Table of Contents

For security reasons, the validity of client/server connections or sessions is limited. After inactivity (no user interaction) of a predefined time, the connection or session is invalidated, and the client has to reconnect if necessary. If no timeouts are defined for JVx, the connections/sessions are not invalidated.

Timeouts can be defined in the following ways:

- Changing timeouts for all applications of a given server (globally and at a central location, per configuration)
- Changing timeouts by application (at a central location, per configuration)
- Changing timeouts by connection/session (in the source code)

All Applications

Multiple applications can be in use on a server; it is, therefore, possible that timeouts are set for all of a server´s applications. The server configuration is handled by the file *config.xml* in the server directory (*/rad/server*).

Timeouts are defined as follows:

config.xml

```
<?xml version="1.0" encoding="UTF-8"?>

<server>
  <timeout>
    <!-- MasterSession timeout in minutes -->
    <mastersession>360</mastersession>
    <!-- SubSession timeout in minutes -->
    <subsession>360</subsession>
  </timeout>
</server>
```

The mastersession timeout sets the timeout for the MasterSession and the subsession timeout sets the timeout for the SubSessions fest. Time is entered in minutes.

Per Application

The application usually uses the server´s timeout. However, the application can define its own timeout and overrule the server´s timeouts.

Configuration is done using the file *config.xml* in the directory */rad/apps/<name>/config.xml*:

config.xml

```
<?xml version="1.0" encoding="UTF-8"?>

<application>
```

```
  <timeout>
    <!-- MasterSession timeout in minutes -->
    <mastersession>180</mastersession>
  </timeout>
</application>
```

The example does not define a timeout for the subsession. The timeout for the subsession is based on the server configuration.

Per Connection/Session

The connection timeout can be modified directly in the client code, e.g.:

```
masterConnection.setTimeout(120);
```

and on the server side via SessionContext, e.g.:

```
SessionContext.getCurrentSession().setMaxInactiveInterval(60);
```

Notice

If the master connection is not used, because you only communicate via a sub connection, the master connection remains active. The activity of the sub connection is also counted as activity in the master connection.

It's also possible to configure the timeout in seconds. Simply use the following key:

```
<timeoutSeconds>
  ...
</timeoutSeconds>
```

If timeout and timeoutSeconds were set, then the minimum of both will be used.

From:
https://doc.sibvisions.com/ - **Documentation**

Permanent link:
https://doc.sibvisions.com/jvx/communication/timeouts

Last update: **2020/07/08 17:47**