Red Hat JBoss Developer Studio 8.0

8.0.0 Release Notes

Information about this release of JBoss Developer Studio

Red Hat JBoss Developer Studio 8.0 8.0.0 Release Notes

Information about this release of JBoss Developer Studio

Red Hat Customer Content Services

Legal Notice

Abstract

These release notes contain important information related to JBoss Developer Studio. New features, resolved issues and known issues of this product release are detailed here.

Table of Contents

1. INTRODUCTION TO RED HAT JBOSS DEVELOPER STUDIO

1.1. About Red Hat JBoss Developer Studio

Red Hat JBoss Developer Studio is a set of Eclipse-based development tools. It contains plug-ins that integrate with Eclipse to extend the existing functionality of the integrated development environment (IDE).

JBoss Developer Studio is designed to increase your productivity when developing applications. You can focus on building, testing and deploying your applications because JBoss application development tools are integrated in one IDE. Furthermore, JBoss Developer Studio can assist your application development with its unique features in the following ways:

- Develop new applications using the wizards and project examples of JBoss Central

- Add powerful functionality to applications with minimal effort using Forge Tools

- Build web interfaces with ease using the visual editing and drag-and-drop utilities of Visual Web Tools and Mobile Web Tools

- Experience browsers automatically refreshing in response to modified application resources with LiveReload Tools

- Incorporate JSF, Seam, JAX-RS, Hibernate, CDI and other popular APIs into applications with simplicity using the tool-driven interface

- Preview and test mobile web applications on a variety of simulation mobile devices using BrowserSim

- Create, build and test Cordova-based hybrid mobile applications for iOS and Android platforms using Hybrid Mobile Tools and CordovaSim

- Deploy applications to JBoss runtime servers and the cloud using JBoss Server Tools and OpenShift Tools

JBoss Developer Studio comes built around Eclipse and packaged with all of the necessary dependencies and third-party plug-ins for simplified installing. For developers already running Eclipse, JBoss Developer Studio can also be installed through Eclipse Marketplace. For distinction, this latter JBoss Developer Studio installation is called JBoss Developer Studio BYOE (Bring Your Own Eclipse).

Report a bug

1.2. Use Cases of JBoss Developer Studio

JBoss Developer Studio assists Java EE developers by integrating JBoss technology and APIs in a single development environment. Here are a few ways that JBoss Developer Studio makes the work of a developer easier:

Web applications

JBoss Central provides wizards that generate skeleton and sample projects, enabling you to focus on developing the functionality of your applications. The wizards create web applications based on

different APIs and technologies, showing the usage and advantages of each. JBoss Developer Studio also offers project file templates in a range of popular programming languages, including HTML, XHTML, and JSF.

Palettes in JBoss Developer Studio give access to the core elements of the JSF, RichFaces and Seam APIs, for use in developing the user interfaces of your applications. Elements of these APIs can be dragged and dropped directly into your project so that you can create richer user interfaces quickly. Visual Web Tools offers graphical and source viewing of files and defaults to dedicated editors for different file types. JBoss Developer Studio supports the Java EE specification and provides tools for JAX-RS, Hibernate, and CDI APIs so you can develop the server-side components of your applications effortlessly.

LiveReload Tools automatically refreshes browsers of local or deployed applications as you modify project resources to save you from needing to manually refresh. You can even experience automatic refreshing when viewing applications in browsers on external and mobile devices, with application web addresses easy to navigate to with QR codes.

Web applications optimized for mobile devices

Mobile Web Tools provides support for HTML5 and jQuery Mobile to enable you to create web applications optimized across desktop and mobile clients. The HTML5 Project wizard in JBoss Central generates a sample application using HTML5 and jQuery Mobile technologies and, together with HTML5 and jQuery Mobile project file templates, helps you to get up and running with these APIs and technologies quickly. HTML5 and jQuery Mobile widgets can be dragged from the jQuery Mobile palette into your project files and, in conjunction with the widget wizards, enable you to effortlessly develop customized user interfaces for your mobile web applications.

BrowserSim allows you to view your web applications on a variety of simulated mobile devices so that you can ensure they will be correctly formatted. LiveReload also extends to BrowserSim allowing you to experience automatic browser refreshing as you develop your mobile web applications. The integration of Firebug Lite and Weinre capabilities with BrowserSim assists you to inspect the page source of web pages with familiar tools.

Hybrid mobile applications

Hybrid Mobile Tools provides support for developing and building Cordova-based hybrid mobile applications for iOS and Android platforms. The Hybrid Mobile application wizard assists you to quickly generate new projects, while the Cordova Configuration Editor and Cordova Plug-in Discovery wizard help you to efficiently modify the capabilities of your projects. Hybrid Mobile Tools provides actions that simplify your workflow, for example calling your system installed Android and iOS SDKs from within the IDE to emulate or run your hybrid mobile applications. With wizards to export workspace projects to Cordova-enabled native projects or ready-to-sign applications, you can quickly be ready to share your hybrid mobile projects and applications.

CordovaSim enables you to view and test your hybrid mobile applications on Android and iOS simulated mobile devices so that you can ensure they look and work as expected. You can interact with your mobile applications through BrowserSim and use the device input panel to provide sample data to your applications for device functions like cameras and accelerometers. An advantage of CordovaSim is that it does not require native SDKs to be installed on your system, unlike native SDK emulators. Additionally, by teaming the device control panel with BrowserSim, you get all of the great functionality of BrowserSim, such as skins and LiveReload, while simulating your hybrid mobile applications.

Applications for cloud deployment

OpenShift Tools deploys your applications directly to the cloud on the Red Hat OpenShift platform. You can create and manage your OpenShift account and manage the deployment of applications to OpenShift within the IDE. In addition to using the OpenShift Application wizard to create and deploy

new OpenShift applications, OpenShift Tools can import applications already deployed on OpenShift so that you can further develop them and manage their deployment from the comfort of the IDE.

Report a bug

2. ABOUT THIS RELEASE

JBoss Developer Studio 8.0.0 is an upgrade of JBoss Developer Studio 7. The upgrade has three purposes:

- It supports Eclipse Luna (4.4) and contains an Eclipse Luna update, SR1.

- It contains new features for the existing tools.

- It introduces development with AngularJS (Early Access) and Ionic (Technology Preview) frameworks through plug-ins available to install in JBoss Central.

- It resolves issues identified in earlier versions of JBoss Developer Studio.

For more information about operating systems, chip architectures and Java developer kits supported by this release, see https://access.redhat.com/site/articles/427493 on the Red Hat Customer Portal.

Report a bug

3. NEW FEATURES

This release of JBoss Developer Studio includes new features to enhance and extend its functionality as detailed here:

Installation

- Red Hat JBoss Enterprise Application Platform 6.3.0 is included in the combined JBoss Developer Studio and JBoss EAP installer.

BrowserSim

- JavaFX web engine is available as an alternative to SWT WebKit for running BrowserSim when Oracle JDK is available. Note that use of Oracle JDK with JBoss Developer Studio is not supported by Red Hat for all operating systems; for more information, see https://access.redhat.com/site/articles/427493 on the Red Hat Customer Portal.

- Chrome Dev Tools Debugger is available for use with BrowserSim. This feature is offered with the JavaFX web engine only at this time.

- Process logs, error messages and JavaScript console function output are listed in the existing IDE `Console` view.

- BrowserSim (and CordovaSim) can be stopped and closed from the `Console` view.

CDI Tools

- DeltaSpike partial beans are recognized by CDI Tools. This extends CDI Tools features such as validation, code completion, hyperlinks and refactoring to DeltaSpike partial beans.

- CDI 1.1 projects are now supported. The CDI version of a project is automatically detected by CDI Tools when CDI 1.1 `.jar` files are included in the classpath.

Forge Tools

- The embedded Forge 1 runtime server is version 1.4.4.Final. Forge 1.x runtime server is feature-frozen and installation of other Forge 1.x runtime servers is disabled.

- Forge 2 runtime and command line interface are included as part of the core JBoss Developer Studio distribution, with Forge 2 set as the default Forge in new JBoss Developer Studio installations.

  - The embedded Forge 2 runtime server is version 2.12.1.Final.

  - The Forge 2 command line interface is accessed through the **Forge Console** view, along with Forge 1.

  - Forge 2 has new project commands, many of which are also available as wizards. The available commands and wizards are responsive, changing based on the selected project item.

- **Forge Console** view is part of JBoss perspective by default and the default Forge runtime starts on pressing `Ctrl+4`.

- New Forge console commands `cd #` and `cd #/PROJECT_NAME` enable navigation to workspace locations and existing projects inside the current workspace.

- More actions are available in **Forge Console** view:

  - A **Clear Console** button to clear the current output on the Forge command line

  - A **Go To Selection** button to navigate to project resources selected in the **Project Explorer** view on the Forge command line

  - A **Link with Editor** button to automatically navigate to project resources opened for editing from the **Project Explorer** view on the Forge command line

Hybrid Mobile Tools and CordovaSim

- Multiple Cordova versions are now supported, both for developing hybrid mobile applications and for testing these applications using CordovaSim.

  - The Hybrid Mobile Application Project wizard is extended to include selecting the Cordova version for the project and offers options to download and select locally stored Cordova versions.

  - Once created, the Cordova version of a project can also be specified from the project **Properties** window.

- Cordova plug-ins can be added to new projects during their creation as part of the New Hybrid Mobile Application Project wizard.

- On importing a Cordova project to the IDE, plug-ins referenced in the project `config.xml` file are automatically restored. This makes sharing Cordova projects within teams possible without adding plug-ins to a source control system.

- The plug-in registry used by the Cordova Plug-in Discovery wizard is now cached, resulting in less network calls and faster information retrieval.

- A `Show installed` option has been added to the Cordova Plug-in Discovery wizard, which lists plug-ins already installed for an application.

- The Cordova project `config.xml` file is automatically updated when plug-ins are added or removed from projects using the IDE Install and Remove Cordova Plug-in tools.

- The layout of the Cordova config.xml Editor has been updated to provide quicker access to common Cordova development actions.

- The Battery Status plug-in is now supported by CordovaSim and battery status values can be changed in the device input panel to simulate different battery levels and charging status for the simulated device.

- An informative prompt is displayed on deployment to CordovaSim if an application contains plug-ins unsupported by CordovaSim.

- A wizard for importing Cordova CLI-created projects into an IDE workspace is provided.

- XCode simulator integration has been updated to support XCode 6.0.

Java EE

Tooling for Java EE 7 is provided as Technology Preview in this release. This means that it is provided as-is, without support or guarantees. Your feedback is appreciated.

JBoss Central

- Early Access plug-ins are now available to install from JBoss Central. These plug-ins contain new functionality and fixes that are under development, and while you are encouraged to use them and give feedback you must be aware of the associated limited conditions of use.

- New and updated plug-ins in JBoss Central have the following versions:

  - AngularJS Eclipse (Early Access) 0.6.0.201409251425

  - Arquillian (Early Access) 1.1.2.Final-v20141016-1530-B77

  - Findbugs 3.0.0.20140706-2cfb468

  - Eclipse Checkstyle 5.7.0.201402131929

  - JBoss Tools Java Standard Tools AngularJS 3.6.0.Final-v20141016-0045-B76

  - JBoss Tools Maven Packaging Configurator 1.6.0.Final-v20141016-1914-B105

  - Maven integration for WRO4J (m2e-wro4j) 1.1.0.201406251020

  - Mylyn Builds Connector - Hudson/Jenkins 1.5.0.v20140702-2228.

  - PMD Plug-in 4.0.4v20140909-1748

  - Spring IDE 3.6.0.21407080615-RELEASE

  - Subclipse + SVNKit 1.10.4_1.8.4

- TestNG 6.8.6.20130914_0724

- The Hybrid Mobile Project wizard is featured in JBoss Central under **Start from scratch**, making it quicker to locate. This project requires Hybrid Mobile Tools and CordovaSim and when these are not already installed you are prompted by the IDE to install them before continuing with the wizard.

- A new AngularJS Forge project wizard is available in JBoss Central. The wizard creates a project skeleton and the project cheatsheet guides you through the steps necessary to use JBoss Forge and scaffold a user interface layer based on AngularJS.

- A new contacts-mobile-basic quickstart is available in JBoss Central in the **Mobile Applications** category, under **Start from a sample**. This quickstart is composed of two applications: a Cordova-based client application and a REST service back-end application.

- Usage tracking has been extended to include the following: JVM name and architecture, plug-in and quickstart installations from JBoss Central, new server creation, Forge runtime version, BrowserSim web engine type, names of completed IDE wizards, and usage counts for Forge Tools, BrowserSim and CordovaSim, JBoss Tools HTML/JSP Editor tabs and JBoss Tools HTML5 Preview. The collected anonymous information enables Red Hat to understand which features are being used, assisting in continually improving JBoss Developer Studio.

JBoss Server Tools

- Creating server adapters in the IDE for remote servers no longer requires a replica version of the remote application server and remote Java developer kit to be installed locally for the majority of server adapter functions to work. Locally installed replica components may still be required for features such as JMX Explorer.

- Server adapters for Red Hat JBoss Enterprise Application Platform are now based on server profiles, defining a set of functionality and pre-configured options determining how the server is to be run.

  - The basic server profiles for local and remote servers deploy resources locally or through the IDE RSE system. Scripts for starting and stopping the server can also be specified, where previously servers were assumed to be already started.

  - The advanced server profiles for local and remote servers (**Use management options**) connect to the server management service to perform actions such as shutdown, deployment, and determining the state of the server.

- Server runtime environments can be associated with a specific execution environment or JRE that is to be used when starting the server. The list of available environments and JREs are based on the server type and can be set during or after server runtime environment creation.

- A Behavior Profile wizard is provided in the JBoss Server Editor enabling you to modify a server profile. The wizard ensures that any consequences of server profile changes are consistently reflected in the server adapter.

- Classpath containers for Red Hat JBoss Enterprise Application Platform runtime servers are now customized based on JBoss module ID, rather than filesets specified with includes and excludes statements, to accommodate the new JBoss EAP server updates patching process. The classpath containers search all appropriate directories based on the current state of the

JBoss EAP installation to locate the correct versions of the `.jar` files to be added to the project classpath. Additionally, for projects that have a `manifest.mf` declaring **Dependencies**, these dependent modules are also added to the classpath.

- Hot-code replacement is available for module changes when a server is run in debug mode. In the event that the hot-code replacement fails, due to either obsolete methods on the stack or a changing method signature, you are presented with a prompt that allows you to terminate or restart the server, ignore the changes, or redeploy the modules. The hot-code replacement setting can be changed or reset in the **Server Editor**. You can also full-publish a module manually to force a module restart.

- The JMX Navigator view now has three groupings: User-defined connections, Server connections, and Local Processes. This categorization makes clear where connections have originated from and reduces visual noise caused by list reordering as connections are added or removed from the view.

- JVM Navigator information, displayed in the **Properties** view, is available when a JMX connection is established and the JMX node of a server is selected in the **Servers** view.

- JVM Monitor functionality has been added to the JMX Navigator. The JVM Monitor captures profiling data for running Java processes, providing information to assist in debugging applications and identifying server issues. The JMX Navigator can automatically detect system Java processes or processes can be manually connected, with JVM Monitor information displayed in the **Properties** view.

JSF Tools

- The New JSF Project wizard supports servlets 3.1

- The jQuery Mobile palette supports versions 1.3 and 1.4, each with common and unique widget wizards. The palette detects the jQuery Mobile JavaScript file version used in the HTML5 file and switches to the corresponding palette version. The palette version can also be switched manually.

- The versions of jQuery Mobile JavaScript and CSS references inserted into `.html` files by jQuery Mobile widgets can be specified from within the widget wizard. When appropriate and available, version options are listed on the second page of the jQuery Mobile widget wizard.

- A Form Button widget is available in the jQuery Mobile palette for versions 1.3 and 1.4.

- The Panel widget wizard is extended with the option to also generate a menu, combining Panel and ListView widgets in one wizard.

- jQuery Mobile palette drag-and-drop action has been improved as follows:

  - Widgets can be dragged into HTML5 files with (by default) or without the widget wizard opening (press and hold `Ctrl` while dragging).

  - Code snippets of widgets are inserted into text nodes only, preventing existing tags or attributes from being broken by drag-and drop palette actions. If an appropriate location cannot be identified, a warning is shown in the widget wizard header.

- jQuery Mobile and HTML5 palette widget templates can also be inserted into `.html` files through content assist.

- Content assist is available for jQuery Mobile 1.4 data-* attributes.

- AngularJS content assist, code highlighting and navigation are supported by the JBoss Tools HTML Editor for JavaScript projects. This AngularJS tooling is supported as an Early Access feature. The IDE must be in **Early Access enabled** mode and the Early Access version of the **AngularJS for Web Tools** plug-in must first be installed, both of which can be actioned from JBoss Central **Software/Updates** tab.

- A palette of Ionic widgets is now available in the IDE, with drag-and-drop capabilities and a content assist code completion feature. The palette is automatically listed in the **Palette** view when a HTML5 file is opened in an IDE editor. This feature is supported as Technology Preview and the **AngularJS for Web Tools** plug-in must first be installed from JBoss Central **Software/Updates** tab.

- Enhanced content assist for JavaScript is provided with support for ECMA5 and Cordova and it is automatically enabled for JavaScript and Cordova projects, respectively.

- The Properties view for HTML5 files is enhanced, providing an advanced look and feel for basic HTML5, jQuery Mobile and AngularJS components and attributes.

LiveReload Tools

- A reload delay parameter has been introduced that controls the length of time between project modifications being saved and browser reload requests being sent. This option is useful when time is needed by the application server to complete processing associated with modifications.

Maven Tools

- Maven Profile Management user interface has been donated to the Eclipse m2e project. This feature is still available with JBDS as m2e is packaged as a core component of JBDS but it is now also available to all Eclipse users.

- m2eclipse-egit, the Maven SCM Handler for EGit, is now included as part of the core JBoss Developer Studio distribution. This provides a preinstalled Git SCM connector for use in the **Checkout Maven Projects from SCM** import wizard.

- The Maven Central archetype catalog is registered by default, giving fast access to more than 9600 archetypes when using the New Maven Project wizard.

- JBoss Maven Integration has been extended with advanced settings choices when adding new repositories. You can choose the repository layout, enable and disable snapshots or releases, and change the update policy.

- The detection of local repositories by JBoss Maven Integration is improved with the search action now also looking for a unique **.maven-repository** configuration file during recursive scanning.

- JBoss project configurators for m2e now support an activation property in the **<properties>** section of a project **pom.xml** file. Expected values are **true** or **false** and override the workspace-wide preferences. The available properties are as follows:

  - **<m2e.cdi.activation>true</m2e.cdi.activation>** for the CDI Project configurator

  - **<m2e.seam.activation>true</m2e.seam.activation>** for the Seam Project configurator

- **<m2e.hibernate.activation>true</m2e.hibernate.activation>** for the Hibernate Project configurator

- **<m2e.portlet.activation>true</m2e.portlet.activation>** for the Portlet Project configurator

These XML templates are provided by content assist in the **pom.xml** editor.

OpenShift Tools

- The New OpenShift Application wizard has been modified to provide a better and simplified user experience and extended functionality.

- The New OpenShift Application wizard by default only shows projects under no version control or those under Git version control when opting to merging the new application with an existing workspace project.

- Custom **Code Anything** cartridges are supported in the New OpenShift Application wizard both as the basis of the application and as embedded cartridges. When selecting the **Code Anything** cartridge option, the Git URL for the custom cartridge source must be provided.

- OpenShift quickstarts are now available in the New OpenShift Application wizard. These quickstarts are sample OpenShift applications, combining code and one or more cartridges to demonstrate how to make effective OpenShift applications.

- Access to the New OpenShift Application wizard has been extend to the project context menu, providing more convenient wizard access when creating new OpenShift applications from existing workspace projects.

- On encountering an unknown SSL certificate when connecting to servers, OpenShift Tools prompts users to review and accept or decline the certificate.

- The OpenShift server adapter context menu, in the **Servers** view, is extended to display all application-related actions, including access to the **Configure Markers** window.

- The OpenShift server adapter now supports management of Java EE 7 projects deployed on OpenShift.

- The **Publish Changes** prompt for OpenShift projects has been updated with options for customizing your commit message and the files to commit and push as part of the publishing process. Based on the EGit commit window, the wizard also provides features for comparing files with a visual diff.

- New snapshot functionality allows users to create backups and restore OpenShift applications to and from local **.tar.gz** files. Two types of snapshots are available: a full snapshot contains the application and all local files including logs, a deployment snapshot contains a binary of the deployed application.

- On importing OpenShift applications to IDE workspaces, OpenShift Tools automatically makes IDE-specific changes to projects but no longer commits these changes without your approval.

- Additional users can be added to OpenShift domains from the IDE Create Domain and Edit Domain wizards. These wizards provide an **Edit domain members** link which opens the IDE default browser at the required OpenShift URL for managing your domain members and their read and write permissions.

Web Services Tools

- JAX-RS 2.0, part of Java EE 7, is supported, with the following new features:

  - Server-side JAX-RS interceptors and filters detection

  - @NameBinding meta-annotation detection

  - User-defined @NameBinding and @PreMatching annotations validation

  - Hyperlink navigation between filters and interceptors and associated resources methods or applications based on @NameBinding annotations

  - JAX-RS endpoint parameters displayed together with their type in the Project Explorer view

- A new JAX-RS Application wizard is available, which provides a choice between creating a subclass of javax.ws.rs.core.Application or defining a <servlet-mapping> element in the project web deployment descriptor.

- A new JAX-RS Resource wizard is available, which provides default settings for the class name, the **@Path** value at the type level, and the supported media types (by default, application/json and application/xml). The wizard also provides skeleton code generation for CRUD methods.

- Validation now occurs for parameter types in **Resource** methods.

- Quick fixes are provided in the Java editor for **@Target** and **@Retention** meta-annotations of custom NameBinding and HttpMethod annotations that do not have appropriate values and fail validation.

- The scope of Java type declarations and methods are validated as you type while editing them. When the Java element corresponds to a JAX-RS element, validation occurs when the file is saved in order to ensure validation does not impact IDE speed.

- Saving changes to a Java class triggers JAX-RS validation for all classes with related JAX-RS elements, enabling any associated problems in these latter classes to be automatically and immediately identified or resolved.

- Support is included for JAX-RS Resource Fields and Properties (fields or setter methods annotated with @PathParam, @QueryParam, @MatrixParam) as well as the JAX-RS 2.0 Parameter Aggregators (method parameters, fields or setter methods annotated with @BeanParam). These elements are checked for validation and the associated Endpoints URI templates are displayed in the Project Explorer view.

Visual Editor

- A new **HTML Preview** view is available that renders HTML source open in IDE editors. Key features of the view include refreshing as HTML source is changed or saved and source-preview highlighting for fast navigation. The view is compatible with any IDE editor with DOM model support and it is automatically used in the **Visual/Source** and **Preview** tabs of the **JBoss Tools HTML Editor** when XULRunner is not available.

Report a bug

4. ISSUES

4.1. Resolved Issues

To view information about issues resolved in this release of JBoss Developer Studio, see https://issues.jboss.org/issues/?jql= on the JBoss issue tracking system and enter the following advanced search query as one line:

```
(project in (JBDS) AND affectedVersion < "8.0.0.Alpha1" AND fixVersion in
("8.0.0.Alpha1","8.0.0.Alpha2","8.0.0.Beta1","8.0.0.Beta2","8.0.0.Beta3","
8.0.0.CR1","8.0.0.CR2","8.0.0.GA") AND fixVersion not in
("7.1.0.Beta1","7.1.0.CR1") OR project in (JBIDE) AND affectedVersion <
"4.2.0.Alpha1" AND fixVersion in
("4.2.0.Alpha1","4.2.0.Alpha2","4.2.0.Beta1","4.2.0.Beta2","4.2.0.Beta3","
4.2.0.CR1","4.2.0.CR2","4.2.0.Final") AND fixVersion not in
("4.1.0.CR1","4.1.1.Alpha1","4.1.1.Alpha2","4.1.1.Beta1","4.1.1.CR1","4.1.
2.CR1")) AND type in (Bug) AND resolution in (Done)
```

The following customer-reported resolved issues are highlighted:

JBDS-2921

On Linux-based operating systems using GNOME desktop, dragging a file from Nautilus File Manager into an Editor view did not open the file. Drop events of type String were not recognized and consequently the drag-and-drop action for files did not work. This issue has been resolved by ensuring the IDE recognizes drop events of type String and dragging a file from Nautilus File Manager into an Editor view now successfully opens the file.

JBIDE-14001

On the first IDE and server restart or the first workspace and server restart after marking a new file as deployable, the file was no longer deployed. At the session restart, there was no check to ensure the deployed file IDs matched those in the previous session. This issue has been resolved by ensuring that the file is properly found during the next session and now marking a new file as deployable persists from one session to the next.

Report a bug

4.2. Known Issues

To view information about known issues in this release of JBoss Developer Studio, see https://issues.jboss.org/issues/?jql= on the JBoss issue tracking system and enter the following advanced search query as one line:

```
(project in (JBDS) AND affectedVersion >= "7.0.0.GA" AND affectedVersion
<= "8.0.0.GA" AND (resolution in (Unresolved) OR resolution in (Done) AND
fixVersion > "8.0.0.GA") OR project in (JBIDE) AND affectedVersion >=
"4.1.0.Final" AND affectedVersion <= "4.2.0.Final" AND (resolution in
(Unresolved) OR resolution in (Done) AND fixVersion > "4.2.0.Final")) AND
type in (Bug)
```

The query results correspond to reported issues but the issues may not yet have been confirmed by the product team.

The following customer-reported known issues are highlighted:

JBDS-3069

Restarting a virtual machine in which JBoss Developer Studio was running at the time of the virtual machine suspension results in JBoss Developer Studio immediately terminating with errors logged in the output. This may also be followed with error prompts when subsequently restarting JBoss Developer Studio with the same workspace. To work around this issue, to avoid terminating errors ensure you close JBoss Developer Studio prior to suspending the virtual machine. When this is not possible, to workaround workspace error prompts start JBoss Developer Studio with a different workspace and then change to the original workspace by clicking **File→Switch Workspace**.

JBDS-3119

Certain xsd schemas appear to fail validation in Eclipse but deploy correctly. The workaround at the current time is to ignore these errors. The examples and archetypes available from JBoss Developer Studio still deploy.

JBIDE-17176

For Red Hat JBoss Portal 6, the Dynamic Web Project wizard fails to detect the JBoss Portal PortletBridge `.jar` files when JSF Portletbridge Runtime Provider is selected in the **JBoss JSF Portlet Capabilities** step. Consequently, the `.jar` files cannot be selected and associated with the project from within the wizard. To work around this issue, follow the instructions here:

1. In the **JBoss JSF Portlet Capabilities** step of the Dynamic Web Project wizard, select **Disable Library Configuration** and complete any remaining steps of the wizard.

2. In the **Project Explorer** view, right-click the project and click **Build Path→Configure Build Path**.

3. In the **Java Build Path** pane, select the **Libraries** tab and click **Add External JARs**.

4. Navigate to **/path/to/jboss-portal-6.x/modules/system/layers/gatein/org/jboss/portletbridge/** and select the **portletbridge-api-3.3.1.Final-redhat-1.jar** and **portletbridge-impl-3.3.1.Final-redhat-1.jar** files.

5. Click **OK** to close the project **Properties** window.

Report a bug

4.3. Operating System Variations and Issues

The following operating system variations and issues are highlighted:

Microsoft Windows Operating Systems

- When using BrowserSim, the JavaFX browser option is not available for Microsoft Windows operating systems and therefore Chrome Dev Tools for debugging are also not available. To work around this restriction, you can install 32-bit Safari which contains the necessary library requirements for the SWT Browser option. If opting for this work around and using a 64-bit Microsoft Windows operating system running 64-bit version of JBoss Developer Studio, you must additionally set BrowserSim to use a 32-bit JVM by editing the BrowserSim preferences from within the IDE.

- You can install JBoss Developer Studio stand-alone or Eclipse to a shared location for use by multiple users. Post-installation, the system administrator must set the installation directory to read-only for all standard users to ensure that users are only prompted to create workspaces in their own user space.

- Hybrid Mobile Tools iOS features necessitate XCode which is only available for OS X and therefore these features are not available for Microsoft Windows operating systems.

Linux-based Operating Systems

General

- GTK is required as the windowing system for the IDE on Linux distributions but if using GTK3 you may experience IDE rendering issues. To work around any issues, before starting the IDE, on the command line enter

```
export SWT_GTK3=0
```

to disable GTK3 and revert to GTK2.

- Hybrid Mobile Tools iOS features necessitate XCode which is only available for OS X and therefore these features are not available for Linux distributions.

Red Hat Enterprise Linux

- When Oracle JDK is installed through the command line with **yum**, you may find that only a subset of the complete JDK packages are installed. In particular, JavaFX may not be installed on your system and when trying to run BrowserSim with the JavaFx web engine you are prompted with IDE error messages. To ensure you have JavaFX installed, on the command line enter

```
yum install java-<version>-oracle-javafx
```

where **<version>** matches the version of Oracle JDK you are using. Alternatively, download and install the complete JDK, which includes JavaFX, from the Oracle website.

- On Red Hat Enterprise Linux 6 operating systems, only the GTK2 option is available for BrowserSim because the webkitgtk3 package is not distributed. Both SWT and JavaFX Browsers can still be used for BrowserSim giving full scope of the BrowserSim debug tools.

- On Red Hat Enterprise Linux 7 operating systems, the JavaFX Browser option is available only if choosing BrowserSim GTK option as GTK2 and the SWT Browser option is available only if choosing BrowserSim GTK option as GTK3.

Fedora

- When using BrowserSim, the JavaFX Browser option is not available for Fedora operating systems and therefore Chrome Dev Tools for debugging are also not available. The SWT Browser option is still available.

Ubuntu

- When using BrowserSim, the JavaFX Browser option is not available for Ubuntu operating systems and therefore Chrome Dev Tools for debugging are also not available. The SWT Browser option is still available.
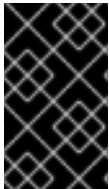
Report a bug

5. APPLY THIS RELEASE

JBoss Developer Studio 8.0.0 is available from a number of sources:

- To install JBoss Developer Studio 8.0.0, use the universal installer available from the Red Hat Customer Portal.

- To install JBoss Developer Studio BYOE 8.0.0 in Eclipse Luna, use Eclipse Marketplace, the JBoss Developer Studio update site or the update `.zip` file available from the Red Hat Customer Portal.

- To update an existing JBoss Developer Studio or JBoss Developer Studio BYOE 8.0.0.Beta or 8.0.0.CR1 installation to version 8.0.0, apply the update available from the JBoss Developer Studio update site through the IDE.

In all cases, for more information see Install Red Hat JBoss Developer Studio and Update Red Hat JBoss Developer Studio.

IMPORTANT

It is not possible to upgrade an existing JBoss Developer Studio or JBoss Developer Studio BYOE 7 installation to version 8.0.0. For more information, see Update a JBoss Developer Studio 7.x or Earlier Installation.

Report a bug

A. REVISION HISTORY

Revision 8.0.0-13   **Tue Nov 25 2014**   **Michelle Murray**

   TOOLSDOC-564: AngularJS and Ionic support statements clarified

   JBQA-11132: Java EE 7 support statement added

Revision 8.0.0-12   **Tue Oct 28 2014**   **Michelle Murray**

   JBQA-11132: Incorporated QE feedback on new features text

Revision 8.0.0-10   **Wed Oct 22 2014**   **Michelle Murray**

   Generated for 8.0.0 release