

Table of Contents

If you're using a desktop client for your application and a communication protocol like http, you have to know about the integrated alive mechanism of JVx.

What's the alive mechanism?

The alive mechanism allows detection of invalid connections. This is important for automatic session handling and memory management. If you start your application, every application has one or more connections to the server-side. If you close your application, all used resources will be released. If you're not using your application for a long time, usually the application will time out and all resources will be released. Sometimes, it happens that you don't use timeouts or your internet/server connection is broken. In such special cases, it should be possible to release all used connections if possible. So, every application sends an alive message (ping) to the server-side. The interval is configured with 30 seconds, as default. The send mechanism is optimized and doesn't send messages too often. So if you're working with your application, an alive message will be every 30 seconds of user inactivity. Otherwise, user activity is an implicate alive message.

The server-side checks the timestamp of last received alive messages and if it didn't receive a message for $4 * \text{alive interval}$ (with default settings: 120 seconds), the connection will be closed and all resources will be released. The server-side alive check already takes care of standard user activity. So if your user is working with the application, the alive mechanism knows that the application is still alive.

You have different options to configure alive mechanism:

- set the alive interval (milliseconds) directly for your Master Connection in your application on client-side: `setAliveInterval(long pInterval)`
- set the alive interval in server-side config.xml of your application:

```
<connection>
  <property name="client.alive.interval">4000</property>
</connection>
```

If you set the interval < 0 , sending alive AND server-side alive check will be disabled. If you set the interval between 0 and 1000, the default interval (30000) will be set automatically.

Configure alive check

If the last alive message is longer than $4 * \text{alive interval}$, then a connection is closed. If you set a short alive interval like 2 seconds, automatic connection close will happen after 8 seconds if e.g. network is unstable. You have some options to change the timeout:

- set a fixed alive timeout via server-side config.xml:

```
<alive>
  <!-- minutes -->
  <timeout>1</code>
  <!-- seconds -->
  <timeoutSeconds>20</code>
  <!-- only factor -->
```

```
<timeoutFactor>8</code>  
</alive>
```

If you set `timeoutSeconds`, `timeout` will be ignored.

Setting a short alive interval is useful for “polling” server-side for changes. But if your network connection isn't very stable, think about a custom alive timeout.

From:
<http://doc.sibvisions.com/> - **Documentation**

Permanent link:
<http://doc.sibvisions.com/jvx/communication/alive>



Last update: **2023/08/08 09:37**