

Updating Applications Running on Application Servers

When running or debugging a Java EE or Web application, you can modify the source code and, almost immediately, see the result of your changes.

Basically, for the changes to take effect, there is no need to restart the application server. This feature is generally known as hot application deployment.

When updating a running application, you can specify what should be updated and how. This may be done [by editing the active run/debug configuration](#) or [right in the Run or Debug tool window](#).

- [Application update time](#)
- [Application update options](#)
- [Initiating application updates](#)
- [Specifying the application update options in the run/debug configuration](#)
- [Specifying the application update options in the Run or Debug tool window](#)
- [Turning the Update on frame deactivation option on or off in the Run or Debug tool window](#)

Application update time

The time taken to update a running application depends, first of all, on the type of the [application artifact](#) you are working with.

The updates are nearly instantaneous for exploded artifacts, when the application exists in the form of an (unpacked) directory structure. For this type of artifacts, resources (HTML, JSP, JavaScript, and image files) are updated practically in no time. A bit more time is needed to update classes (EJBs, servlets, etc.).

Packed artifacts (WARs, EARs, JARs, etc.) need a lot more time to be updated.

Application update options

The update options are different for exploded and packed artifacts.

For exploded artifacts, the available options are:

- **Update resources.** All changed resources (that is, all application components other than the classes) will be updated.
- **Update classes and resources.** All changed resources will be updated; changed classes will be recompiled.

In the [debug mode](#), the updated classes will be hot-swapped. In the run mode, IntelliJ IDEA will just update the changed classes in the output folder. Whether such classes will actually be reloaded in the running application, depends on the capabilities of the runtime being used.
- **Redeploy.** The application will be updated and redeployed.
- **Restart server.** The server will be restarted. The updated version of the application will be deployed at the server startup.

For packed artifacts, the available options are:

- **Hot swap classes.** Changed classes will be recompiled and reloaded at runtime. Note that this option will work only in the [debug mode](#).
- **Redeploy.** The overall application will be rebuilt and redeployed.
- **Restart server.** The server will be restarted. The application will be rebuilt and deployed at the server startup.

All the options with the exception of the **Restart server** are the examples of hot application deployment.

Depending on the application server, some of the update options listed above may be unavailable.

Initiating application updates

The application updates may be initiated in the following ways:

- By using the update function in the **Run** or the **Debug** tool window ( or `javaee.UpdateRunningApplication`).
- By switching from IntelliJ IDEA to a different application (for example, a Web browser). This way of initiating the application update is referred to as frame deactivation. The set of the update options in this case is, basically, the same except that the **Redeploy** and the **Restart server** options are not available. (You can also specify that no update actions are necessary on frame deactivation.)

The Update on frame deactivation option may be changed by [editing the active run/debug configuration](#). You can also [turn the corresponding update option on and off in the Run or Debug tool window](#) (.

Specifying the application update options in the run/debug configuration

1. While your application is running, [open the run/debug configuration](#) being used.
2. On the **Server** tab of the **Run/Debug Configurations** dialog, specify the application update settings:
 - **Perform on 'Update' action.** Select the required [update option](#) from the list.
 - **Show dialog.** Select this check box to be able to see and change the current update option in the **Update <application name>** dialog. This dialog will be shown prior to actually updating the running application.
 - **On frame deactivation.** Select the necessary update option from the list. (If no update actions are required, select **Do nothing**.)
3. Click **OK** in the **Run/Debug Configurations** dialog.
4. Now to update the running application, do one of the following:
 - Click **Update <application name> application**  on the toolbar of the **Run** or **Debug** tool window.
 - On the main menu, choose **Run | Update <application name> application**.
 - Press `javaee.UpdateRunningApplication`.

If you switch from IntelliJ IDEA to a different application, the update option you have selected in the **On frame deactivation** list will be used.

Specifying the application update options in the Run or Debug tool window

1. While your application is running, do one of the following:
 - Click **Update <application name> application**  on the toolbar of the **Run or Debug** tool window.
 - On the main menu, choose **Run | Update <application name> application**.
 - Press `javaee.UpdateRunningApplication`.
2. In the **Update <application name>** dialog that opens, select the necessary [update option](#). Additionally, if you don't want to see this dialog when updating your application, select the **Don't ask again** check box. Click **OK**.

Please note the following:

- If the **Update <application name>** dialog is not shown, you can enable it by turning on the **Show dialog** option in the [current run/debug configuration](#).
- If you select a different update option in the **Update <application name>** dialog, the current run/debug configuration will be changed accordingly.

Turning the Update on frame deactivation option on or off in the Run or Debug tool window

- While your application is running, click  on the toolbar of the [Deployment](#) tab.

If you now open the active run/debug configuration for editing, you'll see that the **On frame deactivation** setting has changed accordingly.

See Also

Concepts:

- [Running, Debugging and Testing](#)
- [Run/Debug Configuration](#)

Procedures:

- [Working with Application Servers](#)
- [Running](#)
- [Creating and Editing Run/Debug Configurations](#)
- [Debugging](#)
- [Java EE](#)

Reference:

- [Run/Debug Configurations](#)
- [Deployment Console](#)

Web Resources:

- [Developer Community](#) 