Upgrading DSpace

These instructions are valid for any of the following upgrade paths:

- *Upgrading ANY prior version (1.x.x, 3.x, 4.x, 5.x or 6.x) of DSpace to DSpace 6.x (latest version)*

For more information about new features or major changes in previous releases of DSpace, please refer to following:

- Releases - Provides links to release notes for all prior releases of DSpace
- Version History - Provides detailed listing of all changes in all prior releases of DSpace

Upgrading database structure/data is now automated!

The underlying DSpace database structure changes and data migrations are now AUTOMATED (using FlywayDB). This means that you no longer need to manually run SQL scripts. Instead, the first time you run DSpace, it will auto-update your database structure (as needed) and migrate all your data to be compatible with the installed version of DSpace. This allows you to concentrate your upgrade efforts on customizing your site without having to worry about migrating your data!

For example, if you were running DSpace 1.4, and you wish to upgrade to DSpace 5, you can follow the simplified instructions below. As soon as you point your DSpace 5 installation against the older DSpace 1.4-compatible database, your database tables (and data) will automatically be migrated to be compatible with DSpace 5.

See below for a specific note on troubleshooting "ignored" migrations (a rare circumstance, but known to happen if you upgrade from DSpace 5 to a later version of DSpace).
Automatic reindex of all content for search/browse after a database upgrade.

If your database structure/data is upgraded, the migration process will now trigger a reindex of all your content after deployment to Tomcat. Some repository content will be not be discoverable until this reindex process is complete. For large repository instances, this process could take some time to complete.

Optionally, you may choose to skip this automatic reindex and run a manual reindex at a later time. For more information on doing so, see the note under the optional "Upgrade your Database" step below.
Please refrain from customizing the DSpace database tables. It will complicate your next upgrade!

With the addition of our automated database upgrades, *we highly recommend AGAINST customizing the DSpace database tables/structure or backporting any features that change the DSpace tables/structure*. Doing so will often cause the automated database upgrade process to fail (and therefore will complicate your next upgrade).

If you must add features requiring new database tables/structure, we recommend creating new tables (instead of modifying existing ones), as that is usually much less disruptive to our automated database upgrade.
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.
In the notes below `[dspace]` refers to the install directory for your existing DSpace installation, and `[dspace-source]` to the source directory for DSpace. Whenever you see these path references, be sure to replace them with the actual path names on your local system.

1 Release Notes / Significant Changes
2 Backup your DSpace
3 Update Prerequisite Software (as necessary)
4 Upgrade Steps
5 Troubleshooting Upgrade Issues
5.1 "Ignored" Flyway Migrations
5.2 Manually Upgrading Solr Indexes

Release Notes / Significant Changes

DSpace 6.0 features some significant changes which you may wish to be aware of before beginning your upgrade:

- **The `build.properties` configuration file has been replaced by an enhanced `local.cfg` configuration file.** The new `local.cfg` allows you to easily override *any* configuration (from `dspace.cfg` or `modules/*.cfg` files) by simply copying it into your `local.cfg` and specifying a new value. It also provides enhanced configuration options as detailed in the Configuration Reference documentation. The old `build.properties` file is no longer used nor supported.
  - **WARNING:** As part of adding this new configuration scheme, many of the configuration settings in DSpace (primarily those in `modules/*.cfg` files) had to be renamed or prepended with the name of the module. This means that 5.x (or below) configurations are no longer guaranteed to be compatible with 6.x. If possible, we recommend starting with fresh configs (see below), and moving all your locally customized settings into the new `local.cfg` file.
  - **WARNING:** Users of Curation Tasks should note that changes to the DSpace configuration support have significantly changed the way that Task Properties work. Please see Curation System for details.
- **The PDF Citation Cover Page configuration file has been renamed** (from `disseminate-citation.cfg` to `citation-page.cfg`). See this features documentation for more details.

- **The legacy search engine (based on Apache Lucene) and legacy Browse system (based on database tables) have been removed** from DSpace 6.0 or above. Instead, DSpace now only uses Discovery (based on Apache Solr) for all Search/Browse capabilities.
- **The DSpace Lightweight Networking Interface (LNI), supporting WebDAV / SOAP / RPC API, has been removed** from DSpace 6.0 or above. We recommend using REST or SWORD (v1 or v2) as a replacement. However, if you still require it, the old (unmaintained) LNI codebase is still available at https://github.com/DSpace/dspace-lni
- **Support for SRB (Storage Resource Broker) file storage has been removed** from DSpace 6.0 or above. As it was unmaintained (and seemingly unused) for many years, this feature was removed along with its configurations. As a replacement, a new file storage plugin system was added, featuring a traditional local file storage option (default) and an Amazon S3 file storage option (see Storage Layer documentation, especially Configuring the Bitstream Store). For more information on the removal of SRB support, also see DS-3055.
- **The user groups Administrator and Anonymous cannot be renamed or deleted.** If you had renamed them, they will be renamed back to the stock names during the upgrade. DSpace is now dependent on these specific names due to internal changes.
- **XPDF PDF Thumbnail generation has been removed.** Please use the ImageMagick or PDFBox thumbnail generators instead.
- **The default strategy to create new handles for versioned Items has changed.** If you have Item Level Versioning enabled and you have versioned Items in your DSpace installation, you may want to change the configuration to continue using the mechanism to create handles as it was in DSpace 4 and 5. You can find more information here: Item Level Versioning#IdentifierServiceOverride.
- **Elasticsearch Usage Statistics feature is deprecated in the 6.0 release.** While they still function, and you can continue to use them, in a future DSpace release they will be removed. If you are looking for a usage statistics option, we recommend instead using the default SOLR Statistics engine and/or DSpace's integration with Google Analytics. See DS-2897 for more information.

Backup your DSpace

Before you start your upgrade, it is strongly recommended that you create a backup of your DSpace instance. Backups are easy to recover from; a botched install/upgrade is very difficult if not impossible to recover from. The DSpace specific things to backup are: configs, source code modifications, database, and assetstore. On your server that runs DSpace, you might additionally consider checking on your cron/scheduled tasks, servlet container, and database.

Make a complete backup of your system, including:

- Database: Make a snapshot/dump of the database. For the PostgreSQL database use Postgres' pg_dump command. For example:

  ```
  pg_dump -U [database-user] -f [backup-file-location] [database-name]
  ```

- Assetstore: Backup the directory (`[dspace]/assetstore` by default, and any other assetstores configured in the `[dspace]/config/dspace.cfg` "assetstore.dir" and "assetstore.dir.#" settings)
- Configuration: Backup the entire directory content of `[dspace]/config`.
- Customizations: If you have custom code, such as themes, modifications, or custom scripts, you will want to back them up to a safe location.
- Statistics data: what to back up depends on what you were using before: the options are the default SOLR Statistics, deprecated Elasticsearch Usage Statistics, or the legacy statistics. Legacy stats utilizes the dspace.log files, Elasticsearch stats stores data in `[dspace]/elasticsearch`, SOLR Statistics stores data in `[dspace]/solr/statistics`. A simple copy of the data directory should give you a point of recovery, should something go wrong in the update process. We can't stress this enough, your users depend on these statistics more than you realize. You need a backup.

Update Prerequisite Software (as necessary)

DSpace 6.x requires the following versions of prerequisite software:

- Java 7 or 8 (Oracle or OpenJDK)
- Apache Maven 3.0.5 or above
- Apache Ant 1.8 or above
- Database
  - PostgreSQL 9.4 or above (with pgcrypto installed), OR
  - Oracle 10g or above
- Tomcat 7 or above

If you're using a theme based on Mirage 2, you will need Maven 3.1 or above. If you are building with `mirage2.deps.included=false` (more information at https://github.com/DSpace/DSpace/tree/dspace-6_x/dspace-xmlui-mirage2) then the following dependencies may need to be updated as well:

- node, to version 6 or above
- npm, to version 3.10.8 or above

Refer to the Prerequisite Software section of "Installing DSpace" for more details around configuring and installing these prerequisites.

Upgrade Steps

1. **Ensure your database is compatible:** Starting with DSpace 6.x, there are new database requirements for DSpace (refer to the Prerequisite Software section of "Installing DSpace" for full details).
   a. *PostgreSQL databases*: PostgreSQL 9.4 or above is required and the "pgcrypto" extension must be installed.
      i. Notes on installing pgcrypto
         1. On most Linux operating systems (Ubuntu, Debian, RedHat), this extension is provided in the "postgresql-contrib" package in your package manager. So, ensure you've installed "postgresql-contrib".
         2. On Windows, this extension should be provided automatically by the installer (check your "[PostgreSQL]/share/extension" folder for files starting with "pgcrypto")

ii. Enabling pgcrypto on your DSpace database. (Additional options/notes in the Installation Documentation)

```
# Login to your "dspace" database as a superuser
psql --username=postgres dspace
# Enable the pgcrypto extension on this database
CREATE EXTENSION pgcrypto;
```

b. *Oracle databases:* Oracle database have no additional requirements at this time.

2. **Download DSpace 6.x:** Either download DSpace 6.x from DSpace.org or check it out directly from the Github repository.
   a. NOTE: If you downloaded DSpace do not unpack it on top of your existing installation. Refer to Installation Instructions, Step 3 for unpacking directives.
3. **Merge any User Interface customizations or other customizations (if needed or desired).** If you have made any local customizations to your DSpace installation they *may* need to be migrated over to the new DSpace.
   a. NOTE: If you are upgrading across many versions of DSpace at once (e.g. from 1.x.x to 6.x), you may find it easier to *first* upgrade DSpace, and *then* attempt to migrate over your various customizations. Because each major version of DSpace tends to add new configurations and features to the User Interface, older customizations may require more work to "migrate" to the latest version of DSpace. In some situations, it may even be easier to "start fresh", and just re-customize the brand new User Interface with your local color scheme, header/footer, etc.
   b. Customizations are typically housed in one of the following places:
      i. JSPUI modifications: `[dspace-source]/dspace/modules/jspui/src/main/webapp/`
      ii. XMLUI modifications: `[dspace-source]/dspace/modules/xmlui/src/main/webapp/`
      iii. Config modifications: `[dspace]/config`
   c. For highly customized DSpace instances, note that the format of the following configuration files has changed. If you have customized these configuration files, carefully re-integrate your custom settings.
      i. `dspace/config/dspace.cfg`
         - `Note the escaped comma (\,) separator is now needed between element names: https://github.com/DSpace/DSpace/blob/dspace-6.0-rc3/dspace/config/dspace.cfg#L1020`
      ii. `dspace/config/spring/api/discovery.xml`
         - The following property has been removed
            - <property name="sortOrder" value="COUNT"/>
         - And has been replaced with the following
            - <property name="sortOrderSidebar" value="COUNT"/>
            - <property name="sortOrderFilterPage" value="COUNT"/>
      iii. dspace/modules/xmlui/src/main/webapp/sitemap.xmap
         - Note the presence of new class names in this file. In particular,note the removal of StandardOpenSearchGenerator
4. **Replace your old build.properties file with a local.cfg** *(REQUIRED if upgrading from DSpace 5 or previous):* As of DSpace 6.0, the `build.properties` configuration file has been replaced by an enhanced `local.cfg` configuration file. Therefore, any old `build.properties` file (or similar `[dspace-source]/*.properties` files) WILL BE IGNORED. Instead, you should create a new `local.cfg` file, based on the provided `[dspace-source]/dspace/config/local.cfg.EXAMPLE` and use it to specify all of your locally customized DSpace configurations. This new `local.cfg` can be used to override ANY setting in any other configuration file (`dspace.cfg` or `modules/*.cfg`). To override a default setting, simply copy the configuration into your `local.cfg` and change its value(s). For much more information on the features of local.cfg, see the Configuration Reference documentation and the local.cfg Configuration File section on that page.

```
cd [dspace-source]
cp dspace/config/local.cfg.EXAMPLE local.cfg

# Then edit the local.cfg, specifying (at a minimum) your basic DSpace configuration settings.
# Optionally, you may copy any settings from other *.cfg configuration files into your local.cfg to
override them.
# After building DSpace, this local.cfg will be copied to [dspace]/config/local.cfg, where it will also
be used at runtime.
```

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

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

The above command will re-compile the DSpace source code and build its "installer". You will find the result in `[dspace-source]/dspace/target/dspace-installer`

Defaults to PostgreSQL settings

Without any extra arguments, the DSpace installation package is initialized for PostgreSQL. If you use Oracle instead, you should build the DSpace installation package as follows:
`mvn -Ddb.name=oracle -U clean package`

Enabling and building the optional Mirage 2 theme

Mirage 2 is a responsive theme for the XML User Interface, added as a new feature in DSpace 5. It has not yet replaced the Mirage 1 theme as the XMLUI default theme.

To enable Mirage 2, add the following to the `<themes>` section of `src/dspace/config/xmlui.xconf` , replacing the currently active theme:

```
<theme name="Mirage 2" regex=".*" path="Mirage2/" />
```

It is important to do this before executing the maven build.

Mirage 2 is not yet activated in the default "mvn package" build. To include it as part of the build, run:

```
mvn -U clean package -Dmirage2.on=true
```

The speed of this specific step of the build can be increased by installing local copies of the specific dependencies required for building Mirage 2. The Mirage 2 developer documentation provides detailed instructions for these installations. After the installation of these dependencies, you can choose to run:

```
mvn -U clean package -Dmirage2.on=true -Dmirage2.deps.included=false
```

Warning: The Mirage 2 build process should NOT be run as "root". It must be run as a non-root user. For more information see: Mirage 2 Common Build Issues

6. **Stop Tomcat (or servlet container).** Take down your servlet container.
    a. For Tomcat, use the `$CATALINA_HOME/shutdown.sh` script. (Many Unix-based installations will have a startup/shutdown script in the `/etc/init.d` or `/etc/rc.d` directories.)
7. **Update DSpace Installation.** Update the DSpace installation directory with the new code and libraries. Issue the following commands:

```
cd [dspace-source]/dspace/target/dspace-installer
ant update
```

The above command will also automatically upgrade all your existing Solr indexes (e.g. for Discovery, Statistics, OAI-PMH) to the latest version. For large instances, this may take some time. But, it is important to ensure that your indexes are usable by the latest version of DSpace.

    a. If the Solr index upgrade fails, you may need to Manually Upgrade your Solr Indexes. See the "Troubleshooting Upgrade Issues" section below.
8. **Update your DSpace Configurations and/or move them to local.cfg** *(REQUIRED if upgrading from DSpace 5 or previous)*: You should review your configuration for new and changed configurations in DSpace 6.x.
    a. As mentioned above, DSpace 6.0 now includes a new local.cfg Configuration File. So, rather than editing the `dspace.cfg` (or any of the `modules/*.cfg`), it's *recommended* to simply override the default values in your own `local.cfg`. That way, your `local.cfg` can serve as the record of which configurations you have actually tweaked in your DSpace, which may help to simplify future upgrades.
        i. **WARNING:** in order to create this powerful ability to override configurations in your local.cfg, all `modules/*.cfg` files had their configurations **renamed** to be pre-pended with the module name. As a basic example, all the configuration settings within the `modules/oai.cfg` configuration now start with `"oai."`. *Unfortunately, these means that DSpace 5.x configuration files are NOT guaranteed to be compatible with DSpace 6.* For more information on configurations in DSpace 6 see our updated Configuration Reference.
    b. **Search/Browse requires Discovery:** As of DSpace 6, only Discovery (Apache Solr) is supported for search/browse. Support for Legacy Search (using Apache Lucene) and Legacy Browse (using database tables) has been removed, along with all their configurations.
    c. **XPDF media filtering no longer exists:** XPDF media filtering, deprecated in DSpace 5, has been removed. If you used this, you will need to reconfigure using the remaining alternatives (e.g. PDF Text Extractor and/or ImageMagick PDF Thumbnail Generator)
    d. **Upgrading Configurable/XML Workflow may require minor configuration updates.** If you are currently running the DSpace XMLUI with Configurable/XML Workflow enabled, you may need to *re-enable its configurations in the DSpace 6 configuration files prior to upgrading*. As with past releases, DSpace 6 defaults to using Basic (Traditional) Workflow. Therefore, you should double check the settings required to enable Configurable Workflow in DSpace 6. Pay close attention to the fact that, to enable Configurable Workflow in DSpace 6, all BasicWorkflow settings must also be commented out (in several configs).
    e. It is recommended to review all configuration changes that exist in the `config` directory, and its subdirectories. It is helpful to compare your current configs against a clean checkout of your current version to see what you have customized. You might then also want to compare your current configs with the configs of the version you are upgrading to. A tool that compares files in directories such as Meld or DiffMerge is useful for this purpose.
        i. After reviewing which configurations you've changed, we recommend moving all your customized configurations into your `local.cfg` file, as described above. Examples of how this might be accomplished are provided in the Configuration Reference.
    f. **CHANGE IN DSpace 6.3:** IP Address to geographic location database has been renamed. The old `[dspace]/config/GeoLiteCity.dat` file is no longer maintained by its provider. You can delete it. The new file is named `GeoLite2-City.mmdb` by default. The upgrade process will automatically download a copy of the new database if you don't already have it. If you have configured a different name and/or location for this file, you should check the setting of `usage-statistics.dbfile` in `[dspace]/config/modules/usage-statistics.cfg` (and perhaps move your custom setting to `local.cfg`).
9. **Decide which DSpace Web Applications you want to install.** DSpace comes with a variety of web applications (in `[dspace]/webapps`), each of which provides a different "interface" to your DSpace. Which ones you install is up to you, but there are a few that we highly recommend (see below):

    a. "xmlui" = This is the XML-based User Interface, based on Apache Cocoon. It comes with a variety of out-of-the-box themes, including Mirage 1 (the default) and Mirage 2 (based on Bootstrap).*Between the "xmlui" and "jspui", you likely only need to choose one.*
    b. "jspui" = This is the JSPUI-based User Interface, which is based on Bootstrap. *Between the "xmlui" and "jspui", you likely only need to choose one.*
    c. "solr" (*required*) = This is Apache Solr web application, which is used by the "xmlui" and "jspui" (for search & browse functionality), as well as the OAI-PMH interface. **It must be installed in support of either UI.**

d. "oai" = This is the DSpace OAI interface. It allows for metadata and bitstream (content-file) harvesting, supporting OAI-PMH (Protocol for Metadata Harvest) and OAI-ORE (Object Reuse and Exchange) protocols
e. "rest" = This is the DSpace REST API
f. "sword" = This is the DSpace SWORDv1 interface. More info on SWORD protocol and its usage.
g. "swordv2" = This is the DSpace SWORDv2 interface. More info on SWORD protocol and its usage.
h. "rdf" = This is the DSpace RDF interface supporting Linked (Open) Data.

10. **Deploy DSpace Web Applications**. If necessary, copy the web applications from your `[dspace]/webapps` directory to the subdirectory of your servlet container (e.g. Tomcat):

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

See the installation guide for full details.

11. **Upgrade your database** (*optional, but recommended for major upgrades*). As of DSpace 5 (and above), the DSpace code will automatically upgrade your database (*from any prior version of DSpace*). By default, this database upgrade occurs automatically when you restart Tomcat (or your servlet container). However, if you have a large repository or are upgrading across multiple versions of DSpace at once, you may wish to manually perform the upgrade (as it could take some time, anywhere from 5-15 minutes for large sites).
    a. First, you can optionally verify whether DSpace correctly detects the version of your DSpace database. It is **very important** that the DSpace version is detected correctly before you attempt the migration:

    ```
    [dspace]/bin/dspace database info
    # Look for a line at the bottom that says something like:
    # "Your database looks to be compatible with DSpace version ___"
    ```

    b. In some rare scenarios, if your database's "sequences" are outdated, inconsistent or incorrect, a database migration error may occur (in your DSpace logs). While this is seemingly a rare occurance, you may choose to run the "update-sequences.sql" script PRIOR to upgrading your database. If your database sequences are inconsistent or incorrect, this "update-sequences.sql" script will auto-correct them (otherwise, it will do nothing). Be aware, if you choose to run this script, please run the "update-sequences.sql" file that corresponds to your *current version of DSpace* (i.e. the version you are upgrading *from*), as this script occasionally changes between releases. In the future, we hope to automate this step to avoid any sequence problems:

    ```
    # As of 6.4, this script can instead be run via
    [dspace]/bin/dspace update-sequences

    # For 6.3 and below, you have to run this via database tools
    # General PostgreSQL example
    psql -U [database-user] -f [dspace]/etc/postgres/update-sequences.sql [database-name]

    # Example for a PostgreSQL database named "dspace", and a user account named "dspace"
    # psql -U dspace -f [dspace]/etc/postgres/update-sequences.sql dspace
    ```

    c. Then, you can upgrade your DSpace database to the latest version of DSpace. (NOTE: check the DSpace log, `[dspace]/log/dspace.log.[date]`, for any output from this command)

    ```
    [dspace]/bin/dspace database migrate
    ```

    d. The database migration should also *automatically trigger* your metadata/file registries to be updated (based on the config files in [dspace]/config/registries/). However, if this update was NOT triggered, you can also manually run these registry updates (they will not harm existing registry contents) as follows:

    ```
    [dspace]/bin/dspace registry-loader -metadata [dspace]/config/registries/dcterms-types.xml
    [dspace]/bin/dspace registry-loader -metadata [dspace]/config/registries/dublin-core-types.xml
    [dspace]/bin/dspace registry-loader -metadata [dspace]/config/registries/eperson-types.xml
    [dspace]/bin/dspace registry-loader -metadata [dspace]/config/registries/local-types.xml
    [dspace]/bin/dspace registry-loader -metadata [dspace]/config/registries/sword-metadata.xml
    [dspace]/bin/dspace registry-loader -metadata [dspace]/config/registries/workflow-types.xml
    ```

    e. *If the database upgrade process fails or throws errors*, then you likely have manually customized your database structure (and/or backported later DSpace features to an older version of DSpace). In this scenario, you may need to do some manual migrations before the automatic migrations will succeed. The general process would be something like this:
        i. Revert back to your current DSpace database
        ii. Manually upgrade just your database **past** the failing migration. For example, if you are current using DSpace 1.5 and the "V1.6" migration is failing, you may need to first manually upgrade your database to 1.6 compatibility. This may involve either referencing the upgrade documentation for that older version of DSpace, or running the appropriate SQL script from under `[dspace-src]/dspace-api/src/main/resources/org/dspace/storage/rdbms/sqlmigration/`)
        iii. Then, re-run the migration process from that point forward (i.e. re-run `./dspace database migrate`)
    f. More information on the "database" command can be found in Database Utilities documentation.

By default, your site will be automatically reindexed after a database upgrade

If any database migrations are run (even during minor release upgrades), then by default DSpace will automatically reindex all content in your site. This process is run automatically in order to ensure that any database-level changes are also immediately updated within the search/browse interfaces. See the notes below under "**Restart Tomcat (servlet container)**" for more information.

However, you may choose to **skip automatic reindexing.** Some sites choose to run the reindex process manually in order to better control when /how it runs.

Skipping automatic reindexing involves two main steps, *both of which must be executed prior to restarting Tomcat (or your servlet container):*

1) First, you **must** manually update your database by running `[dspace]/bin/dspace database migrate` (See above for additional notes on this process)

2) If any database migrations are run, a empty "reindex.flag" file will be created at `[dspace]/solr/search/conf/reindex.flag` The existence of this file is what triggers the automatic reindexing when Tomcat starts up. Simply delete the "reindex.flag" file, and your site will now skip automatic reindexing. (NOTE: this file will only ever exist temporarily, and is only created when your database is modified during an upgrade or migration)

As you have disabled automatic reindexing, make sure to manually reindex your site by running `[dspace]/bin/dspace discovery -b` *(This must be run after restarting Tomcat)*

*WARNING: It is not recommended to skip automatic reindexing, unless you will **manually reindex** at a later time, or have verified that a reindex is not necessary. Forgetting to reindex your site after an upgrade may result in unexpected errors or instabilties.*

12. **Sites with Oracle database backends (and Configurable Workflow enabled) may need to run a "repair" on your database.**
    a. In version 6.3, we fixed an Oracle migration issue related to Configurable (XML) Workflow. See DS-3788.
    b. If you are upgrading an Oracle-based site to 6.3 from 6.0, 6.1 or 6.2 AND had Configurable Workflow already enabled, then you will need to manually "repair" your database to align it with the latest schema. *This does not affect PostgreSQL-based backends or any sites that are upgrading from 5.x or below.*
    c. Simply run the following to repair your Oracle database: `[dspace]/bin/dspace database repair`
13. **Restart Tomcat (servlet container).** Now restart your servlet container (Tomcat/Jetty/Resin) and test out the upgrade.
    a. **Upgrade of database:** If you didn't manually upgrade your database in the previous step, then your database will be automatically upgraded to the latest version. This may take some time (seconds to minutes), depending on the size of your repository, etc. Check the DSpace log (`[dspace]/log/dspace.log.[date]`) for information on its status.
    b. **Reindexing of all content for search/browse:** If your database was just upgraded (either manually or automatically), all the content in your DSpace will be automatically re-indexed for searching/browsing. As the process can take some time (minutes to hours, depending on the size of your repository), it is performed in the background; meanwhile, DSpace can be used as the index is gradually filled. *But, keep in mind that not all content will be visible until the indexing process is completed.* Again, check the DSpace log (`[dspace]/log/dspace.log.[date]`) for information on its status.
        i. *If you wish to skip automatic reindexing, please see the Note above under the "Upgrade your Database" step.*
14. **Reindex SOLR Stats** (*only necessary if upgrading from 4.x or below*). If you were previously using SOLR stats, the schema changed with DSpace 5; if you are upgrading from any version earlier than 5.x, you will need to reindex your stats in order to ensure all of your stats data conforms to the new schema specification. NOTE: it is safe to run a reindex on a live site, the script will store incoming usage data in a temporary core.
15. **Check your cron / Task Scheduler jobs.** In recent versions of DSpace, some of the scripts names have changed.
    a. Check the Scheduled Tasks via Cron documentation for details. Especially pay attention to the Solr Index optimization commands, which ideally should be run regularly (as noted in the previous step).
    b. WINDOWS NOTE: If you are running the Handle Server on a Windows machine, a new `[dspace]/bin/start-handle-server.bat` script is available to more easily startup your Handle Server.

Troubleshooting Upgrade Issues

"Ignored" Flyway Migrations

In very rare instances, a Flyway database migration will be "ignored." One known instance of this is documented in DS-3407. If you are upgrading from DSpace 5.x to a later version of DSpace, the migration put in place to address DS-2818 will be "ignored" because it is not necessary. There is a special command you can run which will un-flag this migration as "ignored."

```
dspace database migrate ignored
```

warning: dangerous command: BACK UP YOUR DATABASE!

The `dspace database migrate ignored` command can be dangerous: it will attempt to re-run *all* ignored migrations. In the case outlined above, this is safe. However, under other circumstances, re-running ignored migrations can lead to unpredictable results. To be absolutely safe, be sure you have a current backup of your database.

The presence of `ignored` migrations can indicate a problem in the database. It's best not to use this command unless instructed to.

Manually Upgrading Solr Indexes

If you run into issues with the auto-upgrade of your Solr search/browse indexes (during the final part of the `ant update step`), then you may need to manually upgrade your Solr indexes. Depending on the type of failure, there are a few possible fixes.

1. If the "ant update" process failed to download the `lucene-core-3.5.0.jar`, in order to upgrade a DSpace 1.6.x, 1.7.x or 1.8.x index.
    a. You can manually download the `lucene-core-3.5.0.jar` from http://search.maven.org/remotecontent?filepath=org/apache/lucene/lucene-core/3.5.0/lucene-core-3.5.0.jar
    b. Place the lucene-core-3.5.0.jar in your `[dspace-source]/dspace/target/dspace-installer/` directory (i.e. the directory where you ran "ant update" from)
    c. Re-Run "`ant update`". This time, it should find the `lucene-core-3.5.0.jar` locally and re-attempt the upgrade of your Solr indexes.
2. If some other error occurred, you may need to **manually** upgrade your Solr indexes.
    a. **Upgrading from DSpace 1.6.x, 1.7.x or 1.8.x:** In DSpace 1.x versions, we used and older version of Solr which is no longer compatible with the current version of Solr.
        i. If you are using an older version of DSpace, you will see errors similar to this one until you manually upgrade your index:

```
Caused by: org.apache.lucene.index.IndexFormatTooOldException: Format version is not
supported (resource: segment _386q in resource ChecksumIndexInput(MMapIndexInput(path="
/space/dspace/solr/statistics/data/index/segments_37m6"))): 2.x. This version of Lucene
only supports indexes created with release 3.0 and later.
```

        ii. Manually upgrading your Solr index involves temporarily downloading an older version of Lucene (on which Solr is based), and calling its IndexUpgrader script, e.g.

```
# Download Lucene 3.5.0, which can upgrade older Solr/Lucene indexes
wget "http://search.maven.org/remotecontent?filepath=org/apache/lucene/lucene-core/3.5.0
/lucene-core-3.5.0.jar" -O lucene-core-3.5.0.jar

# Then, actually upgrade the indexes by loading the lucene-core-3.5.0.jar and calling
IndexUpgrader

# Upgrade the Usage Statistics index. Run this if you have Solr Usage Statistics enabled in
your UI.
java -cp lucene-core-3.5.0.jar org.apache.lucene.index.IndexUpgrader [dspace]/solr
/statistics/data/index/

# Upgrade the OAI-PMH indexes. Run this if you use the "oai" webapp.
java -cp lucene-core-3.5.0.jar org.apache.lucene.index.IndexUpgrader [dspace]/solr/oai/data
/index/

# NOTE: You do not need to upgrade the Discovery Search and Browse indexes as they will be
automatically rebuilt on upgrade (See previous upgrade step)
```

        iii. At this point in time, your older indexes will now be compatible with Solr / Lucene 3.5. At this point they are readable by the latest version of Solr.
        iv. However, as a final step, you should still **optimize** each of these indexes using the commands detailed in the "Upgrading from DSpace 3.x or Above" step below
    b. **Upgrading from DSpace 3.x or above:** DSpace provides optimization commands for all Solr indexes. Which ones you need to run depend on which features you are using in DSpace.

```
# First, ensure your Tomcat is started up. All of the below commands will call Solr directly,
which requires Tomcat to be running.

# Optimize Usage Statistics (based on Solr). Run this if you have Usage Statistics enabled in your
UI.
[dspace]/bin/dspace stats-util -o

# Optimize OAI-PMH indexes (based on Solr). Run this if you use the "oai" webapp.
[dspace]/bin/dspace oai import -o

# NOTE: You should not need to optimize the Discovery Search and Browse indexes, as they will be
automatically rebuilt on upgrade (See previous upgrade step)
# However, you still may wish to schedule optimizing of Discovery Search & Browse (via cron or
similar)
# [dspace]/bin/dspace index-discovery -o
```