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, 6.x or 7.x) of DSpace to DSpace 7.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).

⚠ **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.

Release Notes / Significant Changes

DSpace 7.0 features some significant changes which you may wish to be aware of before beginning your upgrade:

- **XMLUI and JSPUI are no longer supported or distributed with DSpace.** All users should install and utilize the new Angular User Interface. See the "Installing the Frontend (User Interface)" instructions in Installing DSpace

- **Solr must be installed separately** due to changes in the packaging of recent Solr releases. The indexes have been reconfigured and must be rebuilt. See below.
- **GeoIP location database must be installed separately** due to changes in Maxmind's terms and conditions. MaxMind has changed the terms and procedure for obtaining and using its GeoLite2 location database. Consequently, DSpace no longer automatically downloads the database during installation or update, and the DSpace-specific database update tool has been removed. If you wish to (continue to) record client location data in SOLR Statistics, you will need to make new arrangements. See below.
- **ElasticSearch Usage Statistics have been removed.** Please use SOLR Statistics or DSpace Google Analytics Statistics.
- **Configuration has been upgraded to Apache Commons Configuration version 2.** For most users, you should see no effect or difference. *No DSpace configuration files were modified during this upgrade and no configurations or settings were renamed or changed.* However, if you locally modified or customized the [dspace]/config/config-definition.xml (DSpace's Apache Commons Configuration settings), you will need to ensure those modifications are compatible with Apache Commons Configuration version 2. See the Apache Commons Configuration's configuration definition file reference for more details.
- **Handle server has been updated to v9.3.** Most users will see no effect or difference, however a minor change to the Handle Server configuration is necessary, see below.
- **Many backend prerequisites have been upgraded to avoid "end of life" versions.** Therefore, pay very close attention to the required prerequisites listed below.

Upgrading the Backend (Server API)

Backup your DSpace Backend

Before you start your upgrade, it is strongly recommended that you create a backup of your DSpace content. 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 [dspace]/config/spring /api/bitstore.xml)
- 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, or the legacy statistics. Legacy stats utilizes the dspace.log files, while SOLR Statistics stores data in [dspace]/solr/statistics. A simple copy of the logs or the Solr core directory tree 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.
- Authority data: stored in [dspace]/solr/authority. As with the statistics data, making a copy of the directory tree should enable recovery from errors.

Update Backend Prerequisite Software

DSpace 7.x requires the following versions of prerequisite software:

- **Updated:** Java 11 (Oracle or OpenJDK)
- **Updated:** Apache Maven 3.3.x or above
- **Updated:** Apache Ant 1.10.x or above
- Database
    - **Updated:** PostgreSQL 11 (with pgcrypto installed), OR
    - Oracle 10g or above
- **Updated:** Tomcat 9.x
- **New:** Solr 8.x. *See step #11 below.*
- **New:** (optional) IP to City Database for location-based Statistics (either MaxMind's GeoLite City Database or DB-IP's City Lite Database). *See step #13 below.*

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

Upgrading the Backend Steps

1. **Ensure that your database is compatible:** Starting with DSpace 6.x, there are new database requirements for DSpace (refer to the Backend Requirements 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. **From your old version of DSpace, dump your** `authority` **and** `statistics` **Solr cores.** (Only necessary when upgrading from DSpace 6 or older & you want to keep both your authority records and/or [SOLR Statistics](#))

```
[dspace]/bin/dspace solr-export-statistics -i authority
[dspace]/bin/dspace solr-export-statistics -i statistics
```

The dumps will be written to the directory `[dspace]/solr-export`. *This may take a long time and require quite a lot of storage.* In particular, the statistics core is likely to be huge, perhaps double the size of the content of `solr/statistics/data`. You should ensure that you have sufficient free storage.

This is not the same as the disaster-recovery backup that was done above. These dumps will be reloaded into new, reconfigured cores [later](#).

If you are **sharding** your statistics data, [you will need to dump each shard separately. The index names for prior years will be](#) `statistics-YYYY` [(for example:](#) `statistics-2017 statistics-2018` etc.) The current year's statistics shard is named `statistics` and you should dump that one too.

3. **Download** the [latest DSpace release](#) from the DSpace GitHub Repository. You can choose to either download the zip or tar.gz file provided by GitHub, or you can use "git" to checkout the appropriate tag (e.g. `dspace-7.0`) or branch.
   a. Unpack it using "unzip" or "gunzip". If you have an older version of DSpace installed on this same server, you may wish to unpack it to a different location than that release. This will ensure no files are accidentally overwritten during the unpacking process, and allow you to compare configs side by side.
   b. For ease of reference, we will refer to the location of this unzipped version of the DSpace release as *[dspace-source]* in the remainder of these instructions.

4. **Replace your old build.properties file with a local.cfg** *(ONLY 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 Backend.** Run the following commands to compile DSpace :

```
cd [dspace-source]
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`

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
```

8. **Update your DSpace Configurations.** Depending on the version of DSpace you are upgrading from, not all steps are required.

a. **If you are upgrading from DSpace 5.x or below.** *If you are upgrading from DSpace 6.x you can skip these steps and move to the next bullet.*
    i. **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.
    ii. **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)
b. **If you are upgrading from DSpace 6.x or below.** *All administrators will need to perform these steps.*
    i. **Review your customized configurations (recommended to be in local.cfg):** As mentioned above, we recommend any local configuration changes be placed in a local.cfg Configuration File. With any major upgrade some configurations may have changed. Therefore, it is recommended to review all configuration changes that exist in the `config` directory, and its subdirectories, concentrating on configurations your previously customized in your local.cfg. See also the Configuration Reference.
    ii. **Remove obsolete configurations.** With the removal of the JSPUI and XMLUI, a large number of server-side (backend) configurations were made obsolete and were therefore removed between the 6.x and 7.0 release. A full list can be found in the Release Notes.
    iii. **Migrate or recreate your Submission configuration.** As of DSpace 7, the submission configuration has changed. The format of the "item-submission.xml" file has been updated, and the older "input-forms.xml" has been replaced by a new "submission-forms.xml". You can choose to either start fresh with the new v7 configuration files, or you can use the steps below to migrate your old configurations into the new format. See the Submission User Interface for more information
        1. First, create a temporary folder to copy your old v6 configurations into

        ```
        # Example of creating a [dspace]/config/temp folder for this migration
        # You must replace [dspace] with the full path of your DSpace 7 installation.
        cd [dspace]/config
        mkdir temp
        ```

        2. Copy your old (v5 or v6) "item-submission.xml" and "input-forms.xml" into that temporary folder
        3. Run the command-line migration script to migrate them to v7 configuration files

        ```
        # This example uses [dspace] as a placeholder for all paths.
        # Replace it with either the absolute or relative path of these files
        [dspace]/bin/dspace submission-forms-migrate -s [dspace]/config/temp/item-submission.
        xml -f [dspace]/config/temp/input-forms.xml
        ```

        4. The result will be two files. These are valid v7 configurations based on your original submission configuration files.
            a. `[dspace]/config/item-submission.xml.migrated`
            b. `[dspace]/config/submission-forms.xml.migrated`
        5. These "*.migrated" files have *no inline comments*, so you may want to edit them further before installing them (by removing the ".migrated" suffix). Alternatively, you may choose to copy sections of the *.migrated files into the default configurations in the `[dspace]/config/` folder, therefore retaining the inline comments in those default files.
    iv. **City IP Database file for Solr Statistics 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. 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`).
    v. **tm-extractors media filtering (WordFilter) no longer exists:** the PoiWordFilter plugin now fulfills this function. If you still have WordFilter configured, remove from `dspace.cfg` and/or `local.cfg` all lines referencing `org.dspace.app.mediafilter.WordFilter` and uncomment all lines referencing `org.dspace.app.mediafilter.PoiWordFilter`.
    vi. **Re-configure Solr URLs:** change the value of `solr.server` to point at your new Solr external service. It will probably become something like `solr.server = https://${dspace.hostname}:8983/solr`. Also review the values of
        1. `discovery.search.server`
        2. `oai.solr.url`
        3. `solr.authority.server`
        4. `solr.statistics.server`
    vii. **Sitemaps are now automatically generated/updated:** A new `sitemap.cron` setting exists in the dspace.cfg which controls when Sitemaps are generated. By default they are enabled to update once per day, for optimal SEO. See Search Engine Optimization docs for more detail
        1. Because of this change, if you had a system cron job which ran "./dspace generate-sitemaps", this system cron job can be removed in favor of the new `sitemap.cron` setting.
9. **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 occurrence, you may choose to run the "update-sequences" command PRIOR to upgrading your database. If your database sequences are inconsistent or incorrect, this "update-sequences" command will auto-correct them (otherwise, it will do nothing).

```
# 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
```

If you are upgrading from DSpace 6 or earlier, there are database changes which were previously optional but now are mandatory (specifically Configurable Workflow database changes). Instead of (or after) the above command:

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

to apply these changes.

d. *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`)
e. 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.

To disable automatic reindexing, set `discovery.autoReindex = false` in `config/local.cfg` or `config/modules/discovery.cfg`.

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.*

⚠ **Sites with Oracle database backends (and Configurable Workflow enabled) may need to run a "repair" on your database.**

In version 6.3, we fixed an Oracle migration issue related to Configurable (XML) Workflow. See DS-3788.

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.*

Simply run the following to repair your Oracle database: `[dspace]/bin/dspace database repair`

10. **Deploy Server web application:** The DSpace backend consists of a single "server" webapp (in `[dspace]/webapps/server`). You need to deploy this webapp into your Servlet Container (e.g. Tomcat). Generally, there are two options (or techniques) which you could use...either configure Tomcat to find the DSpace "server" webapp, or copy the "server" webapp into Tomcat's own webapps folder. For more information & example commands, see the Installation Guide

    a. Optionally, you may also install the deprecated DSpace 6.x REST API web application ("rest" webapp). If you previously used the DSpace 6.x REST API, for backwards compatibility the old, deprecated "rest" webapp is still available to install (in `[dspace]/webapps/rest`). It is NOT used by the DSpace UI/frontend. So, most users should skip this step.
11. **Install new Solr cores and rebuild your indexes.** (Only necessary if upgrading from 6.x or below)

a. Copy the new, empty Solr cores to your new Solr instance.

```
cp -R [dspace]/solr/* [solr]/server/solr/configsets
chown -R solr:solr [solr]/server/solr/configsets
```

b. Start Solr, or restart it if it is running, so that these new cores are loaded.

```
[solr]/bin/solr restart
```

c. Load `authority` and `statistics` from the dumps that you made earlier (not the disaster-recovery backup).

```
[dspace]/bin/dspace solr-import-statistics -i authority
[dspace]/bin/dspace solr-import-statistics -i statistics
```

This could take quite some time.

If you had sharded your statistics, you will need to load the dump of each shard separately. As when dumping, the index names will be `... statistics-2017 statistics-2018 statistics`.

d. For Statistics shards only, upgrade legacy DSpace Object Identifiers (pre-6.4 statistics) to UUID Identifiers.

```
[dspace]/bin/dspace solr-upgrade-statistics-6x -i statistics
```

Again If you had sharded your statistics, you will need to run this for each shard separately. See also SOLR Statistics Maintenance#UpgradeLegacyDSpaceObjectIdentifiers(pre-6xstatistics)toDSpace6xUUIDIdentifiers

e. Rebuild the `oai` and `search` cores.

```
[dspace]/bin/dspace oai import
[dspace]/bin/dspace index-discovery -b
```

If you have a great deal of content, this could take a long time.

12. **Update Handle Server Configuration.** (Only necessary if upgrading from 6.x or below) Because we've updated to Handle Server v9, if you are using the built-in Handle server (most installations do), you'll need to add the follow to the end of the *server_config* section of your `[dspace]/handle-server/config.dct` file (the only new line is the "enable_txn_queue" line)

```
"case_sensitive" = "no"
"storage_type" = "CUSTOM"
"storage_class" = "org.dspace.handle.HandlePlugin"
"enable_txn_queue" = "no"
```

a. Alternatively, you could re-run the `./dspace make-handle-config` script, which is in charge of updating this `config.dct` file.

13. **(Optional) Set up IP to City database for location-based statistics.** If you wish to (continue to) record the geographic origin of client activity, you will need to install (and regularly update) one of the following:
    - Either, a copy of MaxMind's GeoLite City database (in MMDB format)
        - NOTE: Installing MaxMind GeoLite2 is *free.* However, you **must** sign up for a (free) MaxMind account in order to obtain a license key to use the GeoLite2 database.
        - You may download GeoLite2 directly from MaxMind, or many Linux distributions provide the `geoipupdate` tool directly via their package manager. You will still need to configure your license key prior to usage.
        - Once the "GeoLite2-City.mmdb" database file is installed on your system, you will need to configure its location as the value of `usage-statistics.dbfile` in your `local.cfg` configuration file .
        - You can discard any old GeoLiteCity.dat database(s) found in the `config/` directory (if they exist).
        - See the "Managing the City Database File" section of SOLR Statistics for more information about using a City Database with DSpace.
    - Or, you can alternatively use/install DB-IP's City Lite database (in MMDB format)
        - This database is also free to use, but does **not** require an account to download.
        - Once the "dbip-city-lite.mmdb" database file is installed on your system, you will need to configure its location as the value of `usage-statistics.dbfile` in your `local.cfg` configuration file .
        - See the "Managing the City Database File" section of SOLR Statistics for more information about using a City Database with DSpace.
14. **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.*
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. If you have been using the `dspace stats-util --optimize` tool, it is no longer recommended and you should stop.
  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.
16. **Install the new User Interface (see below)**

Upgrading the Frontend (User Interface)

1. **Install the new User Interface per the [Installing DSpace] guide.** The JSPUI and XMLUI are no longer supported and cannot work with the DSpace 7 backend. You will need to install the new (Angular.io) User Interface.
   a. JSPUI or XMLUI based themes cannot be migrated. That said, since the new Angular UI also uses Bootstrap, you may be able to borrow some basic CSS from your old themes. But any HTML-level changes will need to be reimplemented in the new UI.

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 updating the Metadata Registries

The database migration (./dspace database migratte) should *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
```