2.0.0

**Things that are changing with GWT 2.0 that might otherwise be confusing without explanation**

* Terminology changes: We're going to start using the term "development mode" rather than the old term "hosted mode." The term "hosted mode" was sometimes confusing to people, so we'll be using the more descriptive term "development mode" from now on. For similar reasons, we'll be using the term "production mode" rather than "web mode" when referring to compiled script.
* Changes to the distribution: Note that there's only one download, and it's no longer platform-specific. You download the same zip file for every development platform. This is made possible by the new plugin approach used to implement development mode (see below). The distribution file does not include the browser plugins themselves; those are downloaded separately the first time you use development mode in a browser that doesn't have the plugin installed.

**Major New Features**

* In-Browser Development Mode: Prior to 2.0, GWT hosted mode provided a special-purpose "hosted browser" to debug your GWT code. In 2.0, the web page being debugged is viewed within a regular-old browser. Development mode is supported through the use of a native-code plugin called the *GWT Developer Plugin* for many popular browsers. In other words, you can use development mode directly from Safari, Firefox, Internet Explorer, and Chrome.
* Code Splitting: Developer-guided code splitting with [GWT.runAsync()](http://www.gwtproject.org/javadoc/latest/com/google/gwt/core/client/GWT.html" \l "runAsync(com.google.gwt.core.client.RunAsyncCallback)) allows you to chunk your GWT code into multiple fragments for faster startup. Imagine having to download a whole movie before being able to watch it. Well, that's what you have to do with most Ajax apps these days -- download the whole thing before using it. With code splitting, you can arrange to load just the minimum script needed to get the application running and the user interacting, while the rest of the app is downloaded as needed.
* Declarative User Interface: GWT's [UiBinder](http://www.gwtproject.org/javadoc/latest/com/google/gwt/uibinder/client/UiBinder.html) now allows you to create user interfaces mostly declaratively. Previously, widgets had to be created and assembled programmatically, requiring lots of code. Now, you can use XML to declare your UI, making the code more readable, easier to maintain, and faster to develop. The Mail sample has been updated to show a practical example of using UiBinder.
* Bundling of resources via [ClientBundle](http://www.gwtproject.org/javadoc/latest/com/google/gwt/resources/client/ClientBundle.html). GWT introduced [ImageBundle](http://www.gwtproject.org/javadoc/latest/com/google/gwt/user/client/ui/ImageBundle.html) in 1.4 to provide automatic spriting of images. ClientBundle generalizes this technique, bringing the power of combining and optimizing resources into one download to things like text files, CSS, and XML. This means fewer network round trips, which in turn can decrease application latency -- especially on mobile applications.
* Using HtmlUnit for running test cases based on [GWTTestCase](http://www.gwtproject.org/javadoc/latest/com/google/gwt/junit/client/GWTTestCase.html): Prior to 2.0,GWTTestCase relied on SWT and native code versions of actual browsers to run unit tests. As a result, running unit tests required starting an actual browser. As of 2.0, GWTTestCase no longer uses SWT or native code. Instead, it uses *HtmlUnit* as the built-in browser. Because HtmlUnit is written entirely in the Java language, there is no longer any native code involved in typical test-driven development. Debugging GWT Tests in development mode can be done entirely in a Java debugger.

**New Features**

* GWT 2.0 introduces a number of new panels, which together form a stable basis for fast and predictable application-level layout. The official doc is still in progress, but for an overview please see [Layout Design](http://code.google.com/p/google-web-toolkit/wiki/LayoutDesign) on the wiki. The new set of panels includes [RootLayoutPanel](http://www.gwtproject.org/javadoc/latest/com/google/gwt/user/client/ui/RootLayoutPanel.html), [LayoutPanel](http://www.gwtproject.org/javadoc/latest/com/google/gwt/user/client/ui/LayoutPanel.html), [DockLayoutPanel](http://www.gwtproject.org/javadoc/latest/com/google/gwt/user/client/ui/DockLayoutPanel.html),[SplitLayoutPanel](http://www.gwtproject.org/javadoc/latest/com/google/gwt/user/client/ui/SplitLayoutPanel.html), [StackLayoutPanel](http://www.gwtproject.org/javadoc/latest/com/google/gwt/user/client/ui/StackLayoutPanel.html), and [TabLayoutPanel](http://www.gwtproject.org/javadoc/latest/com/google/gwt/user/client/ui/TabLayoutPanel.html).
* UiBinder now directly supports LayoutPanel. For example:



* [Window.Navigator](http://www.gwtproject.org/javadoc/latest/com/google/gwt/user/client/Window.Navigator.html) now provides access to the native browser's navigator object.

**Fixed Issues**

* In UiBinder <ui:style> blocks, css class names may contain dashes.
* Non-Java method safe characters in inline class names doesn't work ([#4052](http://code.google.com/p/google-web-toolkit/issues/detail?id=4052))
* @external does not work reliably for inline styles in ([#4053](http://code.google.com/p/google-web-toolkit/issues/detail?id=4053))
* Various false alarm warnings about invalid JSNI references have been fixed.
* Various Swing UI improvements.
* RPC calls leaking memory for IE ([#4133](http://code.google.com/p/google-web-toolkit/issues/detail?id=4133))
* deRPC raise an Error 500 instead of propagating the correct RuntimeException in ProdMode ([#4237](http://code.google.com/p/google-web-toolkit/issues/detail?id=4237))

## 2.1.0

**Fixed Issues**

* Creation broken if the entity id is of type String ([#1430](https://jira.springsource.org/browse/ROO-1430))
* addon-gwt is putting a boolean isChanged method in each proxy, and those don't work ([#1457](https://jira.springsource.org/browse/ROO-1457))
* ValueListBox showing redundant entries ([#1287](https://jira.springsource.org/browse/ROO-1287))
* Implement new update / create / acquire / delete events ([#1238](https://jira.springsource.org/browse/ROO-1238))
* Ensure that DefaultValueStore is always responsive ([#1217](https://jira.springsource.org/browse/ROO-1217))
* Allow both String and Long keys ([#951](https://jira.springsource.org/browse/ROO-951))
* Does the mobile.user.agent property provider actually work? ([#1468](https://jira.springsource.org/browse/ROO-1468))
* Banging on the UI can produce NPEs in DevMode window ([#1282](https://jira.springsource.org/browse/ROO-1282))
* NPE on resume in AbstractRecordListActivity([#1230](https://jira.springsource.org/browse/ROO-1230))
* RequestFactoryServlet always throws when debugging with Chrome ([#1229](https://jira.springsource.org/browse/ROO-1229))
* Implement java.math.BigDecimal support for GWT
* Update the Expenses sample to pull dependencies in via a Maven repo ([#991](https://jira.springsource.org/browse/ROO-991))
* Add the custom Expense Report as a sample ([#965](https://jira.springsource.org/browse/ROO-965))
* Find replacement for velocity ([#956](https://jira.springsource.org/browse/ROO-956))
* Hard coded refusal to send fields named "password" in servlet ([#1262](https://jira.springsource.org/browse/ROO-1262))
* No uncaught exception handler in scaffold app ([#1250](https://jira.springsource.org/browse/ROO-1250))
* Remove web.xml welcome file handling from GWT addon ([#1512](https://jira.springsource.org/browse/ROO-1512))
* Use a publicly accessible DTD for ApplicationCommon.gwt.xml ([#1271](https://jira.springsource.org/browse/ROO-1271))

## 2.2.0

The 2.2 release of GWT contains an integrated UI designer (part of the Google Plugin for Eclipse), support for HTML5 functionality, such as the Canvas/Audio/Video tags, an updated CellTable widget that now supports sortable columns and fixed column widths, and a more lenient SafeHtml template parser.

2.3.0 (M1)

This is milestone 1 of GWT 2.3.

**General Enhancements**

* Added the following functionality to the Google Plugin for Eclipse:
  + Google API integration
  + Project import from Google Project Hosting
  + Single sign on, for accessing Project Hosting and App Engine
* Added GWT SDK support for HTML5 local storage

**Noteworthy Fixed Issues**

* Updated GPE's UIBinder editor to provide support for attribute auto-completion based on getter/setters in the owner type
* Optimizations to speed up GPE launch configuration UI
* "Check for Updates", within GPE, will now detect updates to GWT and GAE SDKs
* Launching against an external URL that contains a port number now works properly in Eclipse 3.6
* Updated IE9 support [(#5125)](http://code.google.com/p/google-web-toolkit/issues/detail?id=5125)
* Fixed iFrame loading issues within Internet Explorer [(#1720)](http://code.google.com/p/google-web-toolkit/issues/detail?id=1720)

2.4.0

This is the General Availability release of GWT 2.4. See the release notes below for the full list of features and bug fixes included in this release.

The 2.4 General Availability release of GWT contains new App Engine tools for Android, incremental RPC tooling, Apps Marketplace support, a faster UI Designer with better UiBinder support, a persistent unit cache for faster iterative development, a scrolling DataGrid with fixed header, Beans.isDesignTime() support, and bundled installers that make it easier to install and configure the GPE, GWT and GAE.

**General Enhancements**

* App Engine tools for Android: Build installable Android apps that rely on App Engine for server-side support.
* Incremental RPC Tooling: Add server-side methods to App Engine code and GPE will generate the necessary serialization and Android code on the fly.
* Apps Marketplace Support: Deploy apps to the Google Apps Marketplace as easily as to App Engine.
* UI Designer: Faster startup and editing times, split-mode editing support for UiBinder, simplified CSS property editing, UiBinder morphing, IsWidget support, and more.
* Persistent Unit Cache: GWT Compiler and Development mode now cache compilation artifacts between runs. This results in faster startup time for iterative development.
* Scrolling DataGrid ([#188](http://code.google.com/p/google-web-toolkit/issues/detail?id=188)): The new DataGrid widget supports vertical scrolling with a fixed header (above) and footer (below).
* Design Time Support ([#226](http://code.google.com/p/google-web-toolkit/issues/detail?id=226)): The Beans.isDesignTime() method was added to the GWT emulation library in order to better isolate runtime-only code when a UI is edited in the UI Designer.

2.5.0 (RC1)

This is release candidate 1 of GWT 2.5. See the [What's new in GWT 2.5](http://www.gwtproject.org/doc/latest/ReleaseNotes.html) page as well as release notes below for the full list of features and bugfixes in this release.

**Major Enhancements**

* Super Dev Mode, a replacement for Development Mode (experimental)
* The Elemental library (experimental): efficient DOM access and HTML5 API's (experimental)
* New compiler optimizations from Closure
* The fragment merging optimization, for reducing latency in large apps
* Development mode performance and refresh time speedups
* A new accessibility library for setting [ARIA](http://www.w3.org/TR/wai-aria/) roles, states and properties on DOM elements
* UIBinder enhancements for rendering cells and accessing inline styles

**Changes since RC1**

* The GWT tools can now run on JDK 7. (However, no JDK 7 language or library features are available in GWT code yet.)
* The accessibility library introduced in RC1 has been cleaned up for release.
* Support for [validation](http://www.gwtproject.org/doc/latest/DevGuideValidation.html) is improved, documented, and no longer considered experimental.
* Other fixes; see the [issue tracker](http://code.google.com/p/google-web-toolkit/issues/list?q=FixedIn2_5RC2) for more.

### Security vulnerability from 2.4 to 2.5 Final

The GWT team recently learned that the Security vulnerability discovered in the 2.4 Beta and Release Candidate releases was only partially fixed in the 2.4 GA release. A more complete fix was added to the 2.5 GA release. If you have an app that's been built with GWT 2.4 or one of the 2.5 RCs, then you'll need to get the latest 2.5 release, recompile your app, and redeploy.