INDEX sort_title_idx on ItemsByTitle(sort_title);
CREATE INDEX date_issued_idx on ItemsByDate(date_issued);

DROP VIEW CollectionItemsByDateAccessioned;

DROP VIEW CommunityItemsByDateAccessioned;
CREATE VIEW CommunityItemsByDateAccession as SELECT
    Community2Item.community_id, ItemsByDateAccessioned.* FROM
    ItemsByDateAccessioned, Community2Item WHERE
    ItemsByDateAccessioned.item_id = Community2Item.item_id;
CREATE VIEW CollectionItemsByDateAccession AS SELECT
    collection2item.collection_id,
    itemsbydateaccessioned.items_by_date_accessioned_id,
    itemsbydateaccessioned.item_id,
    itemsbydateaccessioned.date_accessioned FROM itemsbydateaccessioned,
    collection2item WHERE (itemsbydateaccessioned.item_id =
    collection2item.item_id);

```

- Fix your JSPs for Unicode. If you've modified the site 'skin' (*jsp/local/layout/header-default.jsp*) you'll need to add the Unicode header, i.e.:

```

<meta http-equiv="Content-Type" content="text/html;
    charset=UTF-8">

```

to the <HEAD> element. If you have any locally-edited JSPs, you need to add this page directive to the top of all of them:

```

<%@ page contentType="text/html; charset=UTF-8" %>

```

(If you haven't modified any JSPs, you don't have to do anything.)

- Copy the required Java libraries that we couldn't include in the bundle to the source tree. For example:

```

cd [dspace]/lib
cp *.policy activation.jar servlet.jar mail.jar
    [dspace-1.1-source]/lib

```

- Compile up the new DSpace code, replacing *[dspace]/config/dspace.cfg* with the path to your current, LIVE configuration. (The second line, *touch `find .`*, is a precaution, which ensures that the new code has a current timestamp and will overwrite the old code. Note that those are back quotes.)

```

cd [dspace-1.1-source]
touch `find .`
ant
ant -Dconfig= [dspace]/config/dspace.cfg update

```

- Update the database tables using the upgrader tool, which sets up the new *>last\_modified* date in the item table:

```

Run [dspace]/bin/dsrun
    org.dspace.administer.Upgrade101To11

```

8. Run the collection default authorization policy tool:

```
[dspace]/bin/dsrun  
org.dspace.authorize.FixDefaultPolicies
```

9. Fix the OAI-Cat properties file. Edit *[dspace]/config/templates/oaiCat.properties*. Change the line that says

```
Identify.deletedRecord=yes
```

To:

```
Identify.deletedRecord=persistent
```

This is needed to fix the OAI-PMH 'Identify' verb response. Then run *[dspace]/bin/install-configs*.

10. Re-run the indexing to index abstracts and fill out the renamed database views:

```
[dspace]/bin/index-all
```

11. Restart Tomcat. Tomcat should be run with the following environment variable set, to ensure that Unicode is handled properly. Also, the default JVM memory heap sizes are rather small. Adjust *-Xmx512M* (512Mb maximum heap size) and *-Xms64M* (64Mb Java thread stack size) to suit your hardware.

```
JAVA_OPTS="-Xmx512M -Xms64M -Dfile.encoding=UTF-8"
```