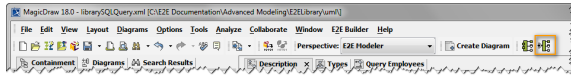# Creating a Library Diagram

The **E2E Library Wizard** helps you to define all components, classes, interfaces, artifacts, and dependencies that are needed to build a complete library diagram. It guides you through all necessary steps and supports you with the customization of all artifacts.The Library Wizard can be started by selecting the menu icon **E2E Library Wizard** in he menu bar shown below. It is used when starting a new library diagram.

Alternatively, you could select **Tools > Library Wizard** from the Model Compiler window.

Wanting to edit an existing library diagram use the **E2E Edit Library Diagram Wizard** that supports you in changing a specific part of the library diagram. It can be started by selecting the appropriate icon from the middle bar of the library diagram.

> You need to click the icon Edit Component Diagram in the diagram toolbar and not the icon E2E Components Wizard in the menu bar in order to edit an existing component diagram. The E2E Components Wizard from the menu bar will delete the existing component diagram and create a new one from scratch.

The following chapters are ordered according the steps needed to create a library component diagram and describe each creation step in detail.

1. Assigning the Diagram Name
2. Defining the Library Composite
3. Defining Classes, Interfaces and Signals
4. Defining Backend Aliases
5. Finishing the Library Wizard

Assigning the Diagram Name

After starting the Library Wizard, the first dialog opens and shows the diagram name.
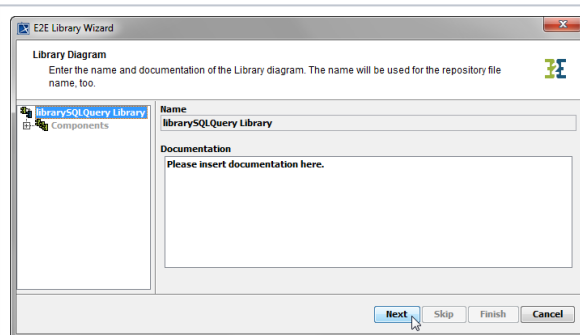

The **Name** field is disabled. This is due to the fact, that the library name is equal to the libraries file name, just without the extension.

> Editing the library name would only be possible by changing the actual model file name.

Enter a meaningful description of the library in field **Documentation**.
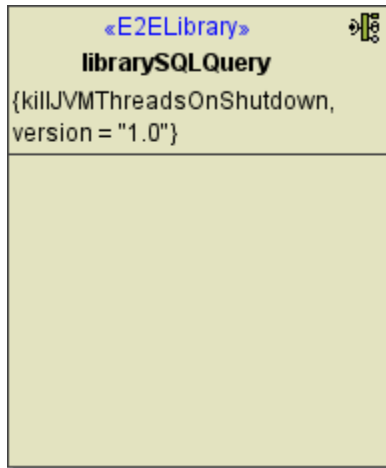
Click **Next** to proceed or **Cancel** to abort.

The Library Wizard can be canceled any time.

	After clicking cancel, the Library Wizard prompts for confirmation.
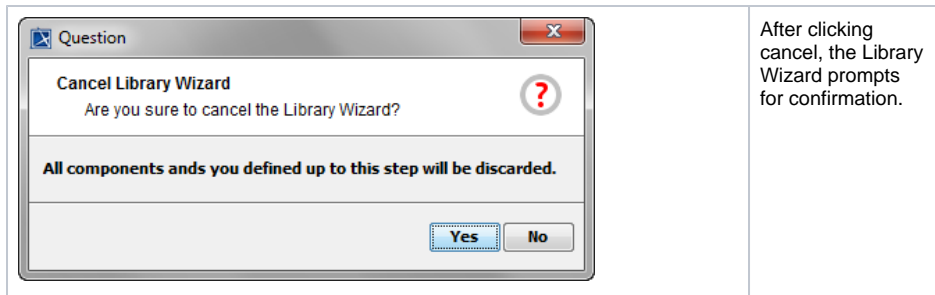
Defining the Library Composite

In the next step, you will define both the library composite. The composite represents the repository of the library and will contain all necessary configuration information.

Clicking **Next** after assigning the diagram name will bring you directly to the following dialog. Create the library composite here.

The library component contains the tagged value **version**. It will be initialized with value 1.0 and can be used to distinct different versions of the library.

> If you change the libraries implementation, it is recommended to increase the version number. This has to be done manually. On re-import of the library into the model using the library, user will get notified, that the library has changed (see chapter Updating E2E Libraries).

In the following dialog, you will create the library artifact. The Library Wizard will create the corresponding component automatically.

		The **Name** field is disabled. This is due to the fact, that the library name is equal to the libraries file name, just without the extension.
New E2E Library — Create New E2E Library. Enter name and documentation of the new E2E Library.		Editing the library name would only be possible by changing the actual model file name.
		Click **OK** to proceed.

New E2E Library

Create New E2E Library

Enter name and documentation of the new E2E Library.

`[Standard]` `JVM`

Name

librarySQLQuery

Version

1.0

Documentation

Please enter documentation here

[OK] [Cancel]

In other cases, e.g. if you want to add additional composites, you can open the above dialog by clicking **New** on **Composites** level of the tree.

E2E Library Wizard

E2E Librarysg

Create a new library artifact.

- librarySQLQuery Library
  - Components
    - Libraries
    - Backend Services
    - Proxy Services

E2E Library Components

E2E Library Artifacts

`>` `<` `New`

No E2E Library added.

[Next] [Skip] [Finish] [Cancel]

The **Create New Composite** dialog contains one more tab.

Using Java services in the composite service, you can specify the JVM options here.

- **JVM Options**: Java Virtual Machine options. The option strings has one of the following formats: "-D=...", "-X...", "_...", The system properties java.class.path and java.library.path are set by the model compiler and can not be overridden in the model.
- **killJVMThreadsOnShutdown**: If threads are still running on shutdown they are killed.
- **Parallel JVM Operations**: Number of parallel open JavaVM threads within the runtime. Default: 100. If the limit is reached, the runtime tries for 60 seconds to obtain a free JVM thread. If it does not succeed, an error (JAVAADLM/19) is thrown.

For more information on Java Components refer to Modeling the Java Components.

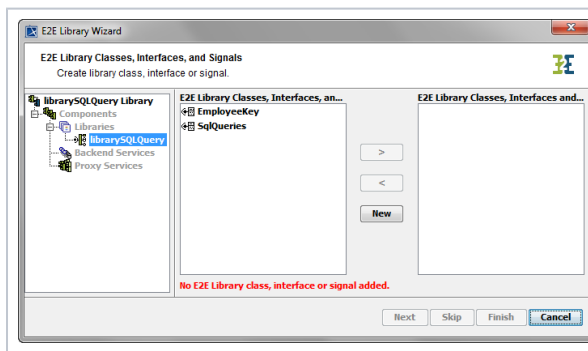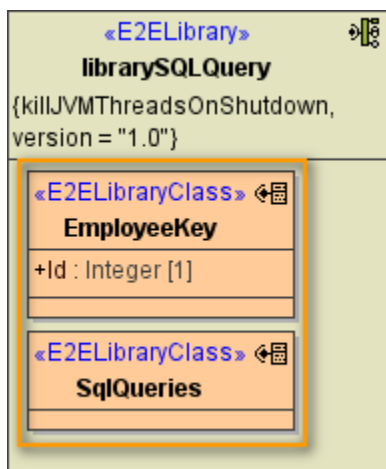Up to this point, you have defined the library component **librarySQLQuery**.
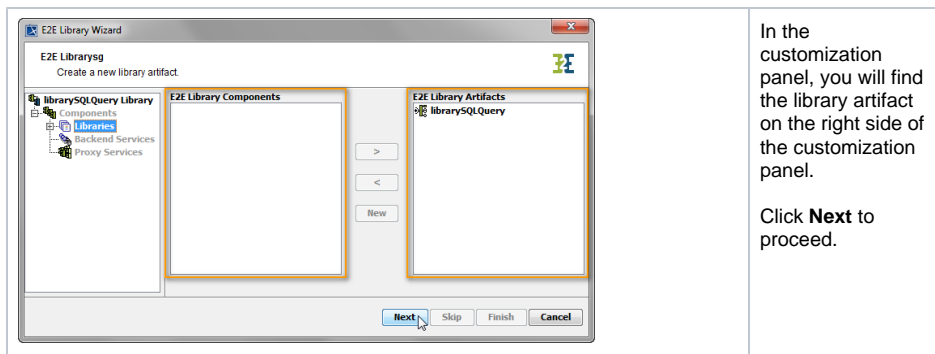
In the customization panel, you will find the library artifact on the right side of the customization panel.
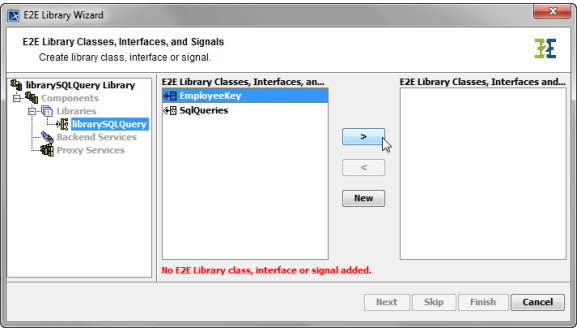
Click **Next** to proceed.

Defining Classes, Interfaces and Signals

In the next step, you will add the interface of the library.

The Library Wizard complains that definitions of classes, an interface or signal are missing. All buttons besides the **New** and the **Cancel** buttons are disabled.

Select the elements you want to be part of the library and click **>** .

Additionally, by clicking the **New** button you may add a JVM (Java Virtual Machine) component and artifact if needed. For more information on this refer to section Defining the Frontend Service which is part of the description of the Components Wizard.

Click **Next** to proceed.
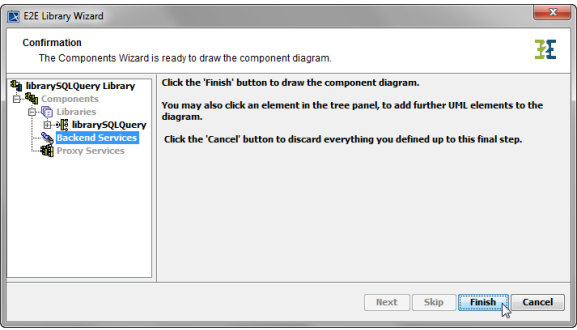
Defining Backend Aliases

Furthermore, the Library Wizard can be used to define backend access as already described for the Components Wizard. For more information on defining backends refer to Defining Backend Aliases of the description of the Components Wizard.
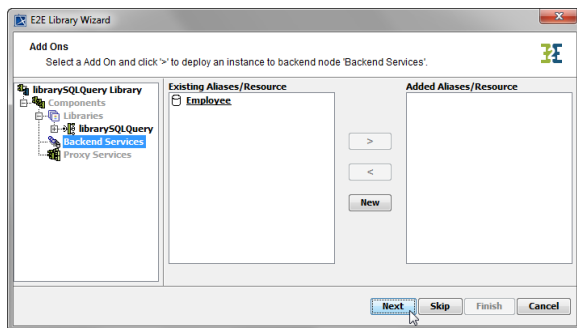
Click **Next** to proceed.

Finishing the Library Wizard

As a final step, you need to confirm to draw the component diagram. If the definitions are not complete yet, you may select an element node in the tree panel and add further UML elements to the diagram.

The Library Wizard generates the library component diagram. If no errors occurred, it is necessary to save the model. On saving, the deployment data of the E2E Model Compiler gets refreshed and you can proceed with e.g. compiling the model.