

# Installing and configuring the CAST Application Engineering Dashboard

## On this page:

- [Introduction](#)
  - [How many CAST Application Engineering Dashboards do I need?](#)
  - [Standalone or combined WAR file?](#)
- [Requirements](#)
- [Pre-installation check list](#)
- [Installation procedure](#)
  - [Step 1: Deploy and configure the CAST-AED.war file on your application server](#)
    - [Deploy the WAR file](#)
    - [context.xml file configuration](#)
      - [Tomcat 7 - configuration for one Dashboard Service stored on CSS2](#)
      - [Tomcat 8 or 8.5 - configuration for one Dashboard Service stored on CSS2](#)
      - [Configuring access to Dashboard Services stored on Microsoft SQL Server or Oracle Server for Tomcat 7/8/8.5](#)
      - [Configuring access to multiple Dashboard Services for Tomcat 7/8/8.5](#)
    - [Connection pooling](#)
    - [Install the license key](#)
    - [Restart Tomcat](#)
    - [Test access to the CAST Application Engineering Dashboard](#)
  - [Step 2: Configure user authentication](#)
  - [Step 3: Configure data authorization](#)
- [Additional information](#)

## Target audience:

CAST AI Administrators



**Summary:** This section describes how to **install** and **configure** the **CAST Application Engineering Dashboard** - used for low level, detailed investigation of data stored in the CAST Dashboard Service generated during the analysis/snapshot generation process.

## Introduction

### How many CAST Application Engineering Dashboards do I need?

CAST highly recommends that you install **one** CAST Application Engineering Dashboard and consolidate **all your CAST Dashboard Services schemas** into this.

### Standalone or combined WAR file?

CAST provides the CAST Application Engineering Dashboard in a **WAR file** that must be deployed to a compatible Application Server (see [Supported Platforms](#) for information about Application Server support). However, CAST provides two WAR files that contain the CAST Application Engineering Dashboard, as follows:

WAR file name	Description
<b>CAST-AED.war</b>	Traditional " <b>standalone</b> " WAR file containing only the CAST Application Engineering Dashboard (and also the <a href="#">CAST Rest API interface</a> ). This document describes the installation using this WAR file.
<b>CAST-AAD-AED.war</b>	" <b>Combined</b> " WAR file containing both the CAST Application Engineering Dashboard and the CAST Application Analytics Dashboard (plus the <a href="#">CAST Rest API interface</a> ).

You are free to use **whichever WAR file you prefer** to deploy the CAST Application Engineering Dashboard (the requirements and prerequisites are identical), however if you are using **both the CAST Application Analytics and CAST Application Engineering Dashboard**, to fulfil (for example) a "risk and quality" use case, then it would make sense to take advantage of the combined **CAST-AAD-AED.war** file and deploy **both dashboards from the same WAR file**. If you do so, please note the following about using the "combined" WAR file:

- There is no need to specifically configure a URL for your CAST Application Engineering Dashboard drilldown (via the "microscope" icon) from the CAST Application Analytics Dashboard - see [2.2.3. Snapshot generation and validation](#) for more information - when using the combined "WAR" file, access to the CAST Application Engineering Dashboard from the CAST Application Analytics Dashboard is **pre-configured**.
- Security access and authentication is **common** to both Dashboards. Therefore, if you setup Active Directory authentication in the web.xml file, then this must be used to access both Dashboards - you cannot use "Static List" for one Dashboard and "Active Directory" for the other.
- Role configuration is **common** to both Dashboards. Therefore "UserA" that has been granted the "admin" role, will have the "admin" role in both Dashboards.

- Data authorization is **common** to both Dashboards. Therefore if you authorize "UserA" to view Application "B" only via the **authorizations.xml file** , then this is true for both Dashboards. Note that authorizations based only on **Tags** and **Categories** created for the CAST Application Analytics Dashboard (Tags and Categories are a feature that is not available in the CAST Application Engineering Dashboard) WILL be applied in the CAST Application Engineering Dashboard when using the combined war file.
- Group configuration is **common** to both Dashboards, regardless of the authentication mode in use. Therefore if you authorize "GroupD" to view Application "E" only, then this is true for both Dashboards.
- On login, the user will be presented with a choice of Dashboard as shown below (*Image 1*) in a "**Welcome**" page. Clicking **Access** will take the user to the custom login screen for the chosen Dashboard where a "**Return to welcome page**" URL is available if the user wishes to login to the other Dashboard (*Image 2*):

Image 1 - Welcome page

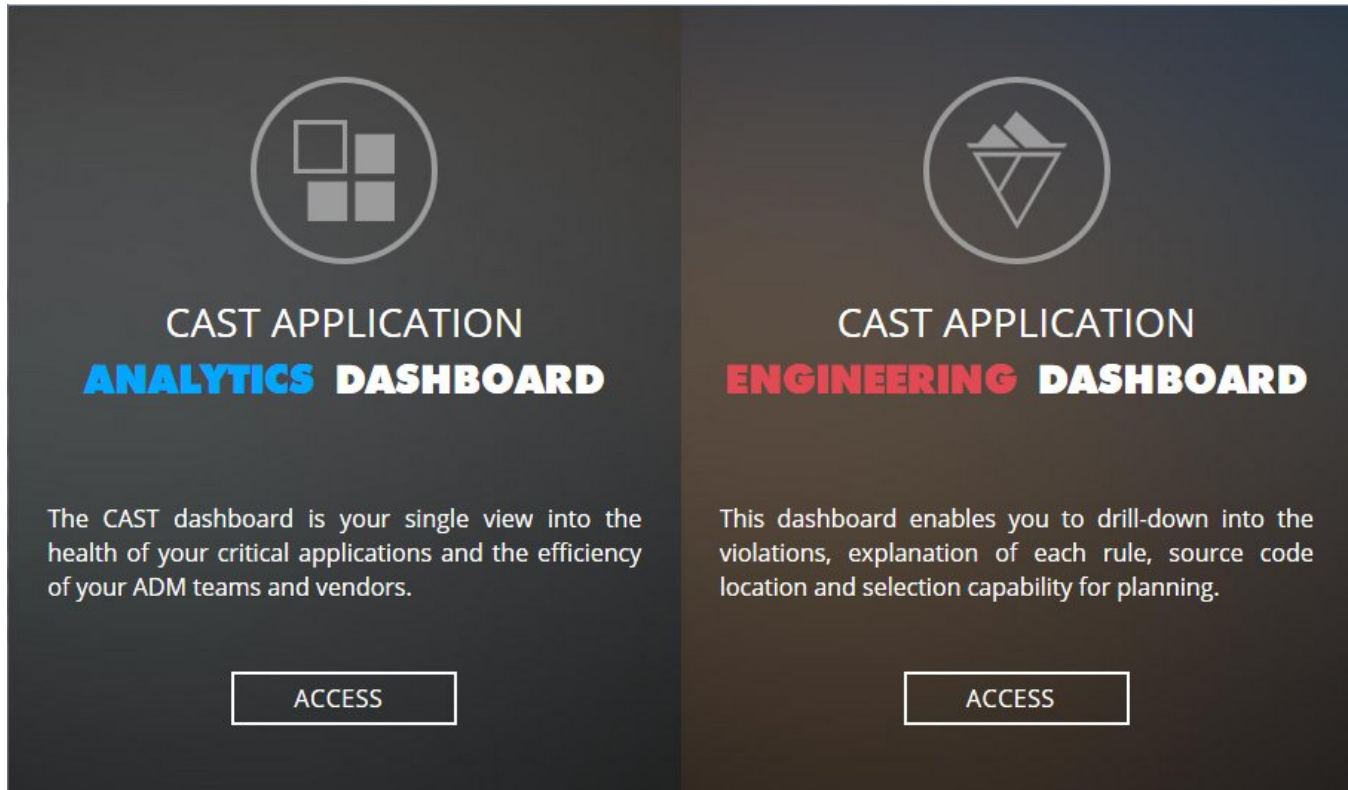
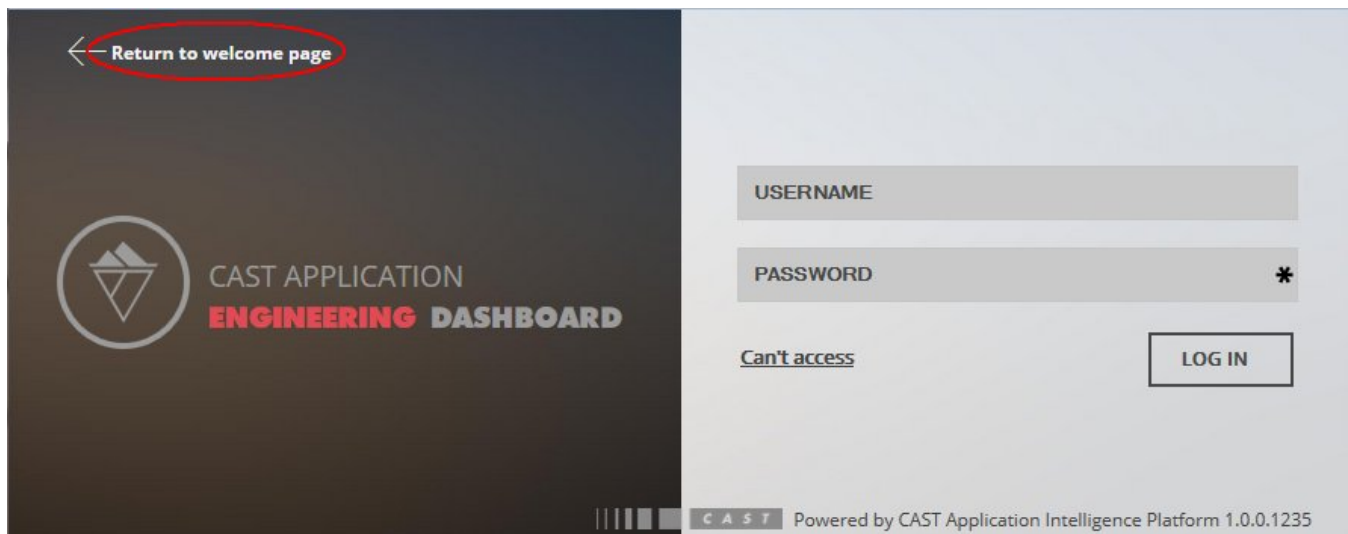
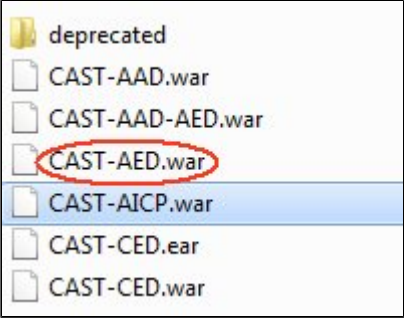


Image 2 - Return to welcome page URL



## Requirements

Installing the CAST Application Engineering Dashboard (CAST AED) requires the following to be in place before you begin:

✓	<b>Java JRE</b>	Must be installed on the machine that will host the Application Server. Please see <a href="#">Supported Platforms</a> about JRE support.
✓	<b>Application Server</b>	Must be installed on a compatible machine. Please see <a href="#">Supported Platforms</a> about Application Server support.
✓	<b>Web browser</b>	Required for accessing the CAST Application Analytics Dashboard. Please see <a href="#">Supported Platforms</a> about Web Browser compatibility.
✓	<b>Dashboard Service</b>	<p>To ensure compatibility you must only use a Dashboard Service installed with a version of CAST AIP that exactly matches the major, minor or Service Pack release number of the CAST AED WAR that you are installing.</p> <div style="border: 1px solid #ccc; padding: 10px; margin-top: 10px;"> <p><b>i</b> You can connect the CAST Application Engineering Dashboard to one or multiple <b>CAST Dashboard Service</b> schemas as follows:</p> <ul style="list-style-type: none"> <li>• Any CAST Dashboard Service installed on any <a href="#">supported RDBMS</a></li> <li>• Each Dashboard Service must contain <b>at least one snapshot</b></li> </ul> </div>
✓	<b>WAR file</b>	<p>The CAST Application Engineering Dashboard is provided as a <b>.WAR</b> file (<b>CAST-AED.war</b>) - it can be retrieved from the <b>WARS</b> folder located in the CAST installation folder:</p> 
✓	<b>License Key</b>	Access to data in the CAST Dashboard Services is governed by a <b>license key</b> - see <a href="#">CAST-AED-RESTAPI - Dashboard Service license key configuration</a> . You must have a valid license key in your possession before you begin.

## Pre-installation check list

Before beginning the installation process, please ensure that you have carried out all of the following tasks and that the following requirements have been met:

✓	Ensure you have read all <a href="#">Release Notes</a> accompanying CAST products for any last-minute information.
✓	Decide where the CAST Application Engineering Dashboard will be installed - CAST Administration workstation, separate dedicated machine or existing machine used for the CAST Application Analytics Dashboard/CAST AIC Portal.
✓	Ensure that your user login on the target machine has sufficient user privileges to install applications (see <a href="#">Appendix - CAST Administration workstation installation and hardware requirements</a> )
✓	Make sure you have the required .WAR file ready for deployment

## Installation procedure

**i** This document assumes you are using **Apache Tomcat** in a **Windows environment** to host the CAST Application Engineering Dashboard. If you are using **IBM Websphere Application Server** in a **Linux environment**, please see [Appendix - AAD - AED - RestAPI - Deployment on IBM WebSphere Application Server](#).

The installation process is divided into various steps:

### Step 1: Deploy and configure the CAST-AED.war file on your application server

#### Deploy the WAR file

Locate the **CAST-AED.war** file and move it to the web application installation location. By default on Apache Tomcat this is set to:

```
%CATALINA_HOME%\webapps
```

- If the Apache Tomcat server is running, the .WAR file will then be unpacked and deployed in a folder called "**CAST-AED**".
- Once deployed you now need to configure the **context.xml** file to tell the web application where the Dashboard Service(s) are stored. This file is located here:

```
%CATALINA_HOME%\webapps\CAST-AED\META-INF\context.xml
```

## context.xml file configuration

### Tomcat 7 - configuration for one Dashboard Service stored on CSS2

If you are using **Tomcat 7** to host the CAST Application Engine Dashboard and **CSS2** to host you CAST Dashboard Services, please follow these instructions:

- Open the **context.xml** file (as detailed above) with a Text editor.
- By default, the file will contain an **uncommented** template as follows - this is tailored for connections to **Dashboard Services** located on **CAST Storage Service 2** using **Tomcat 7**:

```
<Resource name="jdbc/domains/AED" url="jdbc:postgresql://localhost:2280/postgres"
    initConnectionSqls="SET search_path TO [Central Schema];"
    username="operator" password="CastAIP"
    auth="Container" type="javax.sql.DataSource" driverClassName="org.postgresql.Driver"
    validationQuery="select 1"
    initialSize="5" maxActive="20" maxIdle="10" maxWait="-1"/>
```

- Set the **URL** parameter to the server on which the **Dashboard Service** is located (usually you just need to alter the "localhost" to the name of the server on which the host RDBMS is located).
- Change **[Central Schema]** to the name of the **Dashboard Service schema** (see [Install CAST schemas](#)).
- Ensure the **username** and **password** parameters for your target RDBMS/CSS are correct (note that if you need to encrypt the username and password to avoid entering names in clear text, please see: [CAST-AAD-AED - Encrypt login and password for datasource and Active Directory](#))
- You should end up with a section like this:

```
<Resource name="jdbc/domains/AED" url="jdbc:postgresql://NEFYN:2280/postgres"
    initConnectionSqls="SET search_path TO V80_CENTRAL_CSS2;"
    username="operator" password="CastAIP"
    auth="Container" type="javax.sql.DataSource" driverClassName="org.postgresql.Driver"
    validationQuery="select 1"
    initialSize="5" maxActive="20" maxIdle="10" maxWait="-1"/>
```

- Following any changes you make, **save the context.xml file**.



Note that it is **critically important** to configure the **initialSize**, **maxActive**, and **maxIdle** parameters according to your consumption use case and the number of schemas you want to configure. The default values are high but could probably be decreased to limit the number of active connections to your database instance (eg. `initialSize="2" maxActive="10" maxIdle="2"`). Please see the "**Optimize context.xml**" section in [Configuring the CAST Application Analytics Dashboard for large numbers of Applications](#) for more information about these parameters, or consult the relevant Apache Tomcat documentation: <https://tomcat.apache.org>.

### Tomcat 8 or 8.5 - configuration for one Dashboard Service stored on CSS2

If you are using **Tomcat 8** or **8.5** to host the CAST Application Engine Dashboard and **CSS2** to host you CAST Dashboard Services, please follow these instructions:

- Open the **context.xml** file (as detailed above) with a Text editor.
- By default, the file will contain an **uncommented** template as follows - this is tailored for connections to **CAST schemas** located on **CAST Storage Service 2** using **Tomcat 7**:

```
<Resource name="jdbc/domains/AED" url="jdbc:postgresql://localhost:2280/postgres"
    initConnectionSqls="SET search_path TO [Central Schema];"
    username="operator" password="CastAIP"
    auth="Container" type="javax.sql.DataSource" driverClassName="org.postgresql.Driver"
    validationQuery="select 1"
    initialSize="5" maxActive="20" maxIdle="10" maxWait="-1"/>
```

- You need to comment this template by adding `<!--` and `-->` around the section:

```
<!--
<Resource name="jdbc/domains/AED" url="jdbc:postgresql://localhost:2280/postgres"
    initConnectionSqls="SET search_path TO [Central Schema];"
    username="operator" password="CastAIP"
    auth="Container" type="javax.sql.DataSource" driverClassName="org.postgresql.Driver"
    validationQuery="select 1"
    initialSize="5" maxActive="20" maxIdle="10" maxWait="-1"/>
-->
```

- Now locate the **Tomcat 8** template for **CAST Storage Service 2** - this is located at the top of the following section:

```
<!-- For Tomcat 8 : http://tomcat.apache.org/tomcat-8.0-doc/jndi-datasource-examples-howto.html &
http://commons.apache.org/proper/commons-logging/configuration.html

template for resources on CSS :
<Resource name="jdbc/domains/${domainName}"
    url="jdbc:postgresql://${host}:${port}/postgres"
    connectionInitSqls="SET search_path TO ${schema};"
    username="${user}" password="${password}"

    auth="Container" type="javax.sql.DataSource" driverClassName="org.postgresql.Driver"
    validationQuery="select 1" initialSize="5" maxTotal="20" maxIdle="10" maxWaitMillis="-1"/>

template for resources on Oracle :
<Resource name="jdbc/domains/{domainName}"
    url="jdbc:oracle:thin:@{server}:{port}:{instance}"
    connectionInitSqls="ALTER SESSION SET CURRENT_SCHEMA={schema}"
    username="{user}" password="{password}"

    auth="Container" type="javax.sql.DataSource" driverClassName="oracle.jdbc.OracleDriver"
    validationQuery="select 1 from DUAL" initialSize="5" maxTotal="20" maxIdle="10" maxWaitMillis="-1" />

template for resources on MS SQL Server :
<Resource name="jdbc/domains/{domainName}"
    url="jdbc:jtds:sqlserver://{host};instance={instance}"
    connectionInitSqls="use {database}"
    username="{user}" password="{password}"

    auth="Container" type="javax.sql.DataSource" driverClassName="net.sourceforge.jtds.jdbc.Driver"
    validationQuery="select 1" initialSize="5" maxTotal="20" maxIdle="10" maxWaitMillis="-1"/>
-->
```

- Copy the **Tomcat 8 for CSS2** template into **uncommented white space**:

```
<Resource name="jdbc/domains/${domainName}"
    url="jdbc:postgresql://${host}:${port}/postgres"
    connectionInitSqls="SET search_path TO ${schema};"
    username="${user}" password="${password}"

    auth="Container" type="javax.sql.DataSource" driverClassName="org.postgresql.Driver"
    validationQuery="select 1" initialSize="5" maxTotal="20" maxIdle="10" maxWaitMillis="-1"/>
```

- Set the `${domainName}` value to **AED**

- Set the **URL** parameter to the server on which the **Dashboard Service** is located (usually you just need to alter the **\$(host)** and **\$(port)** values to the name/port of the server on which the host CSS is located).
- Change **\$(schema)** to the name of the **Dashboard Service** (see [Install CAST schemas](#)).
- Ensure the **\$(user)** and **\$(password)** parameters are correct - use the **operator** and **CastAIP** username/password combination (note that if you need to **encrypt** the username and password to avoid entering names in clear text, please see: [CAST-AAD-AED - Encrypt login and password for datasource and Active Directory](#))
- You should end up with a section like this:

```
<Resource name="jdbc/domains/AED"
  url="jdbc:postgresql://NEFYN:2280/postgres"
  connectionInitSqls="SET search_path TO V80_CENTRAL_CSS2;"
  username="operator" password="CastAIP"

  auth="Container" type="javax.sql.DataSource" driverClassName="org.postgresql.Driver"
  validationQuery="select 1" initialSize="5" maxTotal="20" maxIdle="10" maxWaitMillis="-1"/>
```

- Following any changes you make, **save the context.xml file**.



Note that it is **critically important** to configure the **initialSize**, **maxTotal** and **maxIdle** parameters according to your consumption use case and the number of schemas you want to configure. The default values are high but could probably be decreased to limit the number of active connections to your database instance (eg. `initialSize="2" maxActive="10" maxIdle="2"`). Please see the **"Optimize context.xml"** section in [Configuring the CAST Application Analytics Dashboard for large numbers of Applications](#) for more information about these parameters, or consult the relevant Apache Tomcat documentation: <https://tomcat.apache.org>.

## Configuring access to Dashboard Services stored on Microsoft SQL Server or Oracle Server for Tomcat 7/8/8.5

If your target Dashboard Service is located on a Microsoft SQL Server or Oracle Server, you need to:

- Remove or re-comment the **uncommented** template for **CSS2/Tomcat 7** from the **context.xml** file
- Copy the **commented** template available in the **context.xml** file for Microsoft SQL Server or Oracle Server into uncommented space - ensure you chose the correct template for **Tomcat 7** or **Tomcat 8 / 8.5**.
- Make the same configuration changes as suggested above:
  - Set the **URL** parameter to the server on which the **Dashboard Service** is located (please see <http://jtds.sourceforge.net/faq.html#urlFormat> for more information about the Microsoft SQL Server jTDS JDBC driver URL format)
  - Set the **initConnectionSqls** parameter to the name of the **Dashboard Service** {database} or {schema}
  - Ensure the **username** and **password** parameters for your target RDBMS are correct - these provide access to the Dashboard Service database or schema.
- Following any changes you make, **save the context.xml file**.

## Configuring access to multiple Dashboard Services for Tomcat 7/8/8.5

If you would like to configure access to multiple Dashboard Services for use with your CAST Application Engineering Dashboard, please do as follows:

- For **each additional** Dashboard Service you want to configure access to, please add an **uncommented configuration template** (these are supplied in the **context.xml** file - ensure you chose the correct template for **Tomcat 7** or **Tomcat 8 / 8.5**) for the target RDBMS/CSS2 underneath the section configured for your initial Dashboard Service
- First change the **name** parameter to something other than **"jdbc/domains/AED"** (the default "name" used in the pre-provided templates) - CAST recommends using the name of your target Dashboard Service - for example **"jdbc/domains/V80\_CENTRAL\_CSS2"** (you must retain the **"jdbc /domains"** part of the the name). This is because each Dashboard Service you connect to the CAST Application Engineering Dashboard must have a **unique** "name" parameter
- Set the **URL** parameter to the server on which the additional **Dashboard Service** is located (usually you just need to alter the "localhost" to the name of the server on which the host RDBMS is located).
- Then, depending on your target RDBMS/CSS2:
  - **CSS2:** Change **[Dashboard Schema]** to the name of the additional **Dashboard Service schema** (see [Install CAST schemas](#)).
  - **Microsoft SQL Server/Oracle Server:** Set the **initConnectionSqls** parameter to the name of the **Dashboard Service** {database} or {schema}
- Ensure the **username** and **password** parameters for your target RDBMS/CSS2 are correct (note that if you need to encrypt the username and password to avoid entering names in clear text, please see: [CAST-AAD-AED - Encrypt login and password for datasource and Active Directory](#))
- Following any changes you make, **save the context.xml file**.

## Connection pooling

It is possible to modify the connection configuration for the CAST Application Engineering Dashboard to pool multiple CAST Dashboard Service schemas /databases in one resource "domain". In other words, you no longer have to configure one resource (in the context.xml) per CAST Dashboard Service "schema/database", instead, one resource (in the context.xml) can function for multiple CAST Dashboard Service schemas/databases. Please see [Appendix x - CAST Dashboard Service schema connection configuration pooling](#) for more information.

## Install the license key

As explained in [CAST-AED-RESTAPI - Dashboard Service license key configuration](#), when you want to access a **CAST Dashboard Service** using the **CAST RestAPI** (i.e. via the **CAST Application Engineering Dashboard**, or via the **CAST Report Generator**), a special **license key** is required. This license key grants specific access to one or multiple CAST Dashboard Services for the web application in which it is installed (i.e. the CAST Application Engineering Dashboard or the CAST RestAPI).

You must therefore install the license key and, if you are using a restricted license key, define data access authorization. These two steps are explained in [CAST-AED-RESTAPI - Dashboard Service license key configuration](#) in the sections **How do I install a license key?** and **How to authorize users when using a RESTRICTED license key.**

## Restart Tomcat

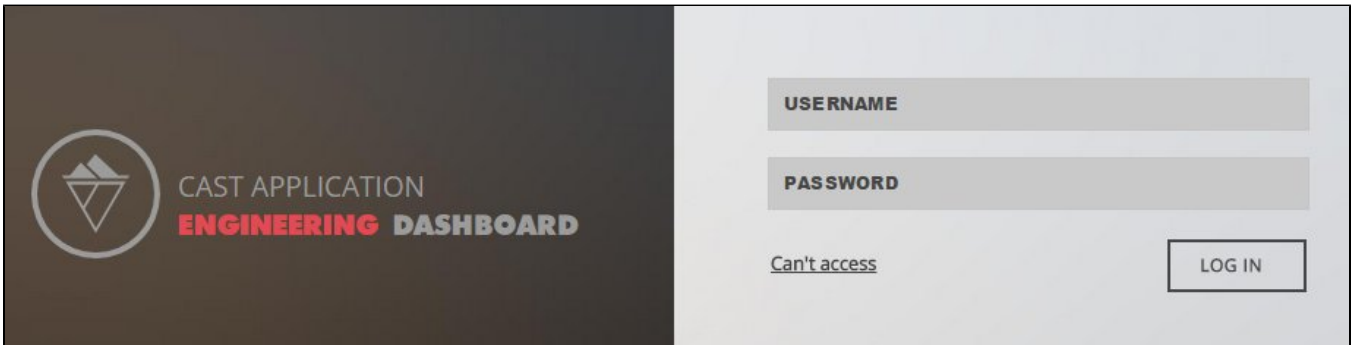
Before proceeding, ensure you **restart** your application server so that the configuration changes you made are taken into account.


## Test access to the CAST Application Engineering Dashboard

- You can now access the CAST Application Engineering Dashboard using the URL:

```
http://<server_name>:[<port_number>]/CAST-AED
```

- You should see the login page as follows - this indicates that the initial setup was successful:



 Error messages are documented in [CAST-AAD-AED - Error Messages](#).

## Step 2: Configure user authentication

Step 2 involves configuring how your users will authenticate with the CAST Application Engineering Dashboard. Most organizations opt for LDAP/Active Directory integration so that users can use their corporate username/password to access the resources they need. The CAST Application Engineering Dashboard also has a built in username/password authentication mechanism which is enabled "out of the box".

Please see [Configuring user authentication](#) for more information.

## Step 3: Configure data authorization

Step 3 involves configuring data **Authorization**. An authorization defines permission to a user or group of users to access and "consume the data" in a specific Application in the CAST Application Engineering Dashboard. If permission is not granted then any information related to this Application will be not accessible: application properties such as name, technologies or grades and measures, etc. Therefore, an Authorization must be defined before a user /group of users can access a specific application.

Please see [Configuring data authorization](#) for more information.

## Additional information

You can find additional information about **advanced configuration changes** in [CAST Application Engineering Dashboard](#) such as:

- [CAST-AED - Dashboard wide configuration options in json](#)
- [CAST-AED - Tile management](#)
- [CAST-AED - Using the GUI and CLI tools](#)

You can find additional information about **advanced security configuration changes** for the **CAST Application Engineering Dashboard**, such as:

- [CAST-AAD - Configuring user authentication](#)
- [CAST-AAD - Configuring data authorization](#)

- [CAST-AED - Configuring user authentication](#)
- [CAST-AED - Configuring data authorization](#)
- [CAST-AAD-AED - Configuring the Log and Audit Trail](#)
- [CAST-AAD-AED - Encrypt login and password for datasource and Active Directory](#)

There is also additional information about implementing an **enhanced security configuration** for your application server in [Common security configuration options for web application deployment](#):

- [Configuring the use of secure https protocol with Tomcat for the CAST web applications](#)
- [Setting up a Reverse Proxy on Apache web server](#)
- [Disabling insecure HTTP methods in Apache Tomcat - webdav](#)
- [Disabling weak SSL cipher suites to improve security](#)