

JavaScriptDebuggerOverallFunctionalSpec

Table of Contents »

The NetBeans JavaScript Debugger

Introduction

- Installing JavaScript Debugger functionality in NetBeans 6.1

- Enabling JavaScript Debugging

 - Enabling the JavaScript Debugger launch

 - HTML and JavaScript Editor integration

- Invocation of JavaScript Debugger

 - Installation of NetBeans JavaScript Debugger Firefox extension

 - NetBeans JavaScript Debugger Firefox extension launch

 - Debugging

 - Termination of debugging session

Architecture

Technology Preview

Proposal for module structure in <http://hg.netbeans.org/main>

- The following two modules remain in:

 - j2ee cluster

 - gsf cluster

 - Basically NetBeans aware NbJS layer:

 - Basically NetBeans agnostic JS layer (all modules below):

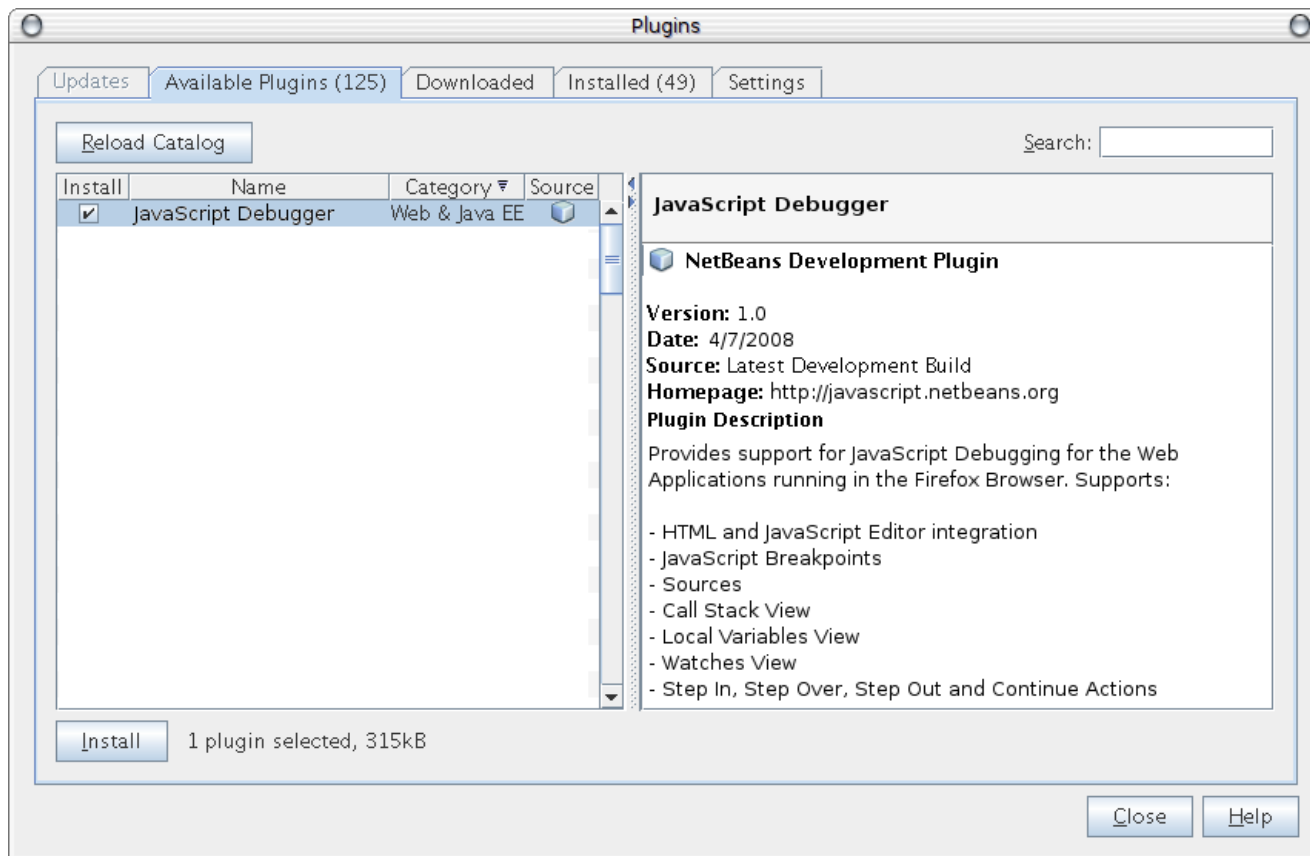
The NetBeans JavaScript Debugger

Introduction

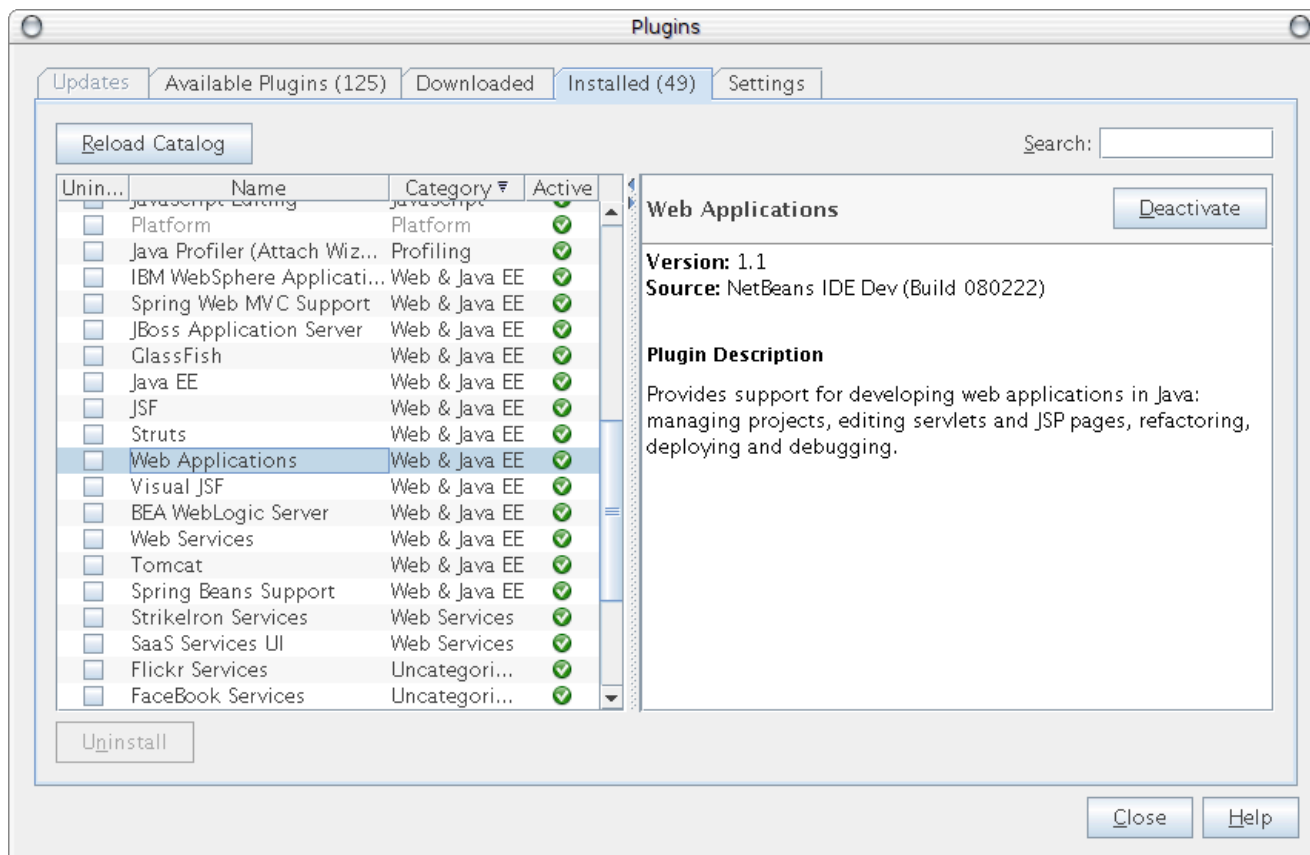
This document describes the overall functionality of the JavaScript debugger and its integration with the NetBeans IDE - specifically the NetBeans Web project, as it will be implemented for the Milestone one (JAVAONE2008 Technology Preview). The detailed functional spec of the debugger windows is [here](#). This document discusses the NetBeans Web project integration strategy.

Installing JavaScript Debugger functionality in NetBeans 6.1

The JavaScript Debugger functionality will be available as Technology Preview for JavaOne 2008. As such the JavaScript Debugger modules will be available on the NetBeans 6.1 Beta Update Center. The JavaScript Debugger modules will be implemented on top of the NetBeans 6.1 code base. Thus to access the JavaScript Debugger functionality the user will have to connect to the NetBeans 6.1 Beta Update center, download the JavaScript Debugger (Technology Preview) Kit (set of modules that implement the JavaScript Debugger) and install them in the NetBeans 6.1 IDE.



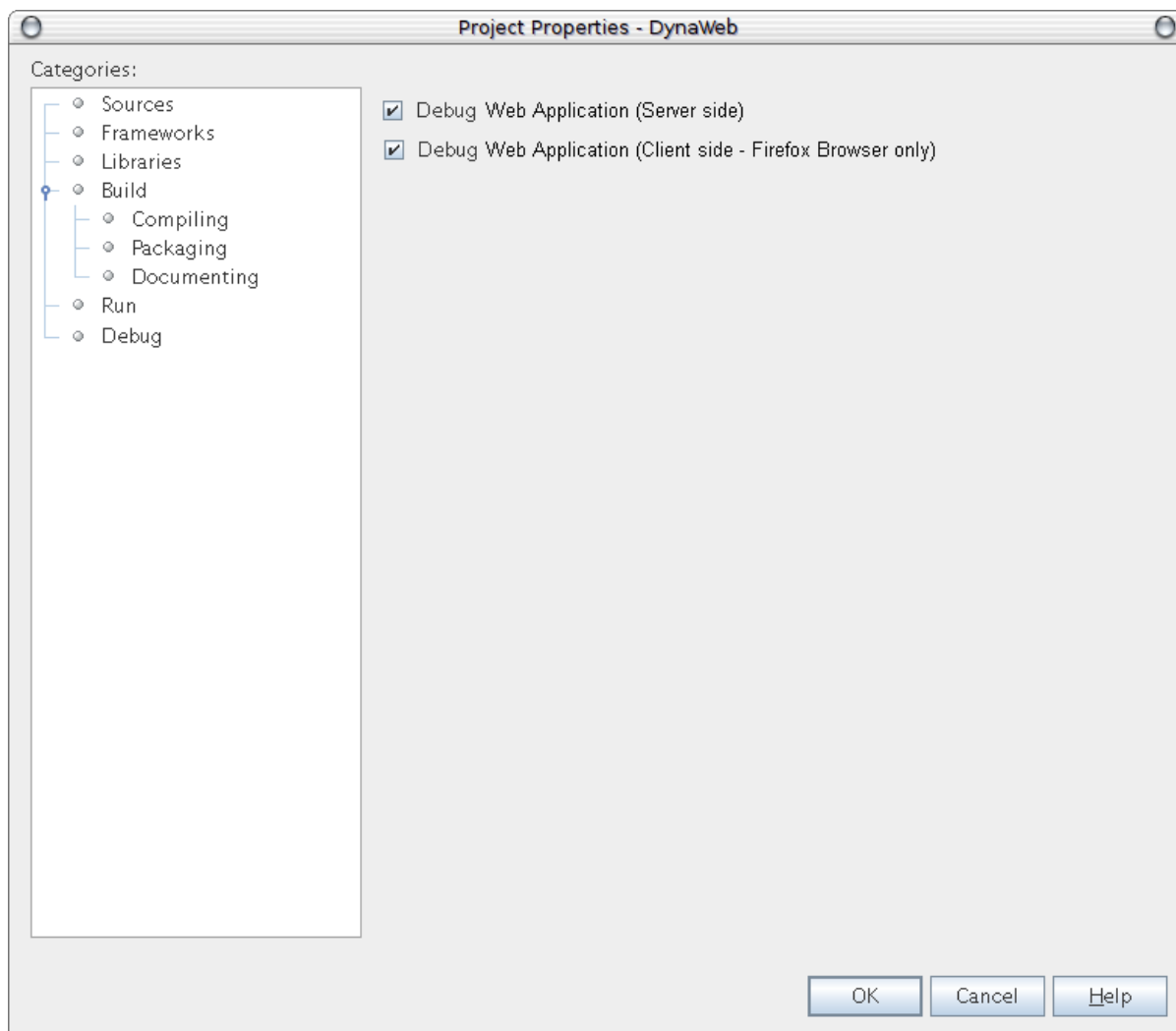
The installation of Web Applications Kit + standalone and embedded JavaScript editing support) is a prerequisite for JavaScript Debugger functionality.



Enabling JavaScript Debugging

Enabling the JavaScript Debugger launch

Once the JavaScript Debugger modules are installed, a new `Debug Web Application (Client Side - Firefox Browser only)` checkbox will appear in the new **Debug** category of the **Project Properties** dialog of NetBeans Web Application project. The JavaScript Debugging can be enabled by selecting the `Debug Web Application (Client Side - Firefox Browser only)` checkbox.



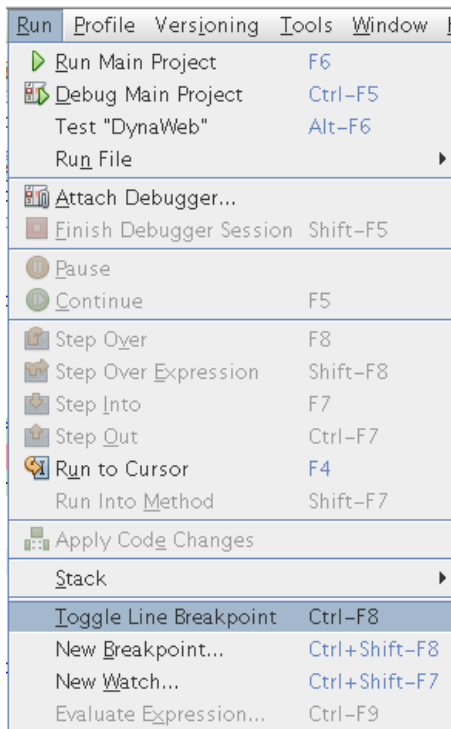
The `Debug Web Application (Server Side)` checkbox controls whether the server side of the web application is run in debug mode or not. If this checkbox is unselected the web application is simply run in the server. This provides independent control over whether to launch server and client side debugging.

The rationale for this design is:

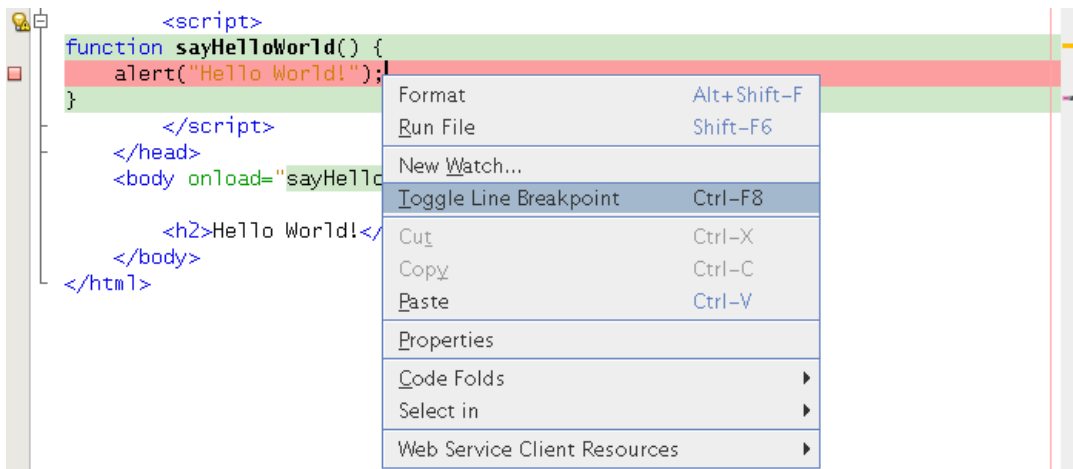
- Preserve the one-click Debug functionality of the NetBeans Web projects. In other words the user is not prompted (thus interrupting one-click Debug flow) for whether to launch the JavaScript Debugger or not.
- Provide the per-project control over the launching of JavaScript Debugger.

HTML and JavaScript Editor integration

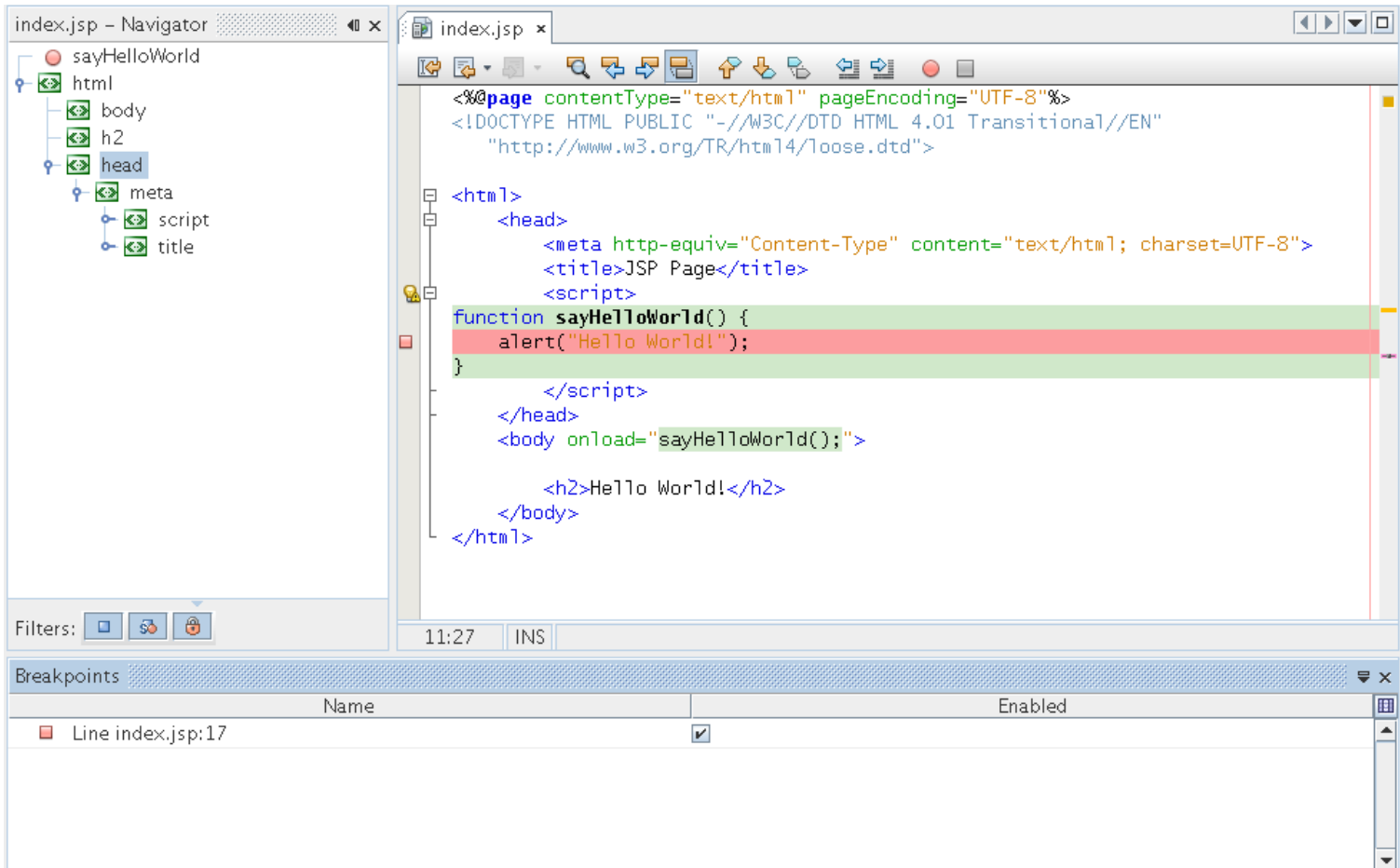
Once the JavaScript Debugger modules are installed, the user will be able to set/unset the breakpoints in the JavaScript functions using the **Run:Toggle Line Breakpoint (Ctrl-F8)** action in the menu bar:



or the **Toggle Line Breakpoint (Ctrl-F8)** action in the HTML or JavaScript editor context menu:



The breakpoint will be shown in the HTML and JavaScript editor's left side gutter using the standard breakpoint icon.



The JavaScript Debugger primarily works with the HTML and JavaScript code as seen by the browser. In case of pure HTML pages and JavaScript files that are sent to the browser verbatim it is possible to map the HTML and JavaScript source seen by the browser to the corresponding artifacts in the NetBeans Web project. In such cases the user will work with the source editor of those project artifacts for setting breakpoints etc. Similarly the source editor for the project artifact will be used to show the current call stack frame annotation as the user steps through the JavaScript code. However, in certain cases, such as web page resulting from a JSF page, it is not always possible to map the HTML seen by the browser to the JSF page source artifact in the project. In such cases the content of the HTML page as seen by the browser will be loaded in a virtual buffer which will be opened in the HTML editor. The user will use this HTML editor window to set breakpoints or while stepping through the code. The breakpoints set in such a buffer will be expressed in terms of the corresponding URL as seen by the browser and will appear as such in the **Breakpoints** window. Similarly the **Sources** window will show the URL as seen by the browser.

The breakpoint will also be listed in the **Breakpoints** window. The exact details of how the breakpoint will appear in the breakpoints window please refer to the [JavaScript Debugger Windows Functional Specs](#).

Invocation of JavaScript Debugger

The JavaScript Debugger will be launched - if the **Debug Web Application (Client Side - Firefox Browser only)** checkbox in the **Debug** category of the **Project Properties** dialog of NetBeans Web Application project is selected and the user invokes

Debug Web Application or **Debug** file action.

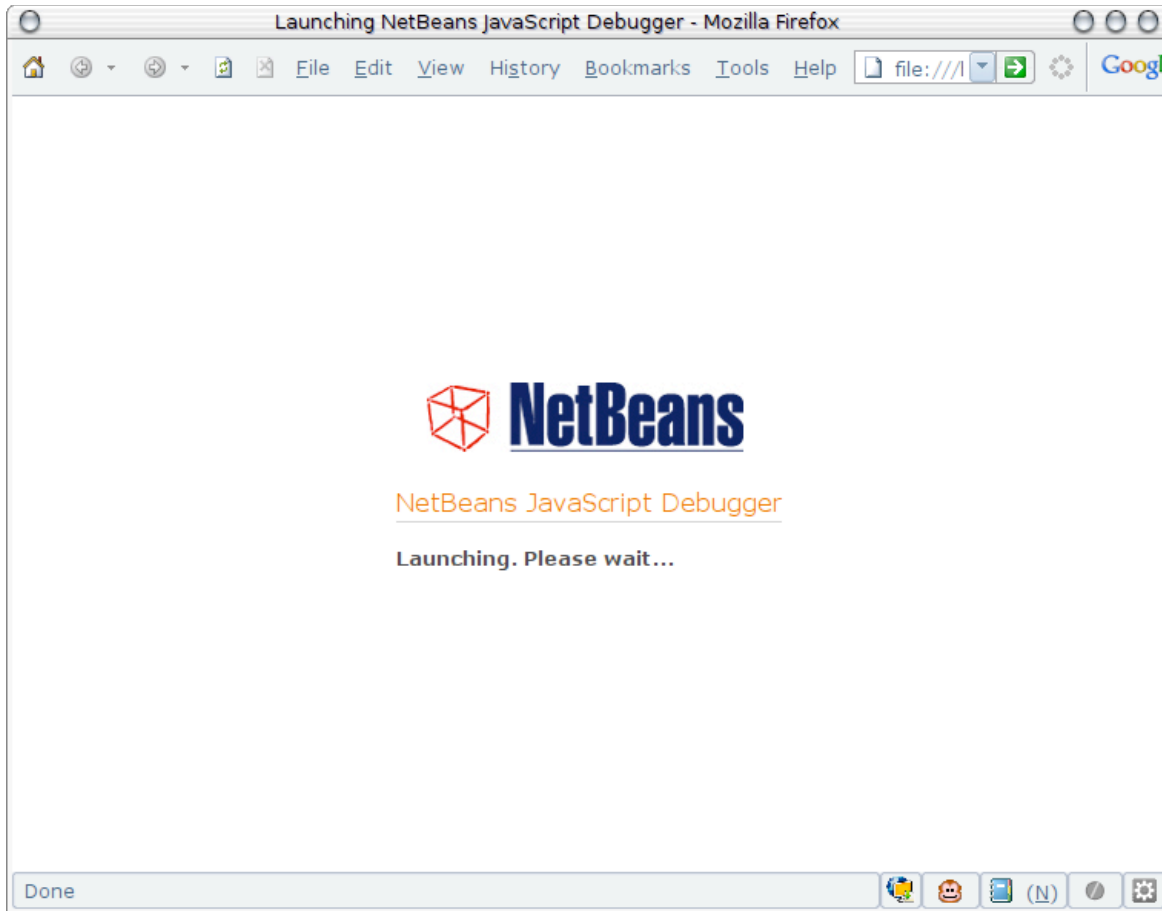
If the user invokes the **Debug** action and the JavaScript debugger is also launched, then the user will see two sessions in the **Sessions** window. The server side and JavaScript Debugger sessions will be clearly identified in the sessions window. Normally the **Sessions** window is not automatically shown by the NetBeans debugger infrastructure when the debugging starts. However in case of above mentioned two debug sessions scenario the **Sessions** window will be shown. The user will be able to switch the focus of other debugging windows (such as **Sources**, **Threads**, **Call stack**, **Local Variables**, **Watches** windows) between the server side and the JavaScript debugger by making the server side or the JavaScript debugging session the *current* session in the **Sessions** window respectively.

Installation of NetBeans JavaScript Debugger Firefox extension

In the Technology Preview the JavaScript debugging will be supported using the the desktop Firefox browser only. This functionality is implemented using a Firefox add-on that is required to be installed in the user's Firefox profile. When the JavaScript Debugger launches it first checks if the NetBeans JavaScript Debugger Firefox add-on is installed. If the add-on is not detected the user is informed and with the user's acknowledgment the add-on is installed in the Firefox profile and as required by the add-on installation mechanism of the Firefox browser - Firefox is restarted to make the add-on operational. Once the NetBeans JavaScript Debugger Firefox add-on is in place the JavaScript debugging can proceed.

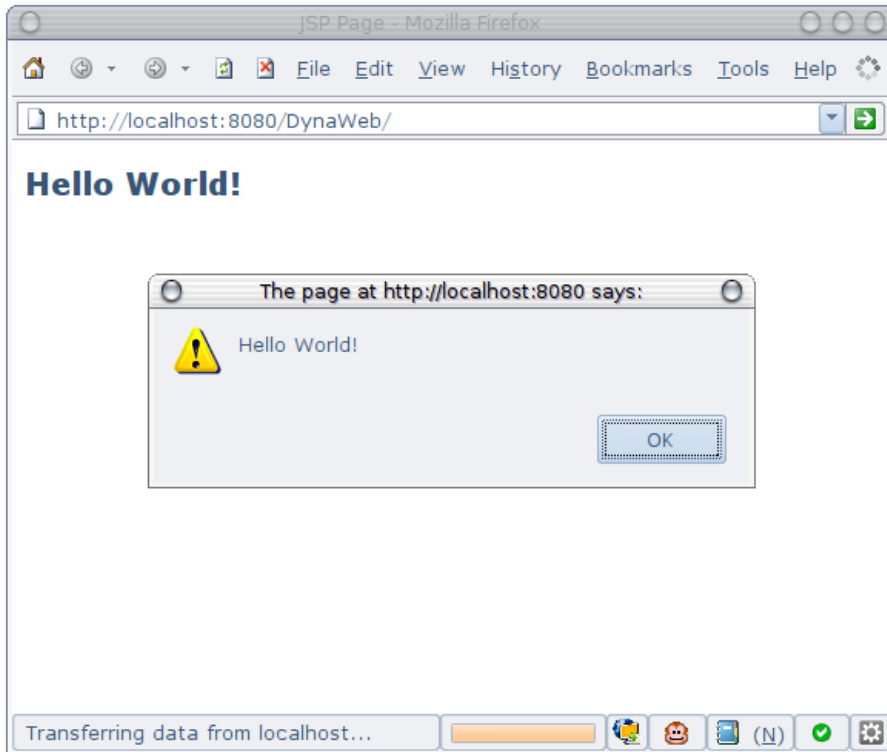
NetBeans JavaScript Debugger Firefox extension launch

When the JavaScript debugger is invoked it launches the Firefox browser and communicates to the NetBeans JavaScript Debugger Firefox add-on the information about the JavaScript Debugging session initiation. The add-on displays the following splash screen (page) while the initial handshake with the NetBeans JavaScript Debugger is in progress.



Debugging

During the handshake the JavaScript Debugger does the version checking, configures the initial breakpoints and other debugging options. Once the initial handshake completes, the JavaScript debugger send the URL of the web page to be debugged to the add-on. The add-on launched the URL in the browser and attaches to the JavaScript debugging context of the page.



Once the page to be debugged is loaded the add-on attempts to set the initial breakpoints. Some may work some may not. It sends the information about the initial window/frameset/frames/iframe structure (used to populate the **Threads** window), loaded HTML sources and external JavaScript scripts (used to populate the **Sources** window) and the successfully set breakpoints (used to populate the **Breakpoints** window) back to the IDE. At this point the JavaScript Debugger windows become visible to the user. Now the user can interact with the web page in the browser. This may result in triggering of the breakpoints set the user. The add-on detects the triggering of the breakpoint (and after evaluating the breakpoint condition to true) and suspends the execution in the browser and sends the breakpoint event to the JavaScript Debugger in the NetBeans IDE. The JavaScript debugger opens the source editor of the place where the breakpoint occurred, selects the breakpoints location in it and shows the *current* stack frame annotation. It also populates the **Thread**, **Call Stack**, **Local Variables** windows. It also tries to evaluate all the watch expressions in the **Watches** window in the context of current call stack frame and shows the evaluated values in the **Watches** window. At this point the user may use the **Step Over**, **Step In**, **Step Out** or **Continue** actions. Based on the user action the debugging session proceeds accordingly.

Termination of debugging session

The debugging session may terminate in the following ways:

- When the user explicitly invokes the **Finish** action on the JavaScript debugging session. In this case the add-on is simply detached from the debugging context and the user can proceed with normal interaction with the web page without any interruption of breakpoints etc.
- The user closes the browser window (abrupt termination).

Architecture

[Components/Layers](#)

Technology Preview

[JavaScriptDebuggerTechnologyPreviewUserGuide](#)

Proposal for module structure in
<http://hg.netbeans.org/main>

The following two modules remain in:

j2ee cluster

- **web.client.javascript.debugger.kit**
- **web.client.javascript.debugger.ant**

gsf cluster

Basically NetBeans aware NbJS layer:













- **web.client.javascript.debugger.api**
 - /src/org/netbeans/modules/web/client/javascript/debugger/filesystem
 - /src/org/netbeans/modules/web/client/javascript/debugger/ui
 - /src/org/netbeans/modules/web/client/javascript/debugger/ui/breakpoints
 - /src/org/netbeans/modules/web/client/javascript/debugger/ui/breakpoints/customizer
 - /src/org/netbeans/modules/web/client/javascript/debugger/ui/resources
 - /src/org/netbeans/modules/web/client/javascript/debugger/ui/options
 - /src/org/netbeans/modules/web/client/javascript/debugger/actions
 - /src/org/netbeans/modules/web/client/javascript/debugger/api
 - /src/org/netbeans/modules/web/client/javascript/debugger/models
 - /src/META-INF
 - /src/META-INF/services
 - /src/META-INF/debugger
 - /src/META-INF/debugger/SessionsView
 - /src/META-INF/debugger/netbeans-javascript-DebuggerInfo
 - /src/META-INF/debugger/BreakpointsView
 - /src/META-INF/debugger/netbeans-javascript-Session
 - /src/META-INF/debugger/netbeans-javascript-Session/LocalsView
 - /src/META-INF/debugger/netbeans-javascript-Session/SourcesView
 - /src/META-INF/debugger/netbeans-javascript-Session/CallStackView
 - /src/META-INF/debugger/netbeans-javascript-Session/ThreadsView
 - /src/META-INF/debugger/netbeans-javascript-Session/WatchesView

Basically NetBeans agnostic JS layer (all modules below):

- **web.client.javascript.debugger.js.api**
 - /src/org/netbeans/modules/web/client/javascript/debugger/js
 - /src/org/netbeans/modules/web/client/javascript/debugger/js/api

- /src/org/netbeans/modules/web/client/javascript/debugger/js/spi
- /src/org/netbeans/modules/web/client/javascript/debugger/js/nondebugger
- **web.client.javascript.debugger.base depends on web.client.javascript.debugger.js.api**
 - /src/org/netbeans/modules/web/client/javascript/debugger/js/api/impl drop impl
 - /src/org/netbeans/modules/web/client/javascript/debugger/dbgp
- **web.client.javascript.debugger.firefox** depends on **web.client.javascript.debugger.js.api**, **web.client.javascript.debugger.js.base** and requires token **netbeans-firefox-extension**
 - /release/modules/ext
 - /src/org/netbeans/modules/web/client/javascript/debugger/firefox
- **web.client.javascript.debugger.firefox.extension_ provides token netbeans-firefox-extension**
 - /extensions/firefox/netbeans-firefox-extension
 - /extensions/firefox/netbeans-firefox-extension/skin
 - /extensions/firefox/netbeans-firefox-extension/locale
 - /extensions/firefox/netbeans-firefox-extension/locale/en-US
 - /extensions/firefox/netbeans-firefox-extension/content
- **web.client.javascript.debugger.ie** depends on **web.client.javascript.debugger.js.api**, **web.client.javascript.debugger.js.base** and requires token **netbeans-ie-extension**
- **web.client.javascript.debugger.ie.extension** provides token **netbeans-ie-extension**
 - /release/lib (Primarily contains the IE extension DLL containing the Browser Helper Object)

Attachments

JavaScriptDebuggerTechnologyPreviewKitPlugins.png	 40478 bytes
NetBeansJavaScriptDebuggerFirefoxAddonSplashScreen.png	 46053 bytes
ProjectPropertiesEnableJavaScriptDebugging.png	 14864 bytes
RequiredWebApplicationsPlugins.png	 38634 bytes
RunToggleLineBreakpointAction.png	 14329 bytes
ToggleLineBreakpointPopupMenuAction.png	 8562 bytes
breakpoint.png	 28409 bytes
components.odg	 28992 bytes
components.pdf	 23423 bytes
index.png	 34082 bytes
netbugs.gif	 297813 bytes
netbugs.png	 14046 bytes