

KIELER Language Server Development

- [Starting the LS \(debug mode\)](#)
- [Problems](#)
 - [My LS does not show the connection message](#)
 - [Your address is already in use](#)
 - [Some binding error or NPE when connecting via a client application](#)
- [Adding a LS Extension](#)

The KIELER language server (KIELER LS) uses Xtext to generate language features and to provide a language server. Sprotty is used as the underlying diagram server implementation. On the client-side is some kind of application, which communicates using an extended language server protocol (LSP). This guide is for developers who want to use the KIELER LS or KlighD Diagram Server (names are not final) or extend it and connect some kind of application to it (we give a Theia, VSCode, and standalone web application example).

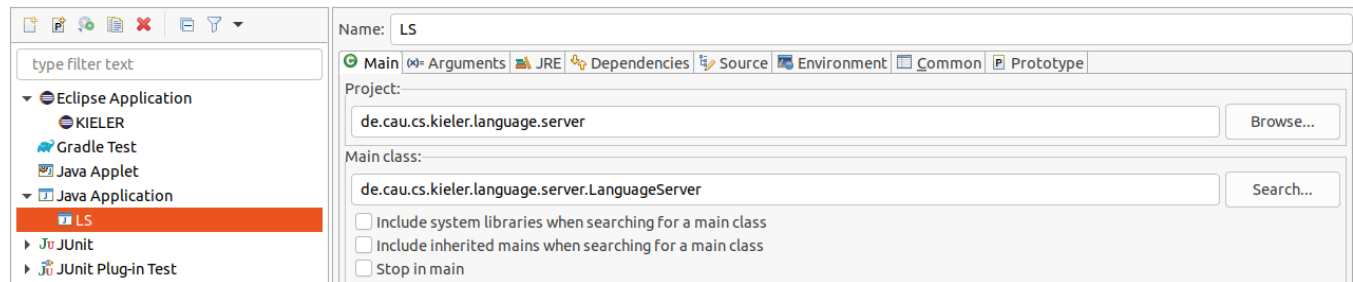
Starting the LS (debug mode)

We assume that you are using Eclipse and used one of our or your Oomph setups to configure yourself a nice development environment with all sources inside.

We further assume that you are using some class to start your language server, let's call it `de.cau.cs.kieler.language.server.LanguageServer` and it resides the `de.cau.cs.kieler.language.server` project (as it is currently the case for the semantics language server in the KIELER project).

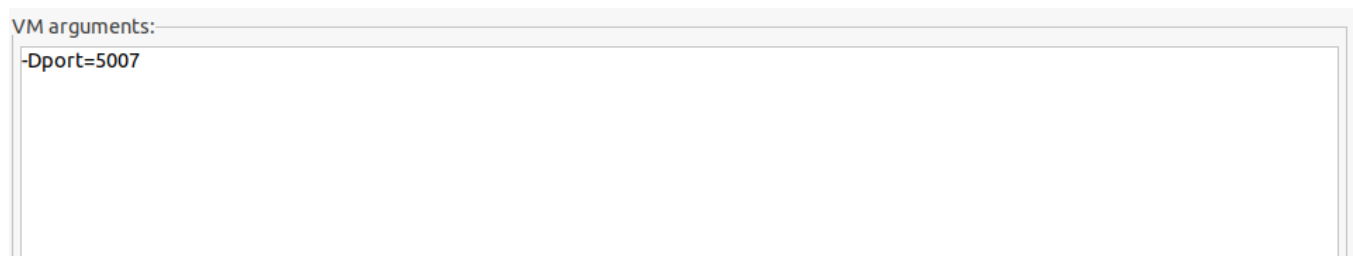
To start your language server in debug mode you need to start some client-side application that will speak to your language server. Since you want to debug your LS you cannot connect via stdin/out but must connect via socket. For the sake of this guide, we will use the socket `5007` as your LSP socket. If you are using Theia or some kind of standalone application do not confuse this socket with the socket your application runs on.

Create a new debug configuration for a Java Application in Eclipse:



Next, you need to specify your arguments to connect via socket and include all dependencies.

Add `-Dport=5007` as a VM argument (in the arguments tab). This will be used by the `AbstractLanguageServer` to establish a connection (the specific class might change).



Include all dependencies by either adding all relevant projects and their folders to the classpath or, which might be easier if it does not cause any problems, add all projects and their folders to your classpath. You can add folders to your classpath by clicking on the classpath, selecting `Advanced...` and selecting `Folder` (currently the case for Eclipse 2021-03 and below).

The following is printed in the console when the LS starts.

```
Connection to: localhost:5007
Starting language server socket
```

Problems

My LS does not show the connection message

```
Connection to: localhost:5007
Starting language server socket
```

You might have forgotten to set the correct VM argument to specify a port or the class that handles the connection is somehow not included or broken. Search whether the string `System.getProperty("port")` is somehow used in your application.

Your address is already in use

```
Connection to: localhost:5007
Starting language server socket
Exception in thread "Thread-0" java.net.BindException: Address already in use
    at java.base/sun.nio.ch.Net.bind0(Native Method)
    at java.base/sun.nio.ch.Net.bind(Net.java:455)
    at java.base/sun.nio.ch.Net.bind(Net.java:447)
    at java.base/sun.nio.ch.AsynchronousServerSocketChannelImpl.bind(AsynchronousServerSocketChannelImpl.
java:164)
    at java.base/java.nio.channels.AsynchronousServerSocketChannel.bind(AsynchronousServerSocketChannel.
java:198)
    at de.cau.cs.kieler.klighd.lsp.launch.AbstractLanguageServer.run(AbstractLanguageServer.java:172)
    at de.cau.cs.kieler.klighd.lsp.launch.AbstractLanguageServer.lambda$0(AbstractLanguageServer.java:85)
    at java.base/java.lang.Thread.run(Thread.java:829)
```

Some other LS you started previously is using this connection. Maybe you configured a watcher to have a better look at the LSP messages. Either way, something is running on your port. Find it and kill it or change port.

You can kill it via `fuser -k 5007/tcp`

Some binding error or NPE when connecting via a client application

You might miss some projects/folders in your classpath

Adding a LS Extension

When creating a language extension as a new plug-in, the run configuration must be updated. Under the dependencies tab of the run configuration the classpath entries must be selected. Now select "Add Projects.." and add the new plug-in. Additionally, click on "Advanced.." and select "Add Folders". Now select the new plug-in again.