

CAST Dashboard Service schema connection configuration pooling

- Introduction
 - Note about pooling Measurement Service schemas
- Connection pooling
 - domains.properties
 - context.xml
 - Tomcat 7
 - Tomcat 8/8.5
 - maxActive and maxTotal attributes
 - When using the combined CAST-Health-Engineering.war file
 - domain.properties
 - context.xml
- Example
 - Standard configuration without connection pooling
 - Connection pooling configuration
 - domain.properties with connection pooling
 - context.xml with connection pooling



Summary: it is possible to modify the connection configuration for the **Engineering Dashboard** and the **RestAPI** to pool multiple CAST Dashboard Service schemas in one resource "domain". In other words, you no longer have to configure one resource (in the context.xml) per CAST Dashboard Service schema, instead, one resource (in the context.xml) can function for multiple CAST Dashboard Service schemas.



This page is **only relevant for Dashboard 1.x**. Dashboards 2.x already use connection pooling by default.

Introduction

When configuring a connection for the Engineering Dashboard or the RestAPI, **each CAST Dashboard Service database/schema** that you want to reference requires one **UNIQUE <resource>** entry in the **context.xml** file. For example, the following code block shows the configuration required for a **Tomcat 7 application server** to specify **four Dashboard Service schemas**, two on one CAST Storage Service server and two on another (note that each <resource> has a different "name" attribute):

- V83_CENTRAL1_CSS_A
- V83_CENTRAL2_CSS_A
- V82_CENTRAL1_CSS_B
- V82_CENTRAL2_CSS_B

```

<Resource name="jdbc/domains/AED1" url="jdbc:postgresql://NEFYN:2280/postgres"
  initConnectionSqls="SET search_path TO V83_CENTRAL1_CSS_A;"
  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"/>

<Resource name="jdbc/domains/AED2" url="jdbc:postgresql://NEFYN:2280/postgres"
  initConnectionSqls="SET search_path TO V83_CENTRAL2_CSS_A;"
  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"/>

<Resource name="jdbc/domains/AED3" url="jdbc:postgresql://CEIRIAD:2280/postgres"
  initConnectionSqls="SET search_path TO V82_CENTRAL1_CSS_B;"
  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"/>

<Resource name="jdbc/domains/AED4" url="jdbc:postgresql://CEIRIAD:2280/postgres"
  initConnectionSqls="SET search_path TO V82_CENTRAL2_CSS_B;"
  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"/>

```

This configuration works well when you only need to reference a **handful** of CAST AIP schemas/RDBMS, however, if you need to **reference tens or hundreds** of CAST AIP schemas, then it can quickly become arduous to configure each resource manually. In addition, for each resource you configure in the context.xml, a database connection will be opened to the target RDBMS - if the CAST AIP schemas/databases are located on the SAME RDBMS, then multiple connections to that same RDBMS will be opened by the web application, which can impact performance.

In order to simplify the configuration of multiple database connections/resources in the context.xml file and as a direct response to the issues outlined above, CAST has introduced the ability to use a **connection pooling** feature. This feature is detailed below.

Note about pooling Measurement Service schemas

The connection pooling feature is aimed at pooling multiple **CAST Dashboard Service schemas** located on the same RDBMS. Whilst technically possible to pool multiple **CAST Measurement Service schemas** this is not recommended nor supported:

- The Health Dashboard does NOT support the display of data from **multiple Measurement Service schemas**, i.e. the dashboard expects one resource named "AAD", which must correspond to one single CAST Measurement Service.
- The CAST Report Generator can technically support connections to multiple Measurement Service schemas, however, this configuration is not recommended nor supported.

Connection pooling

The connection pooling feature is handled through two specific files in the deployed WAR file:

```

CATALINA_HOME\webapps\

```




If you edit either of these files, ensure you restart the web application server so that the changes are taken into account.

domains.properties

The **domains.properties** file contains the configuration required for each domain. By default this is delivered as follows:

```
# empty lines in this file lead to connection error, remove all empty lines
# The Tomcat Connection pool is configured as a resource in the META-INF\Context.xml
# - You have to align [Resource1] with the resource name configured in META-INF\Context.xml
# - You have to replace [Central Schema1] by the central schema name
# - Domains name must be unique
# AED1=Resource1,[Central Schema1]
# AED2=Resource1,[Central Schema2]
```

- **AED1 / AED2** refers to the name of the **domain** you are creating. You can use any domain name notation you want, but **each domain name must be unique**.
- **Resource1** refers to the **name attribute** used in the **<resource>** tag in the **context.xml file**. This identifies the connection parameters (as defined in the context.xml file) for the schema/database. This item can be repeated as many times as necessary - each schema that is located on the same RDBMS will use the same **Resource1**.
- **[Central Schema1] / [Central Schema2]** refers to the name of the CAST Dashboard Service schema you want to connect to.
- The **#** is used to comment out a line. By default all configuration is commented out and therefore no connection pooling is in effect.
- To avoid database connections errors, please ensure that:
 - any **empty lines** in the domains.properties file are removed
 - any **white spaces** in the domain configuration are removed - for example a space after the comma as in "**AED1=Resource1, CENTRAL_SCHEMA**" will cause an issue, whereas "**AED1=Resource1,CENTRAL_SCHEMA**" will not.

 In simple terms, you need to create one line (i.e. AED1=Resource1,[Central Schema1]) per CAST Dashboard Service schema you need to access. So if you have 200 CAST Dashboard Service schemas:

- you need to enter 200 lines
- if all 200 schemas are on the same RDBMS, then the **Resource1** will be identical in each line
- all domain entries must be unique (i.e. **AED1 / AED2**)

context.xml

In the context of the connection pooling feature, the **context.xml** is used only to define the connection parameters to an RDBMS and the **Resource1** item (**name attribute** used in the **<resource>** tag). The required format of the **<resource>** tag is very similar to the format used for a standard connection without connection pooling, except that the CAST Dashboard Service schema/database is not defined at all (i.e. the **initConnectionSqls** (Tomcat 7) or **connectionInitSqls** (Tomcat 8/8.5) attributes are not defined). The schema/database is instead defined in the **domains.properties** file:

Tomcat 7

```
<!-- CSS2 -->
<Resource name="jdbc/domains/{MYRESOURCE}"
  url="jdbc:postgresql://{host}:{port}/postgres"
  username="{user}" password="{password}"
  auth="Container" type="javax.sql.DataSource" driverClassName="org.postgresql.Driver"
  validationQuery="select 1" initialSize="5" maxActive="20" maxIdle="10" maxWait="-1"/>

<!-- Oracle -->
<Resource name="jdbc/domains/{MYRESOURCE}"
  url="jdbc:oracle:thin:@{server}:{port}:{instance}"
  username="{user}" password="{password}"
  auth="Container" type="javax.sql.DataSource" driverClassName="oracle.jdbc.OracleDriver"
  validationQuery="select 1 from DUAL" initialSize="5" maxActive="20" maxIdle="10" maxWait="-1" />

<!-- Microsoft SQL Server -->
<Resource name="jdbc/domains/{MYRESOURCE}"
  url="jdbc:jtds:sqlserver://{host};instance={instance}"
  username="{user}" password="{password}"
  auth="Container" type="javax.sql.DataSource" driverClassName="net.sourceforge.jtds.jdbc.Driver"
  validationQuery="select 1" initialSize="5" maxActive="20" maxIdle="10" maxWait="-1"/>
```

Tomcat 8/8.5

```

<!-- CSS2 -->
<Resource name="jdbc/domains/{MYRESOURCE}"
  url="jdbc:postgresql://{host}:{port}/postgres"
  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"/>

<!-- Oracle -->
<Resource name="jdbc/domains/{MYRESOURCE}"
  url="jdbc:oracle:thin:@{server}:{port}:{instance}"
  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" />

<!-- Microsoft SQL Server -->
<Resource name="jdbc/domains/{MYRESOURCE}"
  url="jdbc:jtds:sqlserver://{host};instance={instance}"
  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"/>

```

maxActive and maxTotal attributes

When configuring connection pooling, CAST highly recommends that you tune the **maxActive** (Tomcat 7) and **maxTotal** (Tomcat 8/8.5) attributes in each **<resource>** tag according to your consumption use case and the number of schemas you are pooling. These parameters govern the maximum number of active connections that can be allocated at the same time. The default value is 20.

Remember that when connection pooling is in use, there is only one **<resource>** tag for each target RDBMS, so you can increase/decrease the number of active connections to fit your need.

When using the combined CAST-Health-Engineering.war file

If you are using the combined **CAST-Health-Engineering.war** file (where you have both Measurement Service and Dashboard Service schemas configured in the context.xml) and you want to use connection pooling for your Dashboard Services, you must ensure you configure the Measurement Service schema connection via the connection pooling mechanism as well otherwise it will be ignored.

For example, you have the following configuration in your **context.xml** for a **Tomcat 7 application server** prior to invoking connection pooling:

```

<!-- Health Dashboard configuration -->
<Resource name="jdbc/domains/AAD" url="jdbc:postgresql://BOB:2280/postgres"
  initConnectionSqls="SET search_path TO v83_MEASURE_BOB;"
  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"/>

<!-- Engineering Dashboard configuration -->
<Resource name="jdbc/domains/AED1" url="jdbc:postgresql://FRED:2280/postgres"
  initConnectionSqls="SET search_path TO V83_CENTRAL1_FRED;"
  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"/>

<Resource name="jdbc/domains/AED2" url="jdbc:postgresql://FRED:2280/postgres"
  initConnectionSqls="SET search_path TO V83_CENTRAL2_FRED;"
  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"/>

```

When you invoke connection pooling for the two Dashboard Service schemas on the FRED CSS server, you need to do the following for the Measurement Service schema:

domain.properties

```

# empty lines in this file lead to connection error, remove all empty lines
# The Tomcat Connection pool is configured as a resource in the META-INF\Context.xml
# - AAD domain is mandatory for the portal
# - You can only connect one measure schema and domain must be "AAD"
# - You have to align [Resource1] with the resource name configured in META-INF\Context.xml
# - You have to replace [Measure Schema] by your measure schema name you want to connect
# AAD=Resource1,[Measure Schema]
# - You have to align [Resource1] with the resource name configured in META-INF\Context.xml
# - You have to replace [Central Schema1] by the central schema name
# - Domains name must be unique
# AED1=Resource1,[Central Schema1]
# AED2=Resource1,[Central Schema2]
AAD=AAD,V83_MEASURE_BOB
DOMAIN1=AED,V83_CENTRAL1_FRED
DOMAIN2=AED,V83_CENTRAL2_FRED

```



Note that the DOMAIN for a Measurement Service schema MUST be set to "AAD" as shown above, otherwise the Health Dashboard will not function correctly.

context.xml

```

<!-- Health Dashboard configuration -->
<Resource name="jdbc/domains/AAD" url="jdbc:postgresql://BOB:2280/postgres"
  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"/>

<!-- Engineering Dashboard configuration -->
<Resource name="jdbc/domains/AED" url="jdbc:postgresql://FRED:2280/postgres"
  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"/>

```

Example

This example will configure the following four Dashboard Service schemas via connection pooling:

Dasboard Service schema name	CSS name
V82_CENTRAL1_FRED	FRED:2280
V82_CENTRAL2_FRED	FRED:2280
V81_CENTRAL1_BILL	BILL:2280
V81_CENTRAL2_BILL	BILL:2280

Standard configuration without connection pooling

Without connection pooling (Tomcat 7/CSS2) you would need to configure the following in the context.xml - **one unique <resource> per Dashboard Service schema:**

```

<Resource name="jdbc/domains/AED1" url="jdbc:postgresql://FRED:2280/postgres"
  initConnectionSqls="SET search_path TO V82_CENTRAL1_FRED;"
  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"/>

<Resource name="jdbc/domains/AED2" url="jdbc:postgresql://FRED:2280/postgres"
  initConnectionSqls="SET search_path TO V82_CENTRAL2_FRED;"
  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"/>

<Resource name="jdbc/domains/AED3" url="jdbc:postgresql://BILL:2280/postgres"
  initConnectionSqls="SET search_path TO V81_CENTRAL1_BILL;"
  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"/>

<Resource name="jdbc/domains/AED4" url="jdbc:postgresql://BILL:2280/postgres"
  initConnectionSqls="SET search_path TO V81_CENTRAL2_BILL;"
  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"/>

```

Connection pooling configuration

To convert the above standard configuration to connection pooling:

domain.properties with connection pooling

Ensure each line is uncommented (except for the first) and then configure four unique domains (one for each CAST Dashboard Service schema) and ensure that the MYRESOURCE item is correctly defined: in the example below, there are two MYRESOURCE items, AED1 and AED2 which refer to each RDBMS:

```

#New connection pool management
DOMAIN1=AED1,V82_CENTRAL1_FRED
DOMAIN2=AED1,V82_CENTRAL2_FRED
DOMAIN3=AED2,V81_CENTRAL1_BILL
DOMAIN4=AED2,V81_CENTRAL2_BILL

```

context.xml with connection pooling

You can see that there are now only **two <resource> elements**:

- The first one defines MYRESOURCE "AED1" pointing to the CSS called "FRED" - two schemas V82_CENTRAL1_FRED and 82_CENTRAL2_FRED are hosted on this CSS
- The second <resource> defines the MYRESOURCE "AED2" called "BILL" - two schemas V82_CENTRAL1_BILL and 82_CENTRAL2_BILL are hosted on this CSS

```

<Resource name="jdbc/domains/AED1" url="jdbc:postgresql://FRED:2280/postgres"
  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"/>

<Resource name="jdbc/domains/AED2" url="jdbc:postgresql://BILL:2280/postgres"
  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"/>

```