

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 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 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 [Upgrading the Backend \(Server API\)](#)
 - 2.1 [Backup your DSpace Backend](#)
 - 2.2 [Update Backend Prerequisite Software](#)
 - 2.3 [Upgrading the Backend Steps](#)
- 3 [Upgrading the Frontend \(User Interface\)](#)
- 4 [Troubleshooting Upgrade Issues](#)
 - 4.1 [See all "Common Installation Issues"](#)
 - 4.2 [Database migrate errors: "Migration V5.7_2017.04.11__DS-3563_Index_metadatavalue_resource_type_id_column.sql failed" or "Migration V5.7_2017.05.05__DS-3431.sql failed"](#)
 - 4.3 [Database migrate errors because of custom database tables/columns](#)
 - 4.4 [Running "Ignored" Flyway Migrations](#)
 - 4.5 [Manually updating the Metadata Registries](#)

Release Notes / Significant Changes

DSpace 7.6.1 only provided improvements and bug fixes. See the [Release Notes](#) for all the details. A few fixes to be aware of when you upgrade:

- Git is no longer required to build DSpace. You can remove Git from your system unless you require Git for your own local version control.
- Hierarchical browse indexes for controlled vocabularies can now be disabled via a backend configuration. See [Configuration Reference#HierarchicalBrowseIndexes](#) for details.

DSpace 7.6 added some new features and many improvements and bug fixes. See the [Release Notes](#) for all the details. These major features are most likely to impact your upgrade:

- **Oracle support has been removed** as was [previously announced in March 2022](#) on our mailing lists.
- **Item counts can now be displayed for all Communities/Collections** *similar to version 6.x*.
- **New default Privacy Statement and End User Agreement.** The new default text of these policies can be found by visiting the links in the footer of our [demo site](#).

DSpace 7.5 added some new features and many improvements and bug fixes. See the [Release Notes](#) for all the details. These major features are most likely to impact your upgrade:

- **Subscribe to email updates from a Community or Collection** - This REQUIRES enabling a new "subscription-send" [Scheduled Tasks via Cron](#).
- **New "dspace database skip" command** - Useful tool for some sites upgrading from 6.x or below *if they encounter this [common migration error](#)*
- **User interface now provides page caching for better performance** - Sites which have encountered slower UI performance may wish to use the new settings in [Cache Settings - Server Side Rendering\(SSR\)](#)
- **User interface now requires Node.js 16 or 18**. While it may work for other versions, it's best to no longer use Node.js 14, as it goes EOL in April 2023

DSpace 7.4 added some new features and many improvements and bug fixes. See the [Release Notes](#) for all the details. These major features are most likely to impact your upgrade:

- **Recent Submissions are now listed on the homepage.** (See ["recentSubmissions" options](#))
- **Thumbnails are now displayed** in all search/browse screens. (See ["showThumbnails" option](#))

DSpace 7.3 added many new features and improvements. See the [Release Notes](#) for all the details. These major features are most likely to impact your upgrade:

- **Oracle Database support has been deprecated.** It will be removed in mid-2023 (likely in the 7.6 release, which is tentative for June 2023). All sites should plan to migrate to PostgreSQL. See <https://github.com/DSpace/DSpace/issues/8214>
- **ORCID Authentication and synchronization to a DSpace Researcher Profile** now exists (requires first enabling [Configurable Entities](#)). See [OR CID Integration](#) and [Researcher Profiles](#)
- **Admin "Health" menu provides basic control panel functionality (based on 6.x Control Panel).** When logged in as an Administrator, select "Health" from the side menu. You'll see a "Status" tab which provides useful information about the status of the DSpace backend, and an "Info" tab which provides an overview of backend configurations and Java information.
- **Media Filter updates to use Apache Tika.** There is now a single "Text Extractor" media filter which you should use (see updated settings in `dspace.cfg` and [Mediafilters for Transforming DSpace Content](#) for more details).

DSpace 7.2 made some major changes to the User Interface build process, including a new configuration file format. See the [Release Notes](#) for all the details. These major features are most likely to impact your upgrade:

- **New Configuration for User Interface to support Runtime Configuration:** *In the User Interface, the "environment.*.ts" configuration files have been replaced with a new "config.*.yaml" file.* A migration script is provided which can migrate your UI configurations from the old format to the new one. More information on that migration is available in the [User Interface Configuration](#) documentation.
- **Submission Process now supports Item Embargoes / access restrictions.** It is disabled by default, but can be easily enabled by uncommenting (or adding) the "itemAccessConditions" step in your item-submission.xml on the backend. See [Submission User Interface](#) and [Emargo](#) for more details.
- **Feedback Form** now exists. It is enabled by default in the UI's footer as long as you set a "feedback.recipient" in your local.cfg on the backend.
- **OpenID Connect (OIDC) Authentication Plugin** now exists. See the [Authentication Plugins](#) page for how to enable it.
- **Improved support for custom "Browse By" configurations.** If you had previously configured custom "Browse by" types in your UI configuration file, those settings can be removed. The "Browse by" types are now read dynamically from the REST API based on configured indexes. See [User Interface Configuration](#) for more details.

DSpace 7.1 primarily added new features and bug fixes on top of 7.0. See the [Release Notes](#) for all the details. A few key changes to be aware of:

- **New Collection "Entity Type" Configuration:** If you were using [Configurable Entities](#) in 7.0, the Entity Type to Collection "mapping" has been moved to the Collection's "Edit Metadata" screen as a new "Entity Type" dropdown. This formalizes the recommended mapping between a Collection and an Entity Type (so we highly recommend each Entity Type be stored in its own Collection). See the "Configure Collections for each Entity type" section of the [Configurable Entities](#) documentation for more details.

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](#)
- **The old REST API ("rest" webapp from DSpace v4.x-6.x) is deprecated and will be removed in v8.x.** The new [REST API](#) (provided in the "server" webapp) replaces all functionality available in the older REST API. If you have tools that rely on the old REST API, you can still (optionally) build & deploy it alongside the "server" webapp via the "-PdSPACE-rest" Maven flag. See [REST API v6 \(deprecated\)](#)
- **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.
- **The Submission Form configuration has changed.** The "item-submission.xml" file has changed its structure, and the "input-forms.xml" has been replaced by a "submission-forms.xml". See [Submission User Interface](#)
- **The traditional, 3-step Workflow system has been removed in favor of the Configurable Workflow System.** For most users, you should see no effect or difference. The default setup for this Configurable Workflow System is identical to the traditional, 3-step workflow ("Approve/Reject", "Approve/Reject/Edit Metadata", "Edit Metadata")
- **The old BTE import framework in favor of Live Import Framework** (features of BTE have been ported to Live Import)
- **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 [configur](#) [ation 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.
- **A large number of old/obsolete configurations were removed.** "7.0 Configurations Removed" section in the [Release Notes](#).

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.x, 12.x or 13.x (with pgcrypto installed), OR
 - **Deprecated:** Oracle 10g or above (Oracle support is deprecated, see <https://github.com/DSpace/DSpace/issues/8214>)
- **Updated:** Tomcat 9.x
- **New:** Solr 8.x. See [Installing new Solr cores when upgrading from DSpace 6.x or 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 [Set up IP to City Database](#).

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

Upgrading the Backend Steps

Migration guide also available



If during the upgrade you are migrating your DSpace backend to a new server/machine, see [Migrating DSpace to a new server](#) guide for hints/tips.

1. **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-7.2) 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.
2. **If upgrading from 6.x or below, a few extra steps are required before you install DSpace 7.x.** *If you are upgrading from a previous version of 7.x, skip this and move along.*
 - a. **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).
 - i. **PostgreSQL databases:** PostgreSQL 9.4 or above is required and the "pgcrypto" extension must be installed.
 1. Notes on installing pgcrypto
 - a. 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".
 - b. On Windows, this extension should be provided automatically by the installer (check your "[PostgreSQL]/share/extension" folder for files starting with "pgcrypto")
 2. 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;
```

- ii. *Oracle databases: Oracle support has been deprecated in DSpace.* It will no longer be supported as of June/July 2023. See <https://github.com/DSpace/DSpace/issues/8214> for more details.
- b. **From your old version of DSpace, dump your authority and statistics Solr cores.** (Only necessary if 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 were **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.

Unfortunately, the "solr-export-statistics" script was not created until DSpace 5.x. Therefore, you will not be able to upgrade statistics from 4.x or below *unless you first upgrade to either 5.x or 6.x*. This upgrade could be done in a test environment, to allow you to export your statistics (so they can be reimported into 7.x below). But, there's unfortunately no direct way to migrate 4.x (or 3.x or 1.x.x) Solr Statistics into 7.x.

- c. **Move your old Solr cores to a safe location** in case of trouble with the upgrade procedure. If you leave them in place, you will get a mixture of old and new files that the new Solr will refuse to load.
- d. **(If upgrading from 5.x or below) Replace your old build.properties file with a local.cfg:** 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.
```

3. **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
```

4. **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.)

5. **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 a prior version of DSpace 7.x, you will need to perform the following steps.**

- i. As of DSpace 7.3, the "db.dialect" configuration has changed from "org.dspace.storage.rdbms.hibernate.postgres.DSpacePostgreSQL82Dialect" to "org.hibernate.dialect.PostgreSQL94Dialect". Therefore, MAKE SURE that your dspace.cfg or local.cfg has this setting:

```
db.dialect = org.hibernate.dialect.PostgreSQL94Dialect
```

- ii. You may wish to review the [Release Notes](#) for details about new features. There may be new configurations you may wish to tweak to enable/disable those features.
 - iii. Make sure your existing 7.x local.cfg is in the source directory (e.g. [dspace-source]/dspace/config/local.cfg). That way your existing 7.x configuration gets reinstalled alongside the new version of DSpace.
- b. If you are upgrading from DSpace 6.x or below, you 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. **Remove BTE Spring configuration:** If it exists, remove the [dspace]/config/spring/api/bte.xml Spring Configuration. This file is no longer needed as the BTE framework was removed in favor of [Live Import from external sources](#).
 - iv. **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.
 - v. **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).
 - vi. **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.
 - vii. **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://localhost:8983/solr. Solr only needs to be accessible to the DSpace backend, and should not be publicly available on the web. It can either be run on localhost or via a hostname (if run on a separate server from the backend). Also review the values of
 - 1. discovery.search.server
 - 2. oai.solr.url
 - 3. solr.authority.server
 - 4. solr-statistics.server
 - viii. **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.
- c. If you are upgrading from DSpace 5.x or below, there are a few additional configuration changes to be aware of.
- 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).

6. **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
```

7. **Upgrade your database** (*required for all upgrades*). 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. (Optional) If desired, you can optionally verify which migrations have not yet been run on your database. You can use this to double check that DSpace is recognizing your database version appropriately

```
[dspace]/bin/dspace database info

# If you are upgrading from 5.x or later, then this will list all migrations
#   which were previously run, along with any which are "PENDING" or "IGNORED"
#   that need to be run to upgrade your database.
# If you are upgrading from 4.x or earlier, this will attempt to detect which
#   version of DSpace you are upgrading from. Look for a line at the bottom
#   that says something like:
#   "Your database looks to be compatible with DSpace version ____"
```

- b. (Optional) 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).

```
# This command only works if upgrading from DSpace 7.0 or later
[dspace]/bin/dspace database update-sequences

# If upgrading from DSpace 6 or below, this script had to be run via psql from [dspace]/etc
# /postgres/update-sequences.sql
# For example:
# psql -U [database-user] -f [dspace]/etc/postgres/update-sequences.sql [database-name]

# NOTE: It is important to run the "update-sequences" script which came with the OLDER version of
# DSpace (the version you are upgrading from)! If you've misplaced # this older version of the
# script, you can download it from our codebase & run it via the "psql" command above.
# DSpace 6.x version of "update-sequences.sql": https://github.com/DSpace/DSpace/blob/dspace-6_x
# /dspace/etc/postgres/update-sequences.sql
# DSpace 5.x version of "update-sequences.sql": https://github.com/DSpace/DSpace/blob/dspace-5_x
# /dspace/etc/postgres/update-sequences.sql
```

- c. (REQUIRED) 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)

```
# If upgrading from DSpace 6.x or below
[dspace]/bin/dspace database migrate ignored

# If upgrading from an earlier version of DSpace 7.x
[dspace]/bin/dspace database migrate
```

If you are upgrading from DSpace 6.x or below be sure you include the "ignored" parameter! There are database changes which were previously optional but now are mandatory (specifically [Configurable Workflow](#) database changes).

- d. *If the database upgrade process fails or throws errors*, then look at the "Troubleshooting Upgrade Issues" section below for possible tips /hints.
- 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 index-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 instabilities.

8. **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.
9. **If upgrading from a previous version of 7.x, a few extra steps may be required before starting Tomcat.**
 - a. **Update your Solr schema definition(s)**
 - i. If you are upgrading from one 7.x release to another, you will need to update your 'search' Solr schema definition with the new version (For example, in 7.1 a new "search.entitytype" field was added to this schema. In 7.6, a new "lastModified_dt" field was added to this schema.).

1. Copy the updated schema.xml to your Solr 'search' core, e.g.

```
# The destination directory may differ per OS.
# Just replace the existing 'search' core "schema.xml" with the new one.
cp [dspace]/solr/search/conf/schema.xml /var/solr/data/search/conf/
```

2. Restart Solr

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

3. Reindex your site

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

10. **If upgrading from 6.x or below, a few extra steps are required to before starting Tomcat. If you are upgrading from a previous version of 7.x, skip this and move along.**

- a. **Install new Solr cores and rebuild your indexes.** (Required when upgrading from 6.x or below. This may be done after starting Tomcat, but is required for DSpace 7.x to function properly.)
 - i. 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
```

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

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

- iii. You can check the status of Solr and your new DSpace cores by using its administrative web interface. Browse to `${solr.server}` (e.g. <http://localhost:8983/solr/>) to see if Solr is running well, then look at the cores by selecting (on the left) Core Admin or using the Core Selector drop list.

1. For example, to test that your "search" core is setup properly, try accessing the URL `${solr.server}/search/select`. It should run an empty query against the "search" core, returning an empty JSON result. If it returns an error, then that means your "search" core is missing or not installed properly.

- iv. Load authority and statistics from the CSV [dumps](#) that you made earlier in Step 2 above.

```
[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 into the "statistics" core. DSpace 7 does not support Solr shards at this time. Unfortunately, this will involve renaming all CSV export files to remove the year (e.g. rename "statistics-2012_export_2013-12_5.csv" to "statistics_export_2013-12_5.csv") and rerunning "[dspace]/bin/dspace solr-import-statistics -i statistics". More advice on this process can be found in this [dspace-tech mailing list thread](#).

- v. 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](#)

- vi. 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.

- b. **Update Handle Server Configuration.** (Required when 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"
```

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

- c. **(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.

- d. **Check your cron / Task Scheduler jobs.** In recent versions of DSpace, some of the scripts names have changed.

- 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.
- 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.

11. **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.

12. **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.

- a. If you wish to skip automatic reindexing, please see the Note above under the "Upgrade your Database" step.

- b. When upgrading from 7.0/7.1/7.2 to 7.3, it is **REQUIRED** to reindex your content. If reindexing does not occur automatically, or you disabled it, then run `./dspace index-discovery -b` to reindex your site.

```
# Manually reindex all content  
./dspace index-discovery -b
```

13. **Review / Update your scheduled tasks (e.g. cron jobs).** For all features of DSpace to work properly, there are some scheduled tasks you MUST setup to run on a regular basis. Some examples are tasks that help create thumbnails (for images), do full-text indexing (of textual content) and send out subscription emails. See the [Scheduled Tasks via Cron](#) for more details.

- a. When upgrading to 7.5 (or later), you will want to make sure the new "subscription-send" task is added to your existing scheduled tasks (in cron or similar). This new task is in charge of sending [Email Subscriptions](#) for any users who have subscribed to updates. (NOTE: "subscription-send" replaces the older "sub-daily" task from 6.x or below). See the [Scheduled Tasks via Cron](#) for more details.

14. Install or Upgrade the new User Interface (see below)

Upgrading the Frontend (User Interface)

1. If upgrading from 6.x or below, 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.
2. If upgrading from a prior version of 7.x, upgrading requires installing the latest version of the User Interface code
 - a. **Upgrade Node.js if necessary**
 - i. If upgrading to 7.5, we highly recommend using either Node.js 16 or 18 as Node.js 14 will enter EOL (end of life) as of 30 April 2023.
 - b. **Download the latest dspace-angular 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-7.2) or branch.
 - i. If you've cloned or copied this code into your own GitHub or GitLab repository, you may wish to simply pull the latest tagged code into your codebase using Git. That will allow you to more easily address any "code conflicts" between your local changes and the new version of DSpace (if any are found).
 - ii. NOTE: For the rest of these instructions, we'll refer to the source code location as [dspace-angular].
 - c. **Install any updated local dependencies** using Yarn in the [dspace-angular] source code directory:

```
# change directory to our repo
cd [dspace-angular]

# install/update the local dependencies
yarn install
```

- d. **Build the latest User Interface code for Production:** This rebuilds the latest code into the [dspace-angular]/dist directory

```
yarn build:prod
```

- e. **If upgrading from 7.0 or 7.1, read the updated [Installation documentation](#).** As of 7.2, we now recommend deploying the compiled User Interface (in [dspace-angular]/dist) to a different directory (which we refer to as [dspace-ui-deploy]) in order to keep your running UI separate from the source code. While it's still possible to run the UI using "yarn start" or "yarn run serve:ssr" (both of which use [dspace-angular]/dist), that older approach will mean that your site goes down / becomes unavailable anytime you rebuild (yarn build:prod). To solve this issue:
 - i. Create a separate [dspace-ui-deploy] location as described in the Installation documentation.
 - ii. Copy the [dspace-angular]/dist folder to that location, as described in the Installation documentation.
 - iii. Update your PM2 configuration or local startup scripts to use Node.js instead of Yarn. Again, see the Installation documentation.
- f. **If upgrading from 7.0 or 7.1, migrate your UI Configurations to YAML.** In 7.2, the format of the UI configuration file changed from Typescript to YAML to support runtime configuration. This means that the older ./src/environment/environment.*.ts configuration files have all been replaced by corresponding ./config/config.*.yaml configuration files (e.g. environment.prod.ts config.prod.yaml).
 - i. Either, manually move your "environment.prod.ts" configurations into a new "./config/config.prod.yaml" file, using the "./config/config.example.yaml" as a guide, along with the [User Interface Configuration](#) documentation.
 - ii. OR, you can migrate your configurations using the provided "yarn env:yaml" migration script. For detailed instructions, see the [Migrate environment file to YAML](#) second of the [User Interface Configuration](#) documentation.
- g. **(Optional) Review Configuration changes** to see if you wish to update any new configurations
 - i. In 7.1, we added the ability to "extend" themes in the "themes" section. See the [User Interface Configuration](#) documentation for details.
 - ii. In 7.2, themes now support optional "headTags" which can be used to customize favicons per theme (see [User Interface Customization](#)). Additionally, "browseBy > types" configurations were removed, as they are now dynamically retrieved from the REST API (see [User Interface Configuration](#)).
 - iii. In 7.3, you can optionally enable Item Access labels in the UI to display the status of an Item as a badge (e.g. "open access", "restricted", "metadata only" or "embargoed"). See [User Interface Configuration](#)
 - iv. In 7.4, many new user interface configurations were added. See [Release Notes](#) and [User Interface Configuration](#) for details.
 - v. In 7.5, a new "Server Side Rendering" page caching option was added which can drastically speed up the initial response of your site. Other new settings also were added. See [Release Notes](#) and [User Interface Configuration](#) for details.
- h. **Update your theme (if necessary)**, if you've created a custom theme in "src/themes" (or modified the existing "custom" or "dspace" themes in that location). Pay close attention to the following..
 - i. In 7.3, a new "eager-theme.module.ts" and "lazy-theme.module.ts" has been added to both the "custom" and "dspace" themes to improve performance. Make sure to copy those to your custom theme. Additionally, this new "eager-theme.module.ts" for your theme MUST be imported/enabled in "src/themes/eager-themes.module.ts". For example, for a local theme under "src/theme/my-theme":

src/themes/eager-themes.module.ts

```
import { EagerThemeModule as MyThemeEagerThemeModule } from './my-theme/eager-theme.module';
...
@NgModule({
  imports: [
```

```

    MyThemeEagerThemeModule,
  ],
})

```

- ii. Additional minor changes may have been made. It's usually best to look for changes to whichever theme you started from. If you started your theme from the "custom" theme, look for any new changes made under `"/src/themes/custom"`. If you started your theme from the "dspace" theme, look for any new changes made under `"/src/themes/dspace"`.

1. Using a tool like "git diff" from the commandline is often an easy way to see changes that occurred only in that directory.

```

# Example which will show all the changes to "src/themes/dspace" (and all subfolders)
# between dspace-7.4 (tag) and dspace-7.5 (tag)
git diff dspace-7.4 dspace-7.5 -- src/themes/dspace/

```

- iii. For the "custom" theme, the largest changes are often:

1. New themeable components (subdirectories) may be added under `"src/themes/custom/app"`, allowing you the ability to now change the look & feel of those components.
2. The `"src/themes/custom/theme.module.ts"` file will likely have minor updates. This file registers any new themeable components (in the `"const DECLARATIONS"` section), and also registers new Modules, i.e. new UI features, (in the `"@NgModule"` `"imports"` section). Make sure those sections are updated in your copy of this file!
3. Sometimes, new styles may be added in the `"styles"` folder, or new imports to `"styles/theme.scss"`
4. If you have locally customized the styles or look & feel of any component, you should also verify that the component itself (in `src/app`) hasn't had updates.

- iv. For the "dspace" theme, the largest changes are often:

1. Existing customized components (subdirectories) under `"src/themes/dspace/app/"` may have minor updates, if improvements were made to that component.
2. The `"src/themes/custom/theme.module.ts"` file will likely have minor updates. This file registers any new themeable components (in the `"const DECLARATIONS"` section), and also registers new Modules, i.e. new UI features, (in the `"@NgModule"` `"imports"` section). Make sure those sections are updated in your copy of this file!
3. Sometimes, new styles may be added in the `"styles"` folder, or new imports to `"styles/theme.scss"`
4. If you have locally customized the styles or look & feel of any additional component, you should also verify that the component itself (in `src/app`) hasn't had updates.

i. **Restart the User Interface.**

- i. If you are using PM2 as described in the [Installing DSpace](#) instructions, you'd stop it and then start it back up as follows

```

# Stop the running UI
pm2 stop dspace-ui.json

# If you had to update your PM2 configs, you may need to delete your old configuration from PM2
# pm2 delete dspace-ui.json

# Start it back up
pm2 start dspace-ui.json

```

- ii. If you are using a different approach, you simply need to *stop the running UI*, and re-run:

```

# First stop the running UI process by killing it (or similar)

# You MUST restart the UI from within the deployment directory
# See Installation Instructions for more info on this directory
cd [dspace-ui-deploy]

# Then restart the UI via Node.js
node ./dist/server/main.js

```

j. **Verify the UI and REST API are both working properly.**

- i. If you hit errors, see the "Troubleshooting Upgrade Issues" section below. Additionally, check the "Common Installation Issues" section of the [Installing DSpace](#) documentation for other common misconfiguration or setup issues.

Troubleshooting Upgrade Issues

See all "Common Installation Issues"

At times the upgrade process may involve configuration changes which could result in one of our "Common Installation Issue" error messages. So, make sure to check that section of the [Installing DSpace](#) documentation, especially if you find your UI is no longer connecting to your REST API.

Database migrate errors: "Migration V5.7_2017.04.11_DS-3563_Index_metadatavalue_resource_type_id_column.sql failed" or "Migration V5.7_2017.05.05_DS-3431.sql failed"

If you are upgrading to DSpace 7.x and receive *either* of the two following errors after running `./dspace database migrate ignored`:

```
Migration V5.7_2017.04.11_DS-3563_Index_metadatavalue_resource_type_id_column.sql failed
```

[or]

```
Migration V5.7_2017.05.05_DS-3431.sql failed
```

This means your database never ran those older migrations during a past upgrade from 5.x6.x (or similar).

Luckily, though, these migrations are both obsolete in DSpace 7.x (and later). This means *you can skip these migration safely*.

As of DSpace 7.5, a new `./dspace database skip` command is provided to easily skip one (or both) of these failing migrations as follows:

```
# If you need to skip "V5.7_2017.04.11_DS-3563_Index_metadatavalue_resource_type_id_column.sql", run this:
./dspace database skip "5.7.2017.04.11"
```

```
# If you need to skip "V5.7_2017.05.05_DS-3431.sql", run this:
./dspace database skip "5.7.2017.05.05"
```

For more information on the `./dspace database skip` command see [Database Utilities](#).

Database migrate errors because of custom database tables/columns

If you have modified your database structure directly (or installed custom plugins which do so), then it is possible a `./dspace database migrate` will fail if it encounters an unexpected database structure. 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:

1. Revert back to your current DSpace database
2. 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/`
3. Then, re-run the migration process from that point forward (i.e. re-run `./dspace database migrate`)

Running "Ignored" Flyway Migrations

During some upgrades, a Flyway database migration will be "ignored." One known instance of this is documented in <https://github.com/DSpace/DSpace/issues/6762>. If you are upgrading from DSpace 5.x to a later version of DSpace, the migration put in place to address <https://github.com/DSpace/DSpace/pull/1128> 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
```

Manually updating the Metadata Registries

The database migration (`./dspace database migrate`) 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:

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