Run/Debug Configuration: OSGi Bundles

Use this dialog box to set up options for running and debugging applications that use OSGi Bundles.

The dialog box consists of the following tabs:

- Framework & Bundles
- Parameters
- Additional Framework Properties

Click here for the description of the options that are common for all run/debug configurations.

Framework & bundles tab

In this tab, select a framework that you need, compose a list of bundles to be installed, specify whether each bundle should be started upon installation, and define the order in which you want the bundles to be started.

Item	Description
OSGi Framework	From a drop-down list choose your framework.
Framework Start Level	Use this field to define a state of the execution in which a framework exists. You can modify the start level of the framework. This start level is used for managing the order of OSGi bundles' execution. If the bundle has a start level grater than the one for the framework, then it will be executed first.
Default Start Level	This field shows the default start level of bundles. In this case, the bundles are added after the framework's execution has started. You can modify the default start level for bundles.
Bundles Name	This read-only field shows the names of the OSGi bundles to be installed.
Start Level	In this text box, specify the default start level for newly installed bundles and thus determine the start order of bundles. The default value is 1. An OSGi system has a current level (called the active start level). If a bundle has a start level higher than the active start level it will not start when the OSGi system starts. The bundle will start as soon as the active start level reaches or exceeds the start level of the bundle. Accordingly, if the active start level becomes below the level of a bundle, the bundle will be shutdown.
Start After Install	Select this check box to have the selected bundle after installation.
Add	Click this button to open the **Select Bundles** dialog box, which displays all the currently available bundles. If the list is large, start typing the bundle name **Search** field - the contents of the list change as you type and show only the matching entries.
Remove	Click this button to delete the selected bundle from the list.

Parameters tab

In these tab, customize the framework run or debug procedure by specifying additional parameters.

Item	Description
VM Options	In this text box, specify the string to be passed to the Virtual Machine for launching the bundles. If the string is too long and does not fit in the text box, click  and type the desired string in the **VM Options** dialog. When specifying the options, follow these rules: ■ Use spaces to separate individual options, for example, `-client -ea -Xmx1024m`. ■ If an option includes spaces, enclose the spaces or the argument that contains the spaces in double quotes, for example, `some" "arg` or `"some arg"`. ■ If an option includes double quotes (e.g. as part of the argument), escape the double quotes by means of the backslashes, for example, `-Dmy.prop=\"quoted_value\"`. The `-classpath` option specified in this field overrides the classpath of the module.
Program Parameters	In this text box, type a list of arguments to be passed to the program in the format you would use in the command line. If the string is too long and does not fit in the text box, click the  button and type the desired arguments in the **Program Parameters** dialog box. Use the same rules as for specifying the VM options.
Use Alternative JRE	Select this check box to have the bundles run using a JRE which is different from the one used by the current project or module. Choose the location of the desired JRE from the drop-down list or click the **Browse** button  and choose the folder in the **Select Alternative JRE** dialog box, that opens.
Runtime Directory	Use this area to specify the runtime path of the framework. You can select from the following options: ■ **Recreate each time** - select this option if you want to have all previous information deleted from the directory before the execution of the framework. ■ **Use this directory** - use this field to specify the directory that the framework will use each time it executes. The execution time is faster. However, the directory might contain the unnecessary information such as artifacts from previous runs. Type the path to the desired folder manually or click the **Browse** button  and choose the folder in the dialog that opens.
Include All Bundles in Class Path	Select this check box to have all the selected bundles included in the classpath.

Additional framework properties

Item	Description
Debug Mode	Select this check box to enable debugging.
System Packages	In this text box, specify the system packages to be exposed inside the OSGi framework. Type the names of the packages using commas as separators. Wildcards are welcome.
Boot Delegation	In this text box, specify `java` packages for which the framework must delegate class loading to the boot class path. Type the names of the packages using commas as separators. Wildcards are welcome.
Start OSGi Console	Select this check box to run a prompt for the specified framework. For example, for the Equinox framework it is `osgi>`.
Run	Use this area to indicate what target you need to run. You can select from the following options: ■ **Just the bundles** - select this option if you need to run just the bundles. ■ **Product** - select this option if you need to run the product. ■ **Application** - select this option if you need to run the application.

Toolbar

Item	Shortcut	Description
	`Alt+Insert`	Click this button to add new configuration to the list.
	`Alt+Delete`	Click this button to remove the selected configuration from the list.
	`Ctrl+D`	Click this button to create a copy of the selected configuration.
	Edit defaults	Click this button to edit the default configuration templates. The defaults are used for the newly created configurations.
 or	`Alt+Up` or `Alt+Down`	Use these buttons to move the selected configuration or group of configurations (folder) up and down in the list. The order of configurations or folders in the list defines the order, in which configurations appear in the **Run/Debug** drop-down list on the main toolbar.
		Use this button to create a new folder. If one or more run/debug configurations have the focus, then the selected run/debug configurations are automatically moved to the newly created folder. If only a category has the focus, then an empty folder is created. Move run/debug configurations to a folder using drag-and-drop, or  buttons.

Common options

Item	Description
Name	In this text box, specify the name of the current run/debug configuration. This field does not appear for the default run/debug configurations.
Defaults	This node in the left-hand pane of the dialog box contains the default run/debug configuration settings. Select the desired configuration to change its default settings in the right-hand pane. The defaults are applied to all newly created run/debug configurations.
Share	Select this check box to make the run/debug configuration available to other team members. If the directory-based project format is used, the settings for a run/debug configuration are stored in a separate xml file in the `.idea\runConfigurations` folder if the run/debug configuration is shared and in the `.idea\workspace.xml` file otherwise. If the file-based format is used, the settings are stored in the `.ipr` file for shared configurations or in the `.iws` file for the ones that are not shared. This check box is not available when editing the run/debug configuration defaults.

Item	Description
Before launch	Specify which tasks should be carried out before starting the run/debug configuration. The specified tasks are performed in the order that they appear in the list. **Item** | **Keyboard shortcut** | **Description**

Item	Item	Keyboard shortcut	Description	Description
	+	Alt+Insert	Click this icon to add a task to the list. Select the task to be added: ▪ **Run External tool.** Select this option to run an application which is external to IntelliJ IDEA. In the dialog that opens, select the application or applications that should be run. If the necessary application is not defined in IntelliJ IDEA yet, add its definition. For more information, see Configuring Third-Party Tools and External Tools. ▪ **Make.** Select this option to have the project or module compiled. The Make Module command will be carried out if a particular module is specified in the run/debug configuration, and the Make Project command otherwise. If an error occurs during the compilation, IntelliJ IDEA won't attempt to start the run/debug configuration. ▪ **Make, no error check.** The same as the **Make** option but IntelliJ IDEA will try to start the run/debug configuration irrespective of the compilation result. ▪ **Build Artifacts.** Select this option to have an artifact or artifacts built. In the dialog that opens, select the artifact or artifacts that should be built. See also, Configuring Artifacts. ▪ **Run Ant target.** Select this option to have an Ant target run. In the dialog that opens, select the target to be run. For more information, see Ant. ▪ **Generate CoffeeScript Source Maps.** Select this option to have the source maps for your CoffeeScript sources generated. In the dialog that opens, specify where your CoffeeScript source files are located. For more information, see CoffeeScript Support. ▪ **Run Maven Goal.** Select this option to have a Maven goal run. In the dialog that opens, select the goal to be run. For more information, see Maven. ▪ **Run Remote External tool**: Add a remote SSH external tool. Refer to the section Remote SSH External Tools for details.	

See Also

Reference:

- OSGi

External Links:

- Osmorc

Web Resources:

- Developer Community