

Deploying multiple 2.x ZIPs or JARs on the same server

- [Introduction](#)
- [Required changes](#)
 - [Step 1 - Change the port on which the Dashboard is running - Linux and Microsoft Windows](#)
 - [Step 2 - Change Microsoft Windows service name - Microsoft Windows only](#)
 - [Modify the dashboard-service-install.bat file](#)
 - [Rename the .exe files to match SERVICE_NAME](#)
 - [Run the dashboard-service-install.bat](#)



This information is only valid for **CAST Dashboard 2.x** when using a **ZIP/JAR** deployment (i.e. without Apache Tomcat).

Introduction

The following situations require some small changes to the Dashboard deployment process so that each Dashboard deployment does not conflict with another:

- when multiple standalone 2.x Dashboards need to be run from the same server
- when embedded AIP Console Dashboards (from v. 1.25) need to be run on the same machine as standalone 2.x ZIP/JAR dashboards

Required changes

Step 1 - Change the port on which the Dashboard is running - Linux and Microsoft Windows

By default the ZIP/JAR Dashboards are preconfigured to run on the following ports to accept incoming connections:

Standalone dashboards	8080
Embedded dashboards	8087

If you want to run multiple ZIPs / JARs on the same server, you must ensure that each Dashboard uses a **unique port**. You can change the port in the following file:

```
<install_folder>\configurations\application.properties
```

Locate the following lines in the file:

```
Standalone dashboards

# Configure server port. This is necessary only for the .jar mode/version
# server.port=8080

Embedded dashboards

# Configure server port. This is necessary only for the .jar mode/version
server.port=8087
```

Uncomment the `server.port` line (if commented) and modify the port number to make it unique and save the file:

```
server.port=8090
```

Restart the application to ensure the changes are taken into account.

i If you change the port number for your **embedded dashboards**, you will need to update the configuration in AIP Console, see [Administration Center - Settings - Dashboard Integration](#):

Dashboard Integration

Url
http://my_server:8087

Username
admin

Integration Key

SAVE AND SYNCHRONIZE

Step 2 - Change Microsoft Windows service name - Microsoft Windows only

If you are deploying the dashboards on Microsoft Windows and you intend to run them via a Microsoft Windows service, you must make some changes **BEFORE** you install the services or run the wizard installer in 2.5 so that each service is unique. For each service that you want to install, perform the following actions:

Modify the dashboard-service-install.bat file

Open the following file with a text editor:

```
<install_folder>\dashboard-service-install.bat
```

Change the following lines to something unique, for example, add a digit for each dashboard:

```
set SERVICE_NAME=dashboard-service  
set DISPLAY_NAME="CAST Dashboard Service"  
set DESCRIPTION="CAST Dashboard Service"
```

E.g.:

```
set SERVICE_NAME=dashboard-service-1  
set DISPLAY_NAME="CAST Dashboard Service 1"  
set DESCRIPTION="CAST Dashboard Service 1"
```

Save the file.

Rename the .exe files to match SERVICE_NAME

Now rename the following .exe files to match EXACTLY the name you chose for **SERVICE_NAME** in **dashboard-service-install.bat**:

```
<install_folder>\dashboard-service.exe  
<install_folder>\dashboard-servicew.exe  
<install_folder>\amd64\dashboard-service.exe
```

E.g. where SERVICE_NAME=dashboard-service-1:

```
<install_folder>\dashboard-service-1.exe  
<install_folder>\dashboard-service-1.exe  
<install_folder>\amd64\dashboard-service-1.exe
```

Run the dashboard-service-install.bat

The final step is to run the following file to install the Microsoft Windows Service for your dashboard:

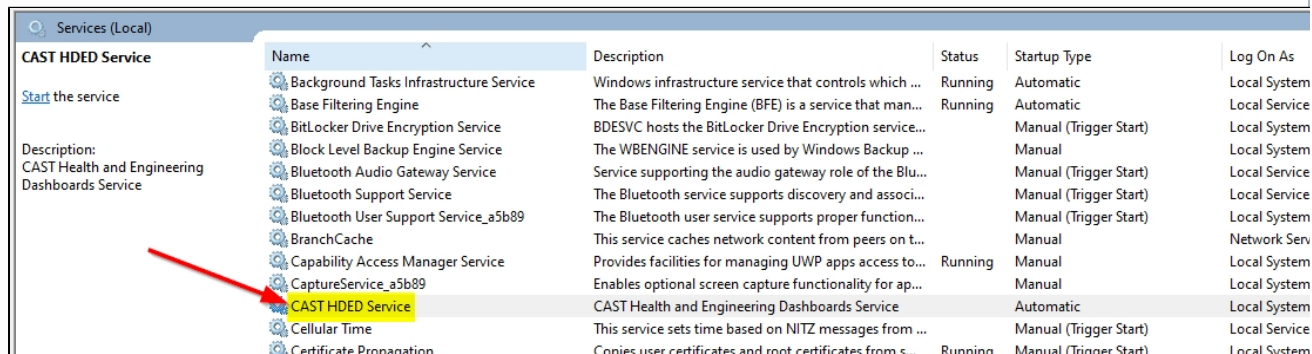
```
<install_folder>\dashboard-service-install.bat
```

Ensure that the service has been installed and run it to check that the dashboard launches correctly.

Deleting the services

Note that if you have installed Microsoft Windows services using the method described above, it will not be possible to use the **dashboard-service-uninstall.bat** provided by CAST to remove the services if they are no longer required. In this scenario, you should instead use the **sc.exe** utility (provided with Microsoft Windows) from a CMD window (the CMD window must be run with **elevated permissions** (run as administrator)), for example:

First, identify the **display name** of the service you want to remove in the services control panel. In the example below this is "**CAST HDED Service**":



Name	Description	Status	Startup Type	Log On As
CAST HDED Service				
Background Tasks Infrastructure Service	Windows infrastructure service that controls which ...	Running	Automatic	Local System
Base Filtering Engine	The Base Filtering Engine (BFE) is a service that man...	Running	Automatic	Local Service
BitLocker Drive Encryption Service	BDESVC hosts the BitLocker Drive Encryption service...		Manual (Trigger Start)	Local System
Block Level Backup Engine Service	The WBENGINE service is used by Windows Backup ...		Manual	Local System
Bluetooth Audio Gateway Service	Service supporting the audio gateway role of the Blu...		Manual (Trigger Start)	Local Service
Bluetooth Support Service	The Bluetooth service supports discovery and associ...		Manual (Trigger Start)	Local Service
Bluetooth User Support Service_a5b89	The Bluetooth user service supports proper function...		Manual (Trigger Start)	Local System
BranchCache	This service caches network content from peers on t...		Manual	Network Serv
Capability Access Manager Service	Provides facilities for managing UWP apps access to...	Running	Manual	Local System
CaptureService_a5b89	Enables optional screen capture functionality for ap...		Manual	Local System
CAST HDED Service	CAST Health and Engineering Dashboards Service		Automatic	Local System
Cellular Time	This service sets time based on NITZ messages from ...		Manual (Trigger Start)	Local Service
Certificate Propagation	Copies user certificates and root certificates from s...	Running	Manual (Trigger Start)	Local System

Now run the following command to identify the service's **key name** (this is not always the same as the display name), where DisplayName = the name displayed in the services list:

```
sc.exe GetKeyName "DisplayName"
```

E.g.:

```
sc.exe GetKeyName "CAST HDED Service"
```

In our example, the key name is **hded-service**:

```
C:\Users\James>sc.exe GetKeyName "CAST HDED Service"  
[SC] GetServiceKeyName SUCCESS  
Name = hded-service
```

Now run the following command to delete the service:

```
sc.exe delete "KeyName"
```

E.g.:

```
sc.exe delete "hded-service"
```

```
C:\WINDOWS\system32>sc.exe delete "hded-service"  
[SC] DeleteService SUCCESS
```