Firing Events from Java EE Applications

Table of Contents

SLEE Connection Factory

The JAIN SLEE specification includes an API for interaction with JAIN SLEE container, with the interface `javax.slee.connection.SleeConnectionFactory` upfront, which allows external applications, such as an EJB, to create connections to a specific JAIN SLEE instance, and use that to fire events.

JAIN SLEE provides two different implementations of the API, one for access in the same JVM, another for remote access. Both implementations expose the SLEE Connection Factory in the local JNDI tree, thus the same code is used by the application independently of the implementation used.

Local SLEE Connection Factory

JAIN SLEE container instances always expose the local SLEE Connection Factory in its JVM, which means that an external application running in the same JVM doesn't need any additional tools or setup to use it.

Remote SLEE Connection Factory

JAIN SLEE includes a tool, named Remote SLEE Connection Tool, which is a JCA connector that can be deployed in any Java EE application server. Once deployed this connector installs the factory into JNDI, which communicates with the remote SLEE container through RMI.

Restcomm Remote SLEE Connection Tool is bundled in all Restcomm JAIN SLEE releases, in the tools/remote-slee-connection directory, including its own documentation.

Using SLEE Connection Factory

Below is example code which retrieves the SLEE Connection Factory and uses it to fire an event into the JAIN SLEE container. The code is the same whether the SLEE container is in the same JVM or not.

```
    // retrieves JNDI context
    InitialContext context = new InitialContext();

    // retrieves the connection factory from JNDI
    SleeConnectionFactory factory = (SleeConnectionFactory) context.lookup
("java:/MobicentsConnectionFactory");

    // creates a connection to the SLEE container
    SleeConnection connection = factory.getConnection();

    // creates the activity handle which will be used to fire the event
    ExternalActivityHandle handle = connection.createActivityHandle();

    // retrieves the event type ID
    EventTypeID eventTypeID = connection.getEventTypeID("CustomEvent", "...", "1.0");

    // creates the event object
    CustomEvent eventObject = new CustomEvent();

    // fires the event in the remote SLEE container
    connection.fireEvent(eventObject, eventTypeID, handle, null);

    // closes the connection
    connection.close();
```