Console V2 Migration Tool - 1.0

> ⓘ **Summary**: Information about the **Console V2 Migration Tool**, a tool to migrate applications managed in Console V1 to Console V2 (Enterprise mode only).

Extension ID

com.castsoftware.aip.console.v2.migration

What's new?

See **Console V2 Migration Tool - Release Notes - 1.0** for more information.

What is the Console V2 Migration Tool?

The **Console V2 Migration Tool** is a command line based tool (`Console-V2-Migration.exe`) that will migrate **ALL applications** managed in **Console V1** into **Console V2** (it is NOT possible to migrate only specific Applications). This migration includes:

- The migration of the Console V1 flat file **H2 database** to the Console V2 "aip_node" schema
- Backup of the **Console V1 data** which will be restored to the V2 environment

Things to know

- CAST recommends that you start with an "empty" Console V2 (i.e. no applications already present) - this is because part of the migration process will transfer global settings/system settings/domains/user rights etc. from Console V1 to Console V2 and will **OVERWRITE** any equivalent settings in Console V2. However, if you have applications already present in Console V2 you can still proceed with a migration (these existing applications in Console V2 will be unaffected), but any global settings/system settings/domains/user rights etc. you have changed in Console V2 may be overwritten by settings from Console V1.
- The migration process does NOT transfer:
  - any user authentication settings. You should set up authentication in Console V2 via Keycloak as described in **Configure authentication and roles using Keycloak - v. 2.x** before you start the migration.
  - any CAST Storage Service/PostgreSQL settings - all CAST Storage Service/PostgreSQL instances configured in Console V1 must be configured in Console V2 (for both Analysis/Dashboard schema and Measurement schema requirements). See **Administration Center - Settings - CSS and Measurement settings**.
  - any logs from Console V1 - therefore when examining application analysis logs using the **Application - Logs** panel the details will not be available.
- The tool must be run on:
  - The Console V1 front-end host server
  - ALL Console V1 Nodes
- The migration process will leave the Console V1 installation untouched.

Prerequisites

Console V1

✅	**Console release**	1.27.0-funcrel
✅	**Node/AIP Core**	All Nodes MUST be running the SAME release of **AIP Core**.
✅	**Deployment/Delivery folders**	All Nodes must use "common" **Deployment** and **Delivery** folders (i.e. a shared network resource).
✅	**Stop all services**	All Console V1 services (front-end and all Nodes) must be stopped before the migration is started.
✅	**Path to tool**	The `Console-V2-Migration.exe` tool must be run from a path that **does not contain white spaces**, e.g.: `C:\CAST\Console-V2-Migration.exe`

Console V2

✅	**Console release**	2.2.0-funcrel - **Enterprise** mode only (Docker or Java JAR installer -see **AIP Console - front-end installation**)
✅	**Node/AIP Core**	All Nodes MUST be running the SAME release of **AIP Core** as the Console V1 Nodes.
✅	**Deployment /Delivery folders**	All Nodes must use the SAME common **Deployment** and **Delivery** folders as used for Console V1.
✅	**CAST Storage Service /PostgreSQL**	All **CAST Storage Service/PostgreSQL** instances configured in Console V1 must be configured in Console V2 (for both Analysis/Dashboard schema and Measurement schema requirements).). See **Administration Center - Settings - CSS and Measurement settings**.
✅	**Measurement schema**	The **Measurement schema** MUST use the SAME name as defined in Console V1. See **Administration Center - Settings - CSS and Measurement settings**.
✅	**Source folder location (optional)**	If you are uploading source code from a **source folder location** in Console V1, you MUST configure the SAME source folder location in Console V2.
✅	**CAST Imaging settings (optional)**	If you are publishing data to **CAST Imaging**, ensure you configure the SAME settings in Console V2 as in Console V1. See **Administration Center - Settings - Imaging Settings**.
✅	**Schema backup (optional)**	Optionally, you may wish to perform a backup of the Console V2 internal schema called "**aip_node**" using PostgreSQL tools such as psql or PGAdmin: • In a **Docker** installation, this schema will be located on the CAST Storage Service/PostgreSQL instance bundled as a Docker image • In a **Java JAR installation**, this schema will be located on the CAST Storage Service/PostgreSQL instance configured during the installation process for the Service Registry (see **2.x - Enterprise mode - Installation of AIP Console front-end via Java JAR installers**)

Console-V2-Migration.exe syntax

```
Console-V2-Migration.exe

--h2dbfile          H2DBFILE      V1 - Path to H2 database without file extension (default: None)
--h2dbtype          H2DBTYPE      V1 - Node or Console (default: None)
--casthomedir       CASTHOMEDIR   V1 - AIP Core installation path (default: None)
--log               LOG           Output log file path (default: None)
--workingdir        WORKINGDIR    Backup processing path (default: None)
--schema            SCHEMA        V2 - "aip_node" schema name (default: None)
--user              USER          V2 - Username used to connect to target V2 CSS/PostgreSQL instance (default:
operator)
--host              HOST          V2 - Host for Postgres server
--port              PORT          V2 - Port for Postgres server (default: 2285)
--database          DATABASE      V2 - Database name of CSS server (default: postgres)
--aipnodedir        AIPNODEDIR    V1 - Provide the path to the aip-node-app.properties (default: None)
--outputdir         OUTPUTDIR     Output path (default: None)
--zipfilename       ZIPFILENAME   Output Zip filename - .zip extension not required (default: None)
--v2path            V2PATH        V2 - Specify the path of the "common_data" shared folder (default: None)
--process           PROCESS       Specify the task (ALL or MIGRATION or CUSTOMSCRIPT or ZIP or MNGTSCRIPT)
(default: ALL)

Note about paths:
- CAST highly recommends using backslashes where paths to folders/files must be specified.
- Quote marks (") around paths are not required, except where the path contains white space. However, using
quote marks for all paths is also accepted by the tool.
```

> ⓘ The tool will prompt for the password of the user specified in the **--user** command (if this command is omitted, "**operator**" is assumed) and will accept only a **plain text password**. If you want to avoid the prompt, you can pass the following command in the command prompt before you run the Console-V2-Migration.exe tool, where <PASSWORD> is the plain text password for the user specified in the --user command:
>
> ```
> SET CSSPASSWORD=<PASSWORD>
> ```

Migration process

The migration process contains distinct steps:

- Run the tool on all Nodes
- Merge and copy output ZIP from all Nodes to Console V2 environment
- Run the tool on the Console front-end host server
- Manually copy Architecture Studio data to Console V2 environment
- Access Console V2 and verify that all expected Applications/data are present

Step 1 - Run the tool on all Nodes

Run the **Console-V2-Migration.exe** tool on **ALL Nodes**, for example:

```
Console-V2-Migration.exe --h2dbfile "C:/ProgramData/CAST/AipConsole/AipNode/db/aip_node_db" --h2dbtype node --
casthomedir "C:/Program Files/CAST/8.3" --log "C:/work/temp/mig/mig-node1.log" --workingdir "C:/work/temp/mig
/workingdir" --schema aip_node --host v2_css_host --aipnodedir "C:/ProgramData/CAST/AipConsole/AipNode" --
outputdir "C:/work/temp/mig/out" --zipfilename node1_backup --v2path "S:/common-data"
```

When the tool completes, the following occurs:

- The tool will export the data from the **Console V1 H2 database** and insert the data to the "**aip_node**" schema in the **Console V2 CAST Storage Service/PostgreSQL** instance defined by the --host argument. Check the "application" table in "aip_node" schema to ensure that all your V1 applications are listed:

Click to enlarge
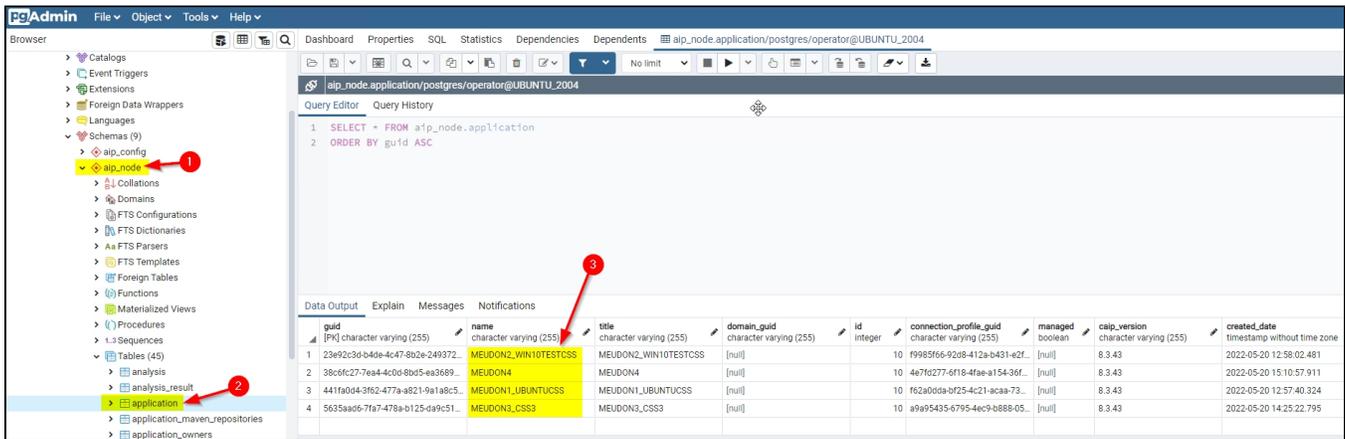
- A **ZIP file** will be generated on each Node containing the Node data. This ZIP file will be located in the path defined by the `--outputdir` argument. You should check that this file exists. Its structure will look like this:
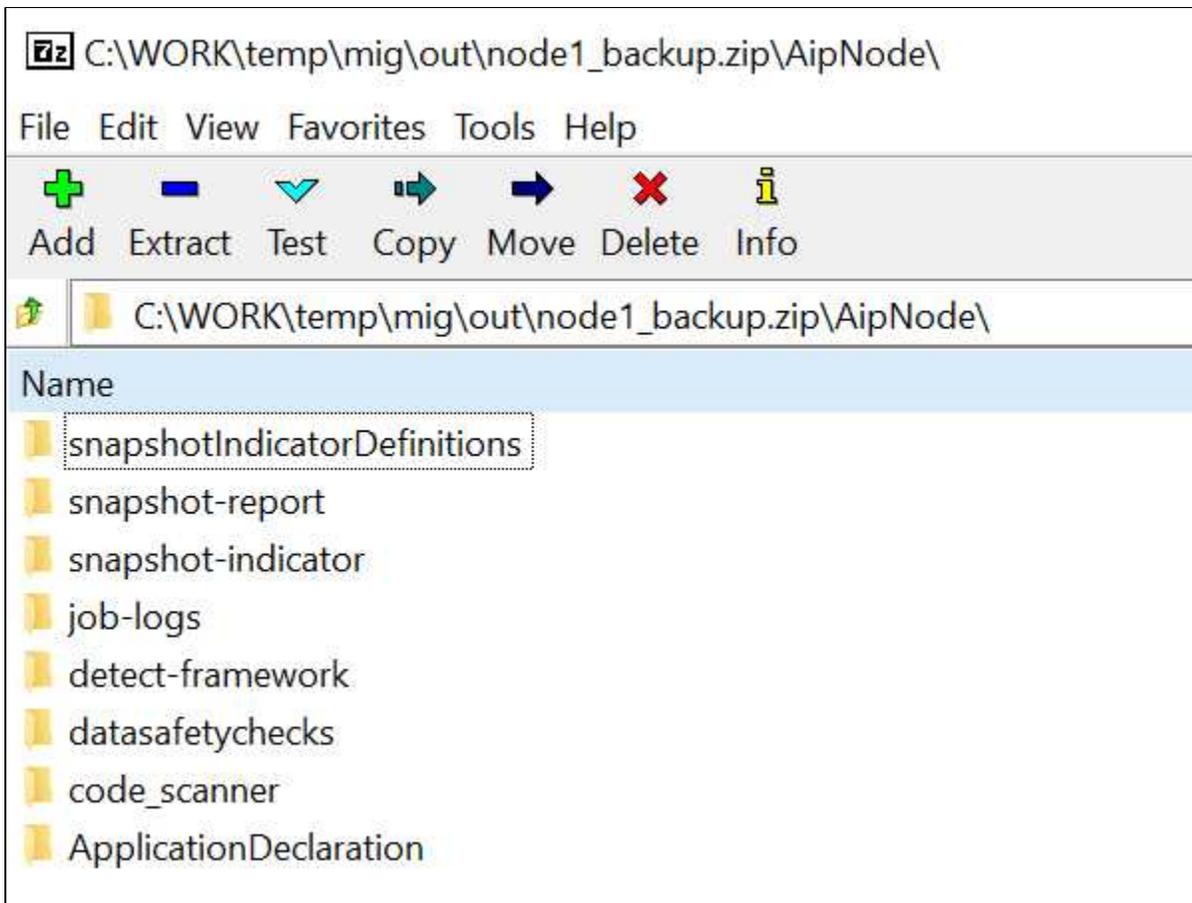
Step 2 - Copy ZIP data from all Nodes to Console V2 environment

This step involves copying the content of all generated ZIP files over to the Console V2 environment into the "**common_data**" shared folder:

- If you have **one single Node** you can simply unpack the ZIP generated on this Node and copy **all the folders** under the parent folder "**AipNode**" over to the "**common_data**" shared folder. All content is unique, therefore no existing content in "**common_data**" will be overwritten.
- If you have **multiple Nodes** and therefore **multiple ZIP files**, you will need to:
    - unpack **each ZIP file** on each Node
    - manually copy the **content** of each folder under the parent folder "**AipNode**" over to the equivalent named folder under the "**common_da ta**" shared folder. **You must NOT copy the folders themselves.**

Step 3 - Perform the migration on the Console front-end host server

This step will migrate Console data such as domains, user rights (but not users or authentication settings), global settings, authorizations, global config files etc. from the Console V1 H2 flat file database over to the the Console V2 "aip_node" schema:

Microsoft Windows host

Run the **Console-V2-Migration.exe** tool on **the front-end Console host server**. For example:

```
Console-V2-Migration.exe --h2dbfile "C:/ProgramData/CAST/AipConsole/AipConsole/db/hellodb" --h2dbtype console --
casthomedir "C:/Program Files/CAST/8.3" --log "C:/work/temp/mig/mig-console.log" --workingdir "C:/work/temp/mig
/workingdir" --schema aip_node --host v2_css_host --v2path "S:/common-data"
```

Linux host

If your Console V1 is hosted on a **Linux server**, then the Console-V2-Migration.exe tool cannot be run on this host. Instead, the steps are slightly different:

- First copy the Console V1 H2 flat file database (located in **$HOME\CAST\AipConsole\AipConsole\db**) over to a Microsoft Windows host on which AIP Core is running - CAST highly recommends using an existing V1 Node host server since this will have the correct release of AIP Core installed on it.
- Finally, run the **Console-V2-Migration.exe** tool on an existing Node, specifying the H2 flat file database you copied from the Console host server in the `--h2dbfile` argument, for example:

```
Console-V2-Migration.exe --h2dbfile "C:/work/temp/mig/db/hellodb" --h2dbtype console --casthomedir "C:/Program
Files/CAST/8.3" --log "C:/work/temp/mig/mig-console.log" --workingdir "C:/work/temp/mig/workingdir" --schema
aip_node --host v2_css_host --v2path "S:/common-data"
```

Step 4 - Manually copy Architecture Studio data to Console V2 environment

The final step is to manually copy the content of the following folder (not the folder itself)

```
Microsoft Windows
C:\ProgramData\CAST\AipConsole\AipConsole\upload\architecture

Linux
$HOME\ProgramData\AipConsole\upload\architecture
```

to:

```
\\share\aip-node-data\common-data\upload\architecture
```

Step 5 - Verify process

The final step is to access Console V2 and perform the following verification checks:

At system level

- Check that all the applications have been imported into Console V2 using the **Administration Center**
- Check the status of your application using the **AIP Console - Application Management** panel - it should match the status in Console V1
- Check in the **Administration Center** that the applications are grouped in the correct domains
- Check that all the options defined in **Administration Center - Global Configurations** are correct
- Check that all the options defined in **Administration Center - System Settings** are correct

At Application level

Take a sample selection of applications and check that the configuration of these applications is correct, for example:

1. In the **Application Overview** panel, check that all the sections in this panel are as they were in Console V1
2. In the **Application - Extensions** panel, check that the same extensions are defined as in Console V1
3. In the **Application - Config** panel, check that you have the same technologies and the same configuration in the Analysis Units as in Console V1
4. Etc..
5. The **module** definition should be identical to Console V1
6. The same for **Architecture** and the other options
7. In the **Application - Config - Advanced** section, check that the **DLM rules** are correctly set (pointing to the DLM rules in the Console V2 common data folder)