EDT 0.8 Stakeholder Meeting
SQL JNDI Data Sources

February 8, 2012
Stakeholder Meeting

- EDT 0.8 is currently under development
  - For more on EDT 0.8 plans, see [http://wiki.eclipse.org/EDT:0.8.0_Planning](http://wiki.eclipse.org/EDT:0.8.0_Planning)
- Meeting Goal - Discuss the design and implementation of a particular feature to ensure it meets stakeholder needs
- Plans are subject to change (mostly based on your input!)
- Feel free to ask questions via the phone or chat to All.
- Press *6 to mute / unmute your phone. Please mute your phone unless you are asking a question.
Resource bindings

- Bind (or access) a resource, like a database or service.
  - Connection details are separate from the application logic
- Configured via the deployment descriptor
- Allows you to make changes to resources without changing your application logic

- For more information, see the Resource Bindings Introduction wiki page:
JNDI basics

- **What is JNDI?**
  - “Java Naming and Directory Interface”
  - Allows lookup of objects by name, in this case SQL data sources

- **Main benefits of using JNDI:**
  - Applications do not hardcode connection information, it is configured as a server administration task
  - Connection pooling has significant performance improvements

- **Authentication**
  - Container-based: the data source has the credentials, application provides no credentials
  - Application-based: The application provides credentials to the data source
    - Not all drivers support application-based! e.g. Tomcat’s BasicDataSource class.
    - If unsure which type to use, container-based is preferred
New SQL type

- Resource bindings are recommended but it's still possible to write the EGL code to connect to a database using JNDI.

- The new type is eglx.persistence.sql.SQLJNDIDataSource

- It extends eglx.persistence.sql.SQLDataSource

- It has no new API, just connects in a different way.

- Using container-based authentication:
  ```java
ds SQLDataSource? = new SQLJNDIDataSource("jdbc/myDB");
```

- Using application-based authentication:
  ```java
credentials Dictionary{user="myUser", password="myPassword"];
ds SQLDataSource? = new SQLJNDIDataSource("jdbc/myDB", credentials);
```
**Updated Deployment Descriptor Editor**

- **SQL Database Binding**
  - Set the properties of the selected SQL database binding

  - **Name:** NewDB2forLinuxUNIXandWindows

  - **Use a connection profile**
    - New DB2 for Linux, UNIX, and Windows

  - **Specify connection information**
    - **SQL Database:**
    - **JDBC Driver Class:**
    - **Connection URL:**
    - **User ID:**
    - **User Password:**
    - **Default Schema:**
    - **Class Locations:**

  - **Connect using a JNDI data source**
    - The application will use JNDI for this binding to access the data source.
    - **JNDI Name:** jdbc/NewDB2forLinuxUNIXandWindows

  - **Use application-based authentication**
IDE Test Server

- Jetty-based server for running services inside the IDE without deploying
- Supports a true JNDI environment
- Uses SQL bindings from deployment descriptors to configure the available data sources
- Configure Apache Tomcat runtime in the workspace via Preferences > Server > Runtime Environments.
- If Tomcat is defined, connection pooling is supported using BasicDataSource
  - Just need to define an Apache Tomcat runtime in the workspace and it will be found
Question for the community

- Should we configure all database bindings to use JNDI by default?
- Or provide an option to configure it in the wizard?
- Questions or comments on JNDI support?

- JNDI enhancement in Bugzilla
  - https://bugs.eclipse.org/bugs/show_bug.cgi?id=361030
Be part of the open-source process!

- Look at EDT’s Bugzilla, used to track enhancements and bugs
  - EDT components https://bugs.eclipse.org/bugs/describecomponents.cgi?product=EDT
  - All EDT .8 items query
    https://bugs.eclipse.org/bugs/buglist.cgi?query_format=advanced;version=0.8.0;classification=Tools;product=EDT
  - If you’re interested in a Bugzilla item, add yourself to the cc: list
  - Add comments (after signing on with Eclipse ID)

- Ask questions on the EDT Forum

- Familiarize yourself with the Wiki documents
  - http://wiki.eclipse.org/EDT

- Read the EDT team blog, available via the Community page, http://eclipse.org/edt/#community

- Next stakeholder meeting – IBM i support
  - Watch for forum for an announcement

- THANKS!!