

TIBCO General Interface™

Release Notes

Software Release 3.2.0
November 2006

Important Information

SOME TIBCO SOFTWARE EMBEDS OR BUNDLES OTHER TIBCO SOFTWARE. USE OF SUCH EMBEDDED OR BUNDLED TIBCO SOFTWARE IS SOLELY TO ENABLE THE FUNCTIONALITY (OR PROVIDE LIMITED ADD-ON FUNCTIONALITY) OF THE LICENSED TIBCO SOFTWARE. THE EMBEDDED OR BUNDLED SOFTWARE IS NOT LICENSED TO BE USED OR ACCESSED BY ANY OTHER TIBCO SOFTWARE OR FOR ANY OTHER PURPOSE.

USE OF TIBCO SOFTWARE AND THIS DOCUMENT IS SUBJECT TO THE TERMS AND CONDITIONS OF A LICENSE AGREEMENT FOUND IN EITHER A SEPARATELY EXECUTED SOFTWARE LICENSE AGREEMENT, OR, IF THERE IS NO SUCH SEPARATE AGREEMENT, THE CLICKWRAP END USER LICENSE AGREEMENT WHICH IS DISPLAYED DURING DOWNLOAD OR INSTALLATION OF THE SOFTWARE (AND WHICH IS DUPLICATED IN TIBCO GENERAL INTERFACE RELEASE NOTES). USE OF THIS DOCUMENT IS SUBJECT TO THOSE TERMS AND CONDITIONS, AND YOUR USE HEREOF SHALL CONSTITUTE ACCEPTANCE OF AND AN AGREEMENT TO BE BOUND BY THE SAME.

This document contains confidential information that is subject to U.S. and international copyright laws and treaties. No part of this document may be reproduced in any form without the written authorization of TIBCO Software Inc.

TIB, TIBCO, Information Bus, The Power of Now, TIBCO General Interface, and TIBCO General Interface Builder are either registered trademarks or trademarks of TIBCO Software Inc. in the United States and/or other countries.

All other product and company names and marks mentioned in this document are the property of their respective owners and are mentioned for identification purposes only.

THIS SOFTWARE MAY BE AVAILABLE ON MULTIPLE OPERATING SYSTEMS. HOWEVER, NOT ALL OPERATING SYSTEM PLATFORMS FOR A SPECIFIC SOFTWARE VERSION ARE RELEASED AT THE SAME TIME. PLEASE SEE THE README.TXT FILE FOR THE AVAILABILITY OF THIS SOFTWARE VERSION ON A SPECIFIC OPERATING SYSTEM PLATFORM.

THIS DOCUMENT IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT.

THIS DOCUMENT COULD INCLUDE TECHNICAL INACCURACIES OR TYPOGRAPHICAL ERRORS. CHANGES ARE PERIODICALLY ADDED TO THE INFORMATION HEREIN; THESE CHANGES WILL BE INCORPORATED IN NEW EDITIONS OF THIS DOCUMENT. TIBCO SOFTWARE INC. MAY MAKE IMPROVEMENTS AND/OR CHANGES IN THE PRODUCT(S) AND/OR THE PROGRAM(S) DESCRIBED IN THIS DOCUMENT AT ANY TIME.

THE CONTENTS OF THIS DOCUMENT MAY BE MODIFIED AND/OR QUALIFIED, DIRECTLY OR INDIRECTLY, BY OTHER DOCUMENTATION WHICH ACCOMPANIES THIS SOFTWARE, INCLUDING BUT NOT LIMITED TO ANY RELEASE NOTES AND "READ ME" FILES.

Copyright © 2001-2006 TIBCO Software Inc. ALL RIGHTS RESERVED.

TIBCO Software Inc. Confidential Information

Contents

Release Notes	1
New Features	2
Release 3.2.0	2
Release 3.1.1	14
Release 3.1.0	14
Changes in Functionality	22
Release 3.2.0	22
Release 3.1.0	23
Deprecated Features	24
Release 3.2.0	24
Release 3.1.0	24
Migration and Compatibility	26
Release 3.2.0	26
Release 3.1.1	26
Closed Issues	27
Known Issues	34
Mac and Firefox Known Issues	34
Browser Limitations and the Margin Property	35
General Known Issues	36

Release Notes

Check the TIBCO Developer Network web site at <http://developer.tibco.com> for product information that was not available at release time.

Topics

- *New Features, page 2*
- *Changes in Functionality, page 22*
- *Deprecated Features, page 24*
- *Migration and Compatibility, page 26*
- *Closed Issues, page 27*
- *Known Issues, page 34*

New Features

This section lists features added since the last major (3.0) release of this product.

Release 3.2.0

This release introduces the following new features and updates:

- TIBCO Developer Network on page 2
- Web Browsers on page 2
- General Enhancements on page 4
- XML Mapping Utility on page 4
- Application Deployment on page 6
- Development Productivity on page 7
- General Interface Framework on page 9
- API Changes on page 11
- GUI Components on page 12
- Localization on page 13
- Samples on page 13
- Documentation on page 14

TIBCO Developer Network

The TIBCO Developer Network has moved to a new location. Visit <http://developer.tibco.com> for General Interface forums, sample applications, videos, and much more.

Web Browsers

Browsers for Deployment

TIBCO General Interface applications and TIBCO General Interface Framework can be deployed to the following browsers:

- Internet Explorer 6.0.x for Windows
- Firefox 1.5.x for Windows, Linux, and Mac OS X

Browsers for TIBCO General Interface Builder

TIBCO General Interface Builder runs in the following browsers:

- Explorer 6.0.x for Windows
- Firefox 1.5.x for Windows

Browser Limitations

Some limitations in web browsers could restrict the kinds of applications which can be deployed. For known issues with browsers, see Known Issues on page 34.

General Interface API

Methods that are implementable only in specific browsers are specified in the TIBCO General Interface API documentation. Search for Firefox and Internet Explorer in the API documentation.

Charting Add-in

The Charting add-in is supported on the following browsers and platforms in XHTML and HTML pages:

- Internet Explorer 6.0.x (VML) on Windows
- Firefox 1.5.x (SVG) on Mac OS X, Linux, and Windows

For supported platform versions, see `readme.txt` in the `GL_HOME` directory.

General Enhancements

General enhancements include the following:

- Improvements to the `jsx3.gui.Select` class include:
 - Support for `@jsxstyle` and `@jsximg` CDF attributes
 - Use `@jsxstyle` to apply a CSS style to a record. Specify any CSS code. For example, `jsxstyle="font-weight:bold;"`.
 - Use `@jsximg` to reference a graphic file to use as an icon. Specify the file path relative to the project directory, for example, `images/icon.gif`. Image position depends on the component. For best results, use a 16x16 transparent GIF file.
 - Vetoable Before Select model event
- Menu hot key accelerators follow user interface guidelines for Mac OS X and Windows.
- Support for the Command key on Mac OS X has been added. Actions that require Ctrl+click on Windows translate to Command+click on Mac OS X (Ctrl+click is equivalent to right-click on Mac).
- When General Interface Builder opens XSL and XML files for editing, it honors their encoding.

XML Mapping Utility

The XML Mapping Utility includes these enhancements:

- Re-parse part of the rules tree. To recreate deleted nodes, right-click a node in Rules Tree and choose Reparse to re-parse from that node down.
- Ability to choose a specific prototype for DOM projection.
- Log messages in the Mapper Log are color coded. For example, errors are red, traces are gray, and so on.
- JavaScript code generated by the XML Mapping Utility has changed due to a signature change in the `jsx3.net.Service` class. Rules files IDs are now passed as an argument instead of URLs. See General Interface Framework on page 9.

WSDL File Browsing

The XML Mapping Utility now resolves paths to WSDLs relative to the project. It's recommended that you copy WSDLs to a `wsdl` directory in your project. Then enter `wsdl/wsdl_file.xml` in the URL field of the XML Mapping Utility.

However, if you need to resolve a path to a WSDL outside of your project, use an absolute resolver, such as `jsxuser:///` and `jsxapp:///`.

- **jsxuser:///...** Resolves relative to the parent of the JSXAPPS directory, which is the workspace. When General Interface Builder is running, this URI resolves relative to the workspace directory.
- **jsxapp://app/...** If the Server instance corresponding to the host portion of the URI is loaded into memory, the URI is resolved relative to the application base directory (`jsxappbase`).

jsx3.net.Service

The following enhancements have been made to the `jsx3.net.Service` class. For more information, see the API documentation.

- Added a new `compile()` method that allows the output mapping rules (those that handle the server's response) to be compiled into an equivalent XSLT document, which increases performance. This allows the `Service` class to use the native XSLT processor as opposed to JavaScript in order to convert the server's response to CDF. The Sample performance difference to convert 400 records from SOAP to CDF is as follows: Firefox: 0.09 seconds vs 4.0 seconds and Internet Explorer: 0.13 seconds vs 10.1 seconds.
- Updated the `doCall()` method to accept a parameter, allowing the developer to cancel the send if the outgoing message did not pass the validation rules.
- Updated the `jsx3.net.Service.setInboundURL()` and `jsx3.net.Service.setOutboundStubURL()` methods to support paths that are relative to the project directory.

For example, if the project directory for the context server is `test`, then the following valid inputs are equivalent:

- `xml/typical.xml`
- `jsxapp://test/xml/typical.xml`
- `JSXAPPS/test/xml/typical.xml`

- Added additional debug statements for more detailed debugging capabilities when running in trace mode. Error handling has also been expanded. Added a new event type that is published each time a rule in the rules file is executed, providing even greater control over how each node in a message is created or processed.

Application Deployment

The following deployment enhancements have been added:

- Accelerated load time

- Class loading improvements

Payload size is reduced as applications now dynamically load ("lazy loading") only those classes which are needed. Classes now fall into three categories. (1) Required classes are always loaded. (2) Optional classes introduced in General Interface 3.2 classes are dynamically loaded. For example, `jsx3.gui.ColorPicker`. (3) Optional classes introduced prior to General Interface 3.2 are dynamically loaded only if the `jsx1t` deployment parameter is set to `true` in the web page that launches the application. This improves backwards compatibility. See Deployment Parameters in the Deploying Applications chapter of *TIBCO General Interface Builder Getting Started Guide*.

Use the `jsx3.require()` method to explicitly load classes synchronously. For example, debugger classes are now dynamically loaded and need to be called using the `require()` method. See Debugging on page 22.

```
jsx3.require("jsx3.ide.Debugger");
```

Specify the classpath to your classes on the Classpath page of the Project Settings dialog (Project > Project Settings).

- Changes to how add-ins are loaded improve the load time for TIBCO General Interface Builder and General Interface applications. See Add-ins on page 22.

- Simplified portal deployment

- Separation of `GL_HOME` from the workspace allows for easier application deployment. Choose your own workspace or use the default. Modify the workspace directory in the Workspace field on the Paths page of the IDE Settings dialog (Tools > IDE Settings).

- Application resources are specified using relative paths, which allows for easier portability of code from one project to another and simplifies the relocating of applications in the `JSXAPPS` folder hierarchy.

- Applications can be loaded multiple times on a single portlet page with the `jsxappns` (namespace) deployment parameter that overrides the namespace in the configuration files. See Deployment Parameters in *TIBCO General Interface Builder Getting Started Guide*.

- A loading progress indicator now displays when the General Interface Framework loads. For multiple applications loading in the same page, only one indicator shows the loading of the General Interface Framework, which is

shared by all portlet applications. Subordinate progress indicators show the loading of each portlet application.

To specify the type of progress indicator displayed, add the `jsxapploader` deployment parameter to the HTML `script` element on the web page that launches the application(s). For example, setting `jsxapploader=1`, is a good choice for portlets, because this would load a subordinate progress indicator. See Deployment Parameters in *TIBCO General Interface Builder Getting Started Guide*.

Development Productivity

TIBCO General Interface Builder includes the following productivity enhancements:

- The Common Interface Format (CIF) can be used to create GUI definition files on-the-fly or manually. The CIF format provides improved readability and smaller file size. TIBCO General Interface transforms these files to the current GUI XML structure for rendering. Currently, General Interface Builder can read but **not** save CIF files. Any edits made to a CIF file are saved in the standard serialization format. See Custom TIBCO General Interface Formats in *TIBCO General Interface Builder Getting Started Guide*.
- Reorganization of options and preferences.
 - The Deployment Options dialog has been renamed to Project Settings dialog (Project > Project Settings). Project settings include deployment, add-ins, class path settings, and legacy settings.
 - The Preferences dialog has been renamed to IDE Settings (Tools > IDE Settings). Settings include IDE settings, hot keys, and user home.
- The XML/XSL Merge Tool supports opening multiple instances of the tool.
- Writing and debugging code
 - The `logger.xml` file has been moved to the root of `GI_HOME` and now supports a configurable location. Use the environment variable, `jsx_logger_config`, in the `script` element of the launch page to specify

its location. See Logging and the Application Monitor in *TIBCO General Interface Builder Getting Started Guide*.

- Color coding of messages in the System Log palette, Application Monitor, and the Mapper Log in the XML Mapping Utility. For example, error messages are displayed in red and traces in gray.
- A configurable, optional sound can be triggered when messages are printed to the System Log palette. To enable sound for messages, change the `beepLevel` property in `GL_HOME\logger.xml` from `jsx3.util.Logger.OFF` to a message logging level, such as `jsx3.util.Logger.ERROR`. Sound is played for the specified logging level and higher. Firefox requires a plug-in that can play .wav files.
- JavaScript type-ahead lists (Ctrl+spacebar) are available in any code window and text areas with blue grid backgrounds. Type-ahead lists display the General Interface APIs and allow you to automatically insert package names, class names, methods, and fields.
- Component Hierarchy features: (1) Right-click a component and choose **Copy Name to Clipboard** to get the object name. (2) Right-click a component and choose **Copy Getter Code to Clipboard** to copy code for getting a handle to the object (`getJSXByName`).
- General Interface Builder now traps the Tab key in main stage code editors as 4 spaces. This is a General Interface Builder feature and does not affect runtime.
- When restoring a child component from the Recycle Bin, if the parent is gone, the child is restored to <ROOT>.
- Running General Interface Builder and projects
 - Ability to launch General Interface Builder in XHTML or HTML mode. Choose `GI_Builder.xhtml` or `GI_Builder.html` from `GL_HOME`. Running

- in XHTML mode is useful for developing and testing applications that run in XHTML pages, such as portlet applications.
- When General Interface Builder is run in XHTML mode, an error message is output when the Text/Html property contains non-compliant code.
- Choosing Project > Run Project, runs the project in the same mode as TIBCO General Interface Builder. For example, if General Interface Builder is running in XHTML mode, the project also runs in XHTML mode.
- Run your project from a local HTTP server. Choose **Project > Run Project from HTTP**. The deployed project relative path must match the local drive relative path to shell.html for this run option to work.
- Run Project warning displays if any open tabs have unsaved changes. A dialog displays a warning that some files are not saved and allows you to Run Without Saving, Cancel, or Save and Run.
- General Interface Builder has been rearchitected for improved load performance.
- General Interface Builder keyboard shortcuts can be configured in the IDE Settings dialog (Tools > IDE Settings > IDE Hot Keys).

General Interface Framework

TIBCO General Interface Framework includes the following changes:

- The `persistAll` argument has been added to the `toXML()` method of the `jsx3.app.Model` class, which causes the returned serialized form to include DOM nodes with transient persistence. This is useful when you want to save the exact state of the application for troubleshooting purposes.
- Rules files IDs are now passed as an argument to `jsx3.net.Service` instead of URLs. If rules files are moved to new locations, there's no need to update the URL, because the ID is used instead. The `objService` variable generated by the XML Mapping Utility has a signature change and now accepts three arguments: server, resource ID, and operation name. The new syntax is:


```
var objService = ServerName.loadResource(Project_Resource_File_Id);
objService.setOperation(Operation Name);
```
- The new Matrix component provides an `insertBefore()` method and a visual control for dragging and dropping multiple records to a new location.
- Instantiate CDFs directly without using a GUI control with the new `jsx3.xml.CDF.Document` class. See API Changes on page 11.
- User-defined XSL properties, `jsx1` through `jsx10`, have been added to `jsx3.gui.Matrix`.

- Performance improvements to the `Array.join()` method instead of `String` concatenation for the `String.fromBase64()` and `String.toBase64()`.
- The `Form` class provides an API that allows you to see the hidden `iframe` where the HTML for the form post is being built and where the response comes back from the server.

API Changes

The following changes have been made to the TIBCO General Interface API. Search for "3.2" in *TIBCO General Interface API Reference* in the online help.

- `jsx3.app`
 - `jsx3.app.AddIn` A class that represents a JSX add-in. The JSX system creates an instance of this class for every add-in that is loaded.
 - `jsx3.app.Properties` Represents a repository of dynamic properties. Dynamic properties are simply name-value pairs. Dynamic properties can be loaded from an XML file in the CDF dynamic property format.
- `jsx3.gui`
 - `ColorPicker` A GUI class that allows the user to pick a color using an HSB (hue-saturation-brightness) picker. See `jsx3.gui.ColorPicker` in the API documentation.
 - `HotKey` Encapsulates a keydown event listener that is invoked by a certain combination of keys pressed simultaneously.
 - `Image` Renders an image. See `jsx3.gui.Image` in the API documentation.
 - `Matrix` and `Matrix.Column` See Matrix Components on page 12.
 - `TimePicker` See Time Picker Component on page 12.
 - `Window` Allows for rendering a branch of the DOM of an application in a separate browser window. This is similar to the implementation of the TIBCO General Interface Builder API documentation window.
- `jsx3.net.Service` See `jsx3.net.Service` on page 5.
- `jsx3.net.URIResolver` interface An interface specifying the methods necessary to define a context against which URIs are resolved.
- `jsx3.util`
 - `DateFormat` Formats and parses dates according to a token-based string format. This class is fully localized.
 - `Locale` Represents a region of the world. Other classes may be localized, meaning that their behavior depends on their assigned locale.
 - `MessageFormat` Constructs localized messages from a pattern and parameters. A message format takes a list of arguments, formats them, and inserts them into the pattern at specific places.
 - `NumberFormat` Formats and parses numbers.
- `jsx3.xml.CDF.Document` A subclass of `jsx3.xml.Document` that implements the CDF interface. This class exposes the CDF convenience

methods on an XML document and allows you to work directly with a CDF document, without needing to work with a GUI list control. Use this class for adding records and so on.

- `jsx3.xml.Cacheable` XML Transformers replace XSL ID, XSL URL, and XSL String:
 - `getXMLTransformers()`
 - `setXMLTransformers()`

GUI Components

The classes, `jsx3.gui.Grid`, `jsx3.gui.Column`, and `jsx3.gui.List`, have been deprecated and are not supported in Firefox. The new Matrix classes, `jsx3.gui.Matrix` and `jsx3.gui.Matrix.Column`, have replaced the deprecated classes. New Matrix prototypes are available in the Component Libraries palette. See [Deprecated APIs](#) on page 24.

Tree Component

The Tree prototype has been moved to the Miscellaneous folder of the Component Libraries palette.

Matrix Components

New Matrix components, which are composites of the List, Grid, and Tree components, provide richer functionality. Matrix is fully supported in Firefox and Internet Explorer. It's recommended that you use Matrix instead of Grid, Column, and List as they have been deprecated.

These components support such features as dynamic loading of lists for better performance, tree rendering of 100 levels deep, hierarchical lists that behave like trees, multi-column trees that behave like lists, and many other features. For more information, see `jsx3.gui.Matrix` and `jsx3.gui.Matrix.Column` in *TIBCO General Interface API Reference* and *Using Matrix Components in TIBCO General Interface Builder Getting Started Guide*.

Time Picker Component

The new TimePicker component, `jsx3.gui.TimePicker`, is a clock for selecting the hour, minute, and AM or PM. Seconds, milliseconds, and a 24 hour clock can also be displayed. This control is localized.

New GUI Controls and Behaviors

- Color Picker allows the user to pick a color using an HSB (hue-saturation-brightness) picker.
- Date and Time Picker support keyboard navigation into and out of controls, as well as within the controls.
- Date Picker permits typing of date value into the DatePicker box as an alternative entry method.
- Paint methods are now XHTML compliant.
- Drop-downs for combos and selects can be wider than the unexpanded control.
- Insert stack and tab children in a specific location. See these methods of the `jsx3.app.Model` class in the API documentation: `insertBefore()`, `getFirstChild()`, `getLastChild()`, `getNextSibling()`, and `getPreviousSibling()`.
- Combo now has a `maxLength` property and method.

Localization

This release includes several new localization features:

- Date, time and number formats are now externalized in the `MessageFormat`, `NumberFormat` and `DateFormat` classes. For more information, see the new `jsx3.util` API classes for localization in General Interface Framework on page 9.
- A new deployment option, `Default Locale`, for setting language and country codes for localizing applications.

Samples

Product samples have been updated to the current version. These include `Chart`, `WSDL_Mapping_1`, and `WSDL_Mapping_2`.



Choosing an existing workspace doesn't replace previous sample applications with updated 3.2 sample applications. This built-in functionality is designed to prevent workspace files from getting overwritten. Create a new workspace to get the updated sample applications.

More samples are available at the TIBCO Developer Network at <http://developer.tibco.com>.

Documentation

This release includes the following changes to documentation:

- *TIBCO General Interface Builder GUI Reference* has been updated and included in this release.
- *TIBCO General Interface Builder Chart User's Guide*:
 - Chapter 1, *Working with Charts*, has been moved to *TIBCO General Interface Builder Getting Started Guide*.
 - Chapter 2, *Chart Components*, has been moved to *TIBCO General Interface Builder GUI Reference*.
 - *TIBCO General Interface Builder Chart User's Guide* has been removed.
- Class property descriptions have been added to the Properties Editor palette. Hover over a property, such as *Padding*, to see a description of it in the spyglass.

Release 3.1.1

This release introduces the following enhancements:

- The XML Mapping Utility's *Generate Code* button is now a menu button, which introspects for the available operations. If the rule file has not been saved, selecting an operation triggers an alert which says to save the file. Only when the rule file has been saved does the code generator put the code on the clipboard and alert the user to paste the code to the desired location.
- The signature for the `Model.loadXML()` method now accepts `jsx3.xml.Document` as well as `string`. Without this method the developer cannot pre-cache GUI components in the cache to avoid run-time reloading.

Release 3.1.0

This release introduces a wide variety of new features in the following categories:

- XML Mapping Utility on page 15
- Package Reorganization on page 15
- Usability Enhancements on page 17
- Logging on page 18
- Event Model on page 18
- GUI Components on page 19
- Help on page 21

XML Mapping Utility

- Expanded support, including:
 - WSDL (Document-Literal and RPC-Encoded)
 - Schema (2001)
 - XML/XHTML (including RSS, REST, and so on)
- Step-through tester with detailed logging, describing each node as it is processed by the XML Mapping Utility.
- Automatic UI projection. Fields described in a WSDL (or any other source format) can be leveraged by the XML Mapping Utility to create the auto-generate the GUI.
- Improved logging support with detailed information about why a mapping may have failed, including unsupported schema nodes, malformed XML, improper imports, and so on.
- Support for offline testing and configuration, allowing the mappings to be built and tested without accessing the live services.
- Automatic migration of 3.0 maps to the new 3.1 format.
- Test message generator, capable of simulating the live service.

See the *TIBCO General Interface Builder Getting Started Guide* for information on many of these features.

Package Reorganization

jsx3 The `jsx3` package contains the deprecated classes from General Interface™ 3.0. Most of these classes have been moved into one of the following packages: `app`, `gui`, `net`, `util`, and `xml`. These classes can still be referred to by their old names; however this is deprecated and not guaranteed to work in future versions.

jsx3.app The `jsx3.app` package contains classes related to General Interface applications. Classes in this package are used extensively in application JavaScript code.

The `Server` class is the controller in the JSX architecture. An instance of this class is created by the JSX runtime for each loaded application. `Server` provides access to the application DOM (`getRootBlock()`, `getDOM()`, `getJSX()`, and so on), the application cache (`getCache()`), and application settings (`getSettings()`).

The `Monitor` class, new in 3.1, is a vital tool in debugging running applications. It brings the power of the TIBCO General Interface Builder system log to running applications by displaying logging messages in a separate browser window.

jsx3.chart The `jsx3.chart` package contains all the classes contained in the General Interface Charting add-in. This package provides a collection of GUI controls that render charts. Bar, column, area, line, plot, bubble, pie, and doughnut charts are supported.

The General Interface Charting add-in provides the following features:

- Prototype charts effortlessly in TIBCO General Interface Builder, just like any other GUI control
- Render from any standard CDF data source as a chart
- A rich event model for implementing interactive charts
- Offload chart rendering to the client browser, no plug-in required

jsx3.gui The `jsx3.gui` package contains all the GUI controls available by default in the JSX runtime. Classes in this package are typically not instantiated by the developer. Rather they are prototyped in TIBCO General Interface Builder, serialized to component files, and deserialized at application runtime by the JSX runtime. TIBCO General Interface Builder provides at least one prototype for each class in this package.

General Interface 3.1 contains the following new GUI controls:

- `ImageButton` A simple button class comprised of a separate image file for each state of the button.
- `Slider` A form control comprised of a track and a handler that is draggable along the track.
- `Sound` Provides a standard interface for playing sounds of different formats.

jsx3.lang The `jsx3.lang` package contains the classes that comprise the General Interface class, inheritance, and introspection model. This model, introduced in General Interface 3.1, extends JavaScript 1.5 with the following features:

- A simple syntax for declaring classes, with support for “extends” and “implements”
- Mixin interfaces
- An extended base class, `jsx3.lang.Object`, from which all other classes descend
- Fully introspectable classes, interfaces, packages, and methods
- A base exception class, `jsx3.lang.Exception`, with stack tracing facilities
- Access to the overridden method determined by context, with the `jsxsuper()` and `jsxsupermix()` methods

Developers who use this package, rather than the typical facilities provided by JavaScript 1.5 to define custom classes will tap the significant developer-friendly features provided by this package.

jsx3.net The `jsx3.net` package contains various classes related to HTTP communication. Applications that use the data mapper and/or communicate with web services or legacy HTTP forms will use the classes in this package extensively.

`Form` allows a General Interface application to submit to legacy HTTP forms by either GET or POST and allows for file upload to such forms. `Request` is a platform independent wrapper of the browser XMLHTTP control. It allows any XML data source to be fetched either synchronously or asynchronously. `Service` is the programmatic interface onto the data mapping utility. `URI` is a simple utility class that is useful for parsing and resolving URIs.

jsx3.util The `jsx3.util` package contains various utility classes.

`EventDispatcher` is a mixin interface that when implemented by a class provides functionality according to the publish-subscribe design pattern. The `Logger` class, new in General Interface 3.1, is the primary interface of the General Interface logging system.

jsx3.xml The `jsx3.xml` package contains various classes related to XML processing.

The `Document`, `Entity`, and `Processor` classes provide platform independent interfaces onto the native browser XML DOM and XSLT libraries. `CDF` is a mixin interface, implemented by all CDF GUI controls, that provides an object-oriented interface onto the CDF XML format. `Cacheable`, also implemented by many GUI controls, is a mixin interface that provides functionality related to fetching and storing XML and XSL documents in the application cache.

Usability Enhancements

- TIBCO General Interface Builder can now be launched using `GI_Builder.hta` file in `GI_HOME`. When launched using this file, the browser menus and browser toolbars are hidden, leaving maximum screen area for development tasks.
- Tree components now support selecting multiple nodes. A new Multi-Select property for enabling or disabling this feature was added. When the property is set to Multi-Select, users can press the Ctrl+click to select multiple nodes in the hierarchy.

This change also allows you to select multiple items in the Local Data Cache palette and Project Files palette in TIBCO General Interface Builder, as well as the Rules Tree in the XML Mapping Utility.

- **Palette features** The XSL Parameters and Attributes Editor palettes are now hidden by default. You can display these palettes by manually selecting them in the Palettes menu.

The state and position of floating palettes is now stored as a project setting in TIBCO General Interface Builder, and preserved across project sessions. If a palette is floating when a project is closed, when the project is reopened the palette is still floating in the same position.

Projects

- TIBCO General Interface Builder projects can now be located in subfolders. In the Create New Project dialog, specify the project name as *project_path/project_name*, where *project_path* is one or more directory names under the *GI_HOME/JSXAPPS* directory. Separate directory names with forward slash characters (/).
- A new option for specifying whether TIBCO General Interface Builder automatically opens the last project on startup was added to Tools menu.
- Project files that begin with the string `jsx` are now supported in TIBCO General Interface Builder.
- A new Watermark field for moving the Powered By TIBCO text string to the left, center, or right side of your application window was added to the Project Deployment Options dialog.

Logging

This release contains a new logging system defined in the `jsx3.util.Logger` class and a new `jsx3.app.Monitor` class for displaying logging messages in a separate browser window. For details, see the online API help in TIBCO General Interface Builder.

For more information and code samples, check the TIBCO General Interface Developer Community at <http://developer.tibco.com>. See the document *Advanced Logging*.

Event Model

This release introduces significant changes to the way that application components handle events. The global event object (`jsx3.EVENT`) is deprecated. For API calls and user interactions that interact with events, the new model requires explicitly passing the event object, an instance of `jsx3.gui.Event`. For more information and code samples, check the TIBCO General Interface Developer Community at <http://developer.tibco.com>. See the document *Model Events*.

These changes are implemented in a backward-compatible manner, so that immediate migration is not necessary. Existing projects and applications run seamlessly without modification. However, new projects use the new event protocol unless the Event Protocol field is set to 3.0 in the Deployment Options dialog.

GUI Components

- TextBox and Password prototype components have a new Max Length property for specifying the maximum number of characters a user can type in a text box or password field.
- Select and Combo components support for the XSL attribute `jsxshallowfrom` for specifying a portion of a CDF structure to render in the component. Previously, any hierarchy information in CDF data was not shown by these components. You can add this parameter using the XSL Parameters palette, which can be accessed using the Palettes menu. To display only CDF records that are children of the root element, specify `jsxroot`.
- Tree components support a new `jsxdeepfrom` XSL parameter for specifying the `jsxid` value of the first visible node in the tree. All descendants of this node are also displayed, while the root node remains hidden.
- Support for the `insertHTML` method was added to the `jsx3.gui.Stack` and `jsx3.gui.StackGroup` classes for more efficient painting.
- Select and Combo components now support the `jsxstip` system attribute in the CDF document. This attribute provides tooltip text when the user hovers the mouse cursor over the component.
- The Block and BlockX components now support a double-click event, which allows images and labels to be used as buttons. The `setTip()` method was also enhanced to automatically repaint tooltip text. For information on descendant components, see the description of `jsx3.gui.Block` in the online API help.
- A new class, `jsx3.gui.ImageButton`, was added with support for image rollover, and for relevant events. A different image can be used for each button state. For details, see the class description in the online API help.
- Other new API classes in this release include `jsx3.util.Sound` and `jsx3.gui.Event`.
- **List and Grid components** The List and Grid prototype components have been enhanced as follows:
 - **New events** The Before Sort and After Sort events fire just before or after a user sorts the contents of a List or Grid component. Before Sort also allows the index value of the clicked column to be modified before the data is

sorted. After Sort can be used to perform cleanup tasks after data is repainted.

Similarly, the Before Resize and After Resize events fire when a user resizes the component. Resizing a component with many data rows can be canceled, or on-screen dimensions can be modified to optimize the display.

- **Attributes for XSL parameters** Support for 4 new XSL attributes was added. You can add these parameters using the XSL Parameters palette, which can be accessed using the Palettes menu.

`jsxshallowfrom` allows you to specify a portion of a CDF structure for rendering in non-hierarchical GUI components, such as List and Grid. Previously, any hierarchy information in CDF data was not shown by these components. Specify the name of the immediate parent element for this attribute. To display only CDF records that are children of the root element, specify `jsxroot`.

`jsxbg1` allows you to specify the background color of odd-numbered rows, beginning with the first row. `jsxbg2` allows you to specify the background color of even-numbered rows, beginning with the second row.

`jsxstyle` allows you to specify any valid CSS style for a record. Only the record that includes this attribute uses the specified style.

- **Additional checkbox state** CheckBox components can be selected, unselected, or partially selected. The new partially selected state can be programmatically set, for example to select a parent object in a nested list when a child object is explicitly selected. To indicate the partial selection state, the checkbox displays a minus sign (-) character.
- **Select and Combo components** The tab and arrow keys can now be used to navigate in the list of values and select a value. The type-ahead feature can be used for narrowing the list of values in a Combo component, provided the `jsxid` and `jsxtext` attributes have identical values in the CDF file used by the component.
- **Slider component** A new class, `jsx3.gui.Slider` and a new prototype component, `Slider (H)` were added. Use a slider to allow users to set a numerical value in a defined range. Clicking in the slider bar moves the handle to the clicked location. Dragging the handle changes the slider value.
- **API methods for masking** The `showMask` and `hideMask` methods were added to the `jsx3.gui.Block` class. These methods can be used to hide or disable any area of the screen. An optional message can be displayed.

Help

- A link to information on writing regular expressions was added to the Help menu in TIBCO General Interface Builder. To access this web page, select **Help > Internet Resources > Regular Expressions Guide**.
- **API reference**
 - Better visibility of JavaScript native functions, navigable type links, and automatic search cancellation when the window is closed during a search operation.
 - A menu in the API help window was added with a existing option, Show Inherited Members, and a new option, Show Deprecated Classes and Members.
 - An HTML version, displaying in a separate browser window, is now available.

Changes in Functionality

This section discusses changes in functionality since the last major (3.1.0) release of this product.

Release 3.2.0

Every effort has been made to provide backward compatibility from General Interface 3.2 to 3.1. However, some migration steps are required. For migration instructions, see *Migrating Applications to 3.2* in the Tips and Techniques section of the TIBCO Developer Network web site at <http://developer.tibco.com>.

TIBCO General Interface Framework

- For more flexibility in project deployment, the following changes have been made:
 - Projects are now saved to a workspace directory. The *workspace* is the directory that contains your projects, custom add-ins, custom prototypes, and your user settings for General Interface Builder. You can create a new workspace, choose an existing directory, or use the default directory. To change the workspace, choose **Tools > IDE Settings > Paths** and specify the path in the Workspace field.
 - The launch page and JSXAPPS and JSX folders can each be in any location on the server.

See Application Deployment on page 6.

- Debugging
 - Debugger classes are now dynamically loaded. You must use the `jsx3.require()` method before any `jsx3.ide.debug()` statements to load debugger classes. For example,

```
jsx3.require("jsx3.ide.Debugger");
```

- Add-ins

To accelerate TIBCO General Interface Builder and GI application load time, add-ins are only loaded when required.

- By default, add-ins, such as Charting, are no longer loaded automatically. Add-ins must be enabled in the project to have access to those features. To

enable add-ins, choose **Project > Project Settings > Add-ins**. Select the add-in, Save, and restart TIBCO General Interface Builder.

- The Mapping add-in is always available and doesn't need to be enabled in the Project Settings dialog.
- An `addin` directory has been added to the workspace for storing any custom add-ins you create. These add-ins aren't loaded unless enabled in the project.
- How some properties are specified has changed. Left, Top, Width, and Height properties can be specified in implied pixels or in percentages. Padding, Margin, and Border properties have also been updated. Use the abbreviated TIBCO General Interface syntax for better performance or use the CSS2 syntax. See *Entering CSS Values in TIBCO General Interface Builder Getting Started Guide*.

TIBCO General Interface Builder

The Deployment Options dialog and the Preferences dialog have been renamed and some new options added. See *Reorganization of options and preferences*. on page 7.

Release 3.1.0

Every effort has been made to provide backward compatibility from General Interface 3.1 to 3.0. In most cases applications will work with no changes. For more information on migrating 3.0 projects to 3.1, see *Migrating 3.0 Applications to 3.1* on the TIBCO Developer Network web site at <http://developer.tibco.com>.

For information on migrating General Interface 3.1 to 3.1.1, see *Migrating from Release 3.1.0 to 3.1.1* on page 26.

Deprecated Features

This section describes deprecated features and lists equivalent features that accomplish the same result, if relevant. Any use of a deprecated feature should be discontinued as it may be removed in a future release. You should avoid becoming dependent on deprecated features and become familiar with the equivalent feature.

Release 3.2.0

The following features are deprecated in this release.

Deprecated APIs

The following APIs are deprecated:

- The following `jsx3.xml.Cacheable` properties are deprecated:
 - XSL URL getters and setters
 - XSL ID getters and setters
 - XSL String getters and setters

Instead of these properties, use the XML transformers functionality:
`getXMLTransformers()` and `setXMLTransformers()`.

- `jsx3.gui.insertHTML()` is replaced by `paintChild()`

Deprecated Prototypes

`Grid`, `List`, `Column`, and `MultiSelect` have been deprecated and moved to the **Deprecated** folder in the Component Libraries palette. `Matrix` and `Matrix Column` components replace `Grid`, `List`, and `Column`. See [Matrix Components](#) on page 12.

Release 3.1.0

The following features are deprecated in this release.

- **Class Structure** The class hierarchy has been completely reorganized in this release. This has resulted in the deprecation of many classes. Refer to the API documentation for the list of deprecated classes and their replacements.

- `jsx3.out` is replaced by `jsx3.log()` and class `jsx3.util.Logger`. For information on these features, see *General Interface Logging* at the TIBCO Developer Network web site at <http://developer.tibco.com>.
- **Event Model** Static access to the current event object (`jsx3.EVENT`) was deprecated. In addition, in order to support both the 3.0 and 3.1 model event protocols simultaneously, many methods have been either deprecated or made to branch based on which event protocol is in use. Methods have been deprecated or renamed when they existed solely to generate a model event or when their names implied that they generated a model event. Preexisting methods that have different behavior based on the model event protocol are considered branching methods. For more information, see *Model Events* on the TIBCO Developer Network web site at <http://developer.tibco.com>.
- **Various Methods and Fields** Consult the API documentation in General Interface Builder Help for other deprecated methods and fields. Deprecated members are indicated with a strike-through font.

Migration and Compatibility

This section explains how to migrate from a previous release to this release.

Release 3.2.0

Migrating from Previous Releases to 3.2.0

To migrate projects from previous General Interface releases, see *Migrating Applications to 3.2* in the Tips and Techniques section of the TIBCO Developer Network at <http://developer.tibco.com>.

Release 3.1.1

Migrating from Release 3.1.0 to 3.1.1

1. Make a backup of your General Interface folder. By default, this folder is installed at `c:/tibco/gi/`.
2. Run the installer
(`TIB_gi-enterprise-builder-simple_3.1.1_win_x86.exe` or `TIB_gi-professional-builder-simple_3.1.1_win_x86.exe`).
3. Copy your projects that you would like to migrate to 3.1.1 from your backup `JSXAPPS/` folder into your new 3.1 `JSXAPPS/` folder. By default, this folder is installed at `c:/tibco/gi/3.1/JSXAPPS/`.
4. Also copy the following folder and file from the backup folder to your new 3.1 folders:
 - `user/prototypes/`
 - `JSX/logger.xml`

Migrating from Release 3.0. to 3.1.1

Users of General Interface 3.0 should read Changes in Functionality on page 22 for information on possible changes to existing 3.0 applications for deployment in 3.1 JSX/ environments.

Before installing 3.1.1, be sure to back up your General Interface folder. After installation, copy your project files, prototypes, and `logger.xml` from the backup folder to the new 3.1 folder as described in Migrating from Release 3.1.0 to 3.1.1 on page 26.

Closed Issues

The table in this section list issues that were closed in the named releases.

Closed in Release	Defect #	Summary (Sheet 1 of 7)
3.2.0	1-77QPS3	When accessing data across multiple domains on Firefox 1.5.x in Windows and Mac, an exception was thrown. For example, this occurred when running the WSDL_Mapping_2 sample application.
3.2.0	1-7A5UB3	The cursor was missing in the General Interface Builder editor when running in Firefox.
3.2.0	1-6RBJ3S	Keyboard navigation couldn't be used with the Date Picker component, which obstructed the process of using the Tab key to select components.
3.2.0	1-6OUN7V	<code>jsx3.gui.List/Grid</code> - Unselectable nodes in List/Grid could trap focus, blocking keyboard navigation past those items.
3.2.0	1-7RUQNG	While dragging a record from a can drag Matrix control to another drop-enabled Matrix control, the mouse cursor is changed into a forbidden cursor (/).
3.2.0	1-7D33L9	Calling the <code>setDisplay()</code> method on a Matrix Column child should result in the XSLT for the owning Matrix to be reset.
3.2.0	1-7CHI0R	In General Interface Builder, when adding a new class path on the Classpath page of the Project Settings dialog (Project Settings > Classpath), an error was thrown and the new record didn't immediately display.
3.2.0	1-7CIEMN	In General Interface Framework, there was a problem if you used an edit mask within a Matrix and that given Matrix instance didn't use any pagination (<code>jsx3.gui.Matrix.PAGING_OFF</code>). Calling the <code>redrawRecord()</code> method or any method that calls <code>redrawRecord()</code> , such as <code>insertRecord()</code> or <code>insertRecordNode()</code> , resulted in an error.
3.2.0	1-7CM78U	When using a paginated instance of the Matrix (<code>jsx3.gui.Matrix.PAGING_PAGED</code>), if records were added and removed and the Matrix was then repainted (by calling <code>repaintData()</code> or <code>repaint()</code>), the scrollbars didn't update to reflect the changed record count.

Closed in Release	Defect #	Summary (Sheet 2 of 7)
3.2.0	1-7AOYPQ	Restoring a Matrix Column from the Recycle Bin caused an exception.
3.2.0	1-78Q4ZV	Selecting a Matrix cell on the right-hand side of the grid caused the grid to jump to the left.
3.2.0	1-7AMECV	Selecting a different node in the Component Hierarchy palette while editing a property in the Properties Editor palette might have caused the change to affect the newly selected object, not the intended object.
3.2.0	1-79KHWA	Using the Tab key in the Properties Editor palette after editing a value triggered an exception. The Tab key didn't work in the Matrix component.
3.2.0	1-79I48Z	The Dynamic Properties editor (Matrix control) threw an exception when updating the jsxid column and required refresh to be in synch with underlying data.
3.2.0	1-78QAO1	Unable to delete text area/textbox once it was dropped in the work area toolbar (bottom of work area).
3.2.0	1-785LPA	Stack gradient color disappeared after mouse over, then mouse out.
3.2.0	1-772SE3	Select and Datepicker Heavyweight window were not anchored to the correct position when placed inside a dialog and deployed inside a portal page.
3.2.0	1-785LOQ	Stack () layout had text displayed horizontally instead of vertically.
3.2.0	1-77QQDH	Splitter handle custom image could fail in distributed deployments.
3.2.0	1-76FWL5	When switching projects in General Interface Builder after an exception, General Interface Builder appeared to be unresponsive. High CPU usage was observed (99%).
3.2.0	1-79SFGL	The XML Mapping Utility didn't give an alert when trying to parse a mapping file that didn't exist. The error message reported that the type was unrecognized when it should have reported the file wasn't found.
3.2.0	1-768GLZ	XML Mapping Utility — Static service calls using <code>setInboundURL()</code> and <code>setOutboundURL()</code> don't work. For Inbound mappings to run, the HTTP status code must be 200, 202, or 0. This affected the <code>Service</code> class when running in test/static mode.

Closed in Release	Defect #	Summary (Sheet 3 of 7)
3.2.0	1-72BKSN	The XML Mapping Utility didn't account for different namespaces when the element name was the same.
3.2.0	1-71UVED	The XML Mapping Utility ignored locally declared namespaces (those namespaces declared on a node), so it failed to see the data type for the part.
3.2.0	1-76D12H	Added support to the XML Mapping Utility to handle WSDLs created by BW 5.3.2.
3.2.0	1-75ZTPK	Menus bound to lists didn't always work properly, because the Object ID argument was missing from <code>jsx3.gui.Menu setMenu()</code> .
3.2.0	1-74NКУ7 and 1-75ZTO1	In CDF data, images in <code>tree/menu/list/grid</code> that refer to <code>JSX/images</code> or <code>JSXAPPS/application_dir/images</code> didn't resolve correctly when the <code>JSX</code> directory and <code>JSXAPPS/application_dir</code> directory didn't reside in the same directory as the launching HTML page.
3.2.0	1-748XL9	Dialog failed to paint content after being maximized if it was window-shaded.
3.2.0	1-73QTJ7	Removing a child of a layout grid (and its subclasses such as <code>StackGroup</code>) didn't properly remove the child HTML. This was true if the layout grid had been repainted but not true if the children had been added with <code>insertHTML()</code> .
3.2.0	1-72UPUR	The method, <code>jsx3.net.Service getEndpointURL()</code> , appeared in API help but not when using the type-ahead feature (<code>Ctrl+spacebar</code>).
3.2.0	1-72KUMP	Hot keys registered as properties of the button classes or registered as the <code>jsxkeycode</code> CDF attribute of menu should register themselves with the first ancestor DOM node that was a <code>jsx3.gui.Window</code> , <code>jsx3.gui.Dialog</code> , or root block of a server.
3.2.0	1-72G8V7	CDF records with only numeric IDs cause ID collisions.
3.2.0	1-70ZZYW	Missing a method to unselect a selected item in Grid. There wasn't a functional equivalent to List behavior.
3.2.0	1-70Q86W	Added missing <code>getServer()</code> method to the <code>jsx3.net.Service</code> class, allowing for more consistent experience in both General Interface Builder and runtime.

Closed in Release	Defect #	Summary (Sheet 4 of 7)
3.2.0	1-712JG1	It wasn't possible to combine nodes or documents that were created using differing threading models. The threading model used by the xmlhttp control and the threading model used by static XML documents were different.
3.2.0	1-6A9E3T	<code>jsx3.gui.Tree</code> — Key navigation in a tree followed the order of the records in the underlying CDF document. If the View was resorted using the XSL parameter <code>jsxsortpath</code> , arrow key navigation in the tree wouldn't follow the order of onscreen tree nodes, resulting in unexpected behavior.
3.2.0	1-6FEKS8	General Interface Builder — Certain sequences of author-time events, including toggling an object's properties between Absolute and Relative, could cause strange results later when trying to move the object by dragging the object's center point. The object appeared in a different location than where it was dropped.
3.2.0	1-6OJO49	<code>jsx3.gui.Tree</code> - Trees with hidden root nodes took a long time to paint when they were enclosed several levels deep within other components, and the ancestor components were repainted. Setting the Show Root Node property to Hide caused 15 second repaint. Setting to Show, resulted in instant repaint.
3.2.0	1-6Q2SLD	Use an asynchronous call for the XML Mapping Utility to keep the browser from locking up if the WSDL was unavailable. Provide a Cancel button so the user could cancel if the call was taking too long.
3.2.0	1-6U3XMU	[Author Time] <code>jsx3.GO()</code> and other methods in getting a handle to an object which do not include the namespace found objects which have been placed in the trash can. This made updates to the view at author time unpredictable.
3.2.0	1-6XUR3U	Pressing Tab or Shift+Tab while in an open select or combo box should select the current item, close the select/combo, and move focus to the next (previous) item.
3.2.0	1-6VWJ99	[Author Time] <code>jsx3.gui.Grid</code> — Grid won't accept a drop unless the auto-expand property was set to TRUE. This resulted in an on-screen control that added extra rows to the view each time the user clicked in the last row of the Grid.

Closed in Release	Defect #	Summary (Sheet 5 of 7)
3.2.0	1-6X3V2V	Adding more than two children to a splitter parent caused unpredictable runtime behavior.
3.2.0	1-6X6AG4	General Interface Builder — When the Source XML (Expert) view of a GUI Component (2nd button on the bottom of the work area) was active, only the Project Files palette and the Local Data Cache palette should be enabled. Because other areas of the user interface were enabled, it was possible to make changes to the component using the Component Hierarchy palette or Properties Editor palette, and so on, which were not persisted.
3.2.0	1-6X6AIS	General Interface Builder — There was no warning given when a developer tries to import a Referenced or Referenced-async GUI component as a child of JSXBODY (the topmost node in the Component Hierarchy). All referenced components needed to be a child of an object in the component hierarchy of the referencing component. Otherwise, when the referencing component was saved, the reference was lost.
3.2.0	1-3HVT9B	General Interface Builder — The Properties Editor palette was missing a no border property.
3.1.1	1-6WT94Q	The Radio button prototype column didn't have a default groupname, so the radio button column didn't work as expected.
3.1.1	1-6XJTCH	Dialogs — Clicking a Dialog icon in the taskbar should bring the dialog forward if it wasn't forward. Instead, it minimized the dialog. The requested behavior follows that of Windows taskbar and Windows icons.
3.1.1	1-6XP1CC	XSL document was reloaded when destroying List/Grid.
3.1.1	1-6XY6MN	XML documents remained in the IDE cache after rules creation and update, and then the XML Mapping Utility was closed. These XML documents accumulated each time the XML Mapping Utility was used and closed, taking up resources.
3.1.1	1-6XY6Q4	3.0 Rules with POST and empty message threw errors.
3.1.1	1-6XY6RN	When selecting a SOAP Fault node in the mapper, an unnecessary error message was sent to the General Interface Builder log.

Closed in Release	Defect #	Summary (Sheet 6 of 7)
3.1.1	1-6X6LLE	Pressing the Enter key while in the Type column of the Dynamic Properties Editor (or combo box column in any Grid) moved the focus two rows down.
3.1.1	1-6X98XH	Alerts with large text blocks didn't scroll and the OK button disappeared.
3.1.1	1-6YKVLJL	DatePicker failed in GMT- <i>nn</i> time zones.
3.1.1	1-6YXIXS	Events on chart series (and possibly other chart components) fired twice. The event handlers were incorrectly painted in the group containing the category shapes. They should have only been painted in the category shapes themselves.
3.1.1	1-6Y3BDR	Mapping Rule Migration - The CXF Rule conversion utility didn't automatically set <code>Restriction = Nillable</code> for empty nodes present in 3.0 CXF Rules.
3.1.1	1-6ZL8JH	When overwriting an existing installation, the Installer permanently deleted user-created files (files which were not installed as part of the package).
3.1.1	1-6ZS55H	Changing the Show Deprecated Classes option collapsed the tree, unselected the current class, and prevented the selection of methods.
3.1.1	1-6Z6WKR	XML markup text didn't render in the logger.
3.1.1	1-6Z6C5E	<code>jsx3.app.Monitor</code> didn't escape HTML/XML markup text.
3.1.1	1-70Q8IS	Combo control for grid failed to check for 'null' on the 'select' event, causing quirky behavior.
3.1.1	1-70Q8JU	The Properties Editor palette got out of alignment when the Enter key was used for navigation.
3.1.1	1-72252X	<code>jsx3.net.Service.setTimeout</code> caused JavaScript error, "onTimeout is undefined".
3.1.0	1-63NS8X	In General Interface Builder, the SOAP Mapping Utility (now renamed XML Mapping Utility) could lose the association to the rule being edited. When this happened, the Test Rules tab stopped functioning. This behavior had no impact on runtime behaviors or performance.

Closed in Release	Defect #	Summary (Sheet 7 of 7)
3.1.0	1-692LVH	The JavaScript function <code>setTimeout</code> was not supported in the version of the HTTP object referenced in Project Deployment Options.
3.1.0	1-5ZWDAG	When an empty area chart was created, an error was generated. This error occurred when there wasn't any data to display on chart axes.

Known Issues

This section discusses known issues in this release.

Mac and Firefox Known Issues

Mac OS X and Firefox have the following known issues.

- The JavaScript Step Through Debugger isn't supported in Firefox. To debug in Firefox, use the Venkman extension at <http://getahead.ltd.uk/dwr/ajax/venkman/>.
- Inconsistent scrollbar support on Mac OS X. Also if layering is used, such as a dialog layered over an application, scrollbars bleed through. This occurs whenever z-index is used. See defect 1-7SGRHB on page 36.
- For charting on Firefox with Mac OS X, some SVG behavior is unexpected. For example, text isn't always rendered properly due to clipping. This is possibly related to scrolling. See defect 1-78I4SR on page 36.
- Mac OS X accelerator keys don't always work in Firefox.
- Firefox doesn't support `jsx3.net` Form with file upload. As an alternative, set the Text/HTML property of a Block component to free form HTML containing a `form` element.
- Unexpected layout behavior in Firefox with the Block component, such as misaligned GUI components, when using relative positioning.

Workaround To avoid unexpected layout behavior in deployed applications, it's recommended to use Block as a container **only** if it meets **one** of these requirements:

- The Block is owned by a layout manager, such as LayoutGrid, Tab, Stack, and Splitter.
 - The Block is relatively positioned and has a width of 100%.
 - The Block is absolutely positioned.
- Overflow on a draggable object, such as a dialog, is hidden.
Workaround Make the dialog visible by adding width and height.

- XPath/XSLT in Firefox
 - The XSL must meet these requirements in Firefox or it fails:
 - The XSL must point to the following namespace:
`xmlns:xsl="http://www.w3.org/1999/XSL/Transform"`
 - The XSL must output valid XML. Balanced tags are required. For example,
`<xMessage>Hello World!</xMessage>`
 - XSLT implementation does not support the namespace axis, limiting the ability to query and discover namespaces. The DOM-based interface also fails to implement this axis.
 - XSLT implementation does not support the `node-set()` method, which means that complex parameters and result tree fragments cannot be resolved.
 - XSLT implementation does not allow output escaping to be disabled, which means that escaped entities cannot be resolved during a transformation.
 - The only output formats supported for XSLT processing are HTML 4.0 and XML.

For other browser issues, see General Known Issues on page 36.

Browser Limitations and the Margin Property

The behavior of the margin property varies by browser as follows:

- Internet Explorer applies all margins properly when the object is rendered, but if an ancestor container is ever hidden (`display:none`), the left margin is no longer honored when redisplayed.
- Firefox applies all margins if the object does not specify an overflow property (`overflow:hidden`, `overflow:auto`, `overflow:scroll`). If the object uses overflow, the top and bottom margin settings aren't honored.

In summary, margin properties are honored as follows:

- The only margin setting that is honored by all browsers in all situations is `margin-right`.
- The margin properties, `margin-top`, `margin-bottom`, and `margin-right`, are honored in all browsers if the object does not specify an overflow property of `hidden`, `scroll`, or `auto`.
- The margin properties, `margin-top`, `margin-bottom`, `margin-left` and `margin-right`, are honored in all browsers if the object does not specify an

overflow property (hidden, scroll, or auto) and no ancestor of the object ever has its display property toggled between none and block.

General Known Issues

The table in this section lists known issues in this release.

Defect #	Summary/Workaround (Sheet 1 of 4)
1-78I4SR	<p>Summary [Runtime] On a Mac, text in charts on Firefox 1.5.x doesn't display properly. The text is drawn correctly but after dragging the parent, it isn't redrawn. In some cases, text is drawn when the scroll bar is scrolled. This defect has been reported to https://bugzilla.mozilla.org/show_bug.cgi?id=316564.</p>
1-7SGRHB	<p>Summary [Runtime] On Mac OS X, if multiple dialogs are open, the vertical scroll bars of the bottom dialog bleed through to the dialog on top. This defect has been reported to https://bugzilla.mozilla.org/show_bug.cgi?id=187435.</p>
1-7DS71V	<p>Summary [Runtime] The main body of the Matrix does not respond to the scroll wheel on the mouse. However, the scrollbar does respond.</p> <p>Workaround Hover the mouse over the scrollbar to use the scroll wheel.</p>
1-7BPVY0	<p>Summary [Runtime] In a multiple-level menu, moving the mouse quickly in and out of a nested submenu causes the submenu to detach and move to the top left of the browser window. At author time, an error is displayed in the system log.</p> <p>Workaround Move the mouse more slowly.</p>
1-6Y5QMD	<p>Summary [Runtime] Trees which contain more than 15-20 nested levels do not render completely in the browser. Although the HTML is painted by the General Interface Framework, the browser does not render all the levels.</p> <p>Workaround Use the Matrix components.</p>
1-6BYOQP	<p>Summary [Runtime] <code>jsx3.ide.debug()</code> statements at runtime (or Project > Run) throw a JavaScript error in the browser.</p> <p>Workaround Wrap debug statements in an <code>if</code> statement, which refers to a global <code>isDebug</code> variable. Then set the variable to <code>True</code> for development and <code>False</code> for deployment.</p>
1-7TIWOV	<p>Summary [Author Time] The API documentation incorrectly states that the Matrix <code>getSortPath()</code> and <code>setSortPath()</code> methods require the name of the attribute to be preceded by an <code>@</code> symbol. The <code>@</code> symbol is not required and will break the sort if it is used.</p>

Defect #	Summary/Workaround (Sheet 2 of 4)
1-7T87PZ	<p>Summary [Author Time] On Firefox, when you drop a charting component onto the work area and try to shift focus to that component using the Show/Hide Focus Rectangle button, the charting component doesn't get the focus. Hence you can't drag-and-drop the component as desired. This defect has been reported to https://bugzilla.mozilla.org/show_bug.cgi?id=360141.</p>
1-7T8VF1	<p>Summary [Author Time] Not all classes that extend Block implement all of the methods or input types available to Block. In these cases, the API documentation lists those methods. The Properties Editor palette accepts and displays only the correct members.</p>
1-7T33C9	<p>Summary [Author Time] On Mac, menu accelerator hot keys aren't working for some General Interface Builder menu items. For example, IDE Settings (Ctrl+), Project Settings (Ctrl+Shift+), and all palettes (Ctrl+1, Ctrl+2, and so on).</p>
1-7QJTEX	<p>Summary [Author Time] In Internet Explorer, you can't create a new folder under the C:\ root directory using the Create New Folder button while creating a workspace folder. Creating subfolders under an existing folder is not an issue.</p> <p>Workaround Create the folder under the C:\ root directory in the Windows file manager. In General Interface Builder, choose the Browse button in the Create or Locate a Workspace dialog or in the Workspace field of the IDE Settings dialog (Tools > IDE Settings > Paths) and select the newly created folder.</p>
1-7KV0SU	<p>Summary [Author Time] The work area tabs don't cascade to the next row if they exceed the viewable space.</p> <p>Workaround Scan through the files by pressing the left and right arrow keys.</p>
1-7HWJMF	<p>Summary [Author Time] When running General Interface Builder in Internet Explorer, the following steps cause an exception:</p> <ol style="list-style-type: none"> 1. The browser does not occupy the entire screen. 2. The Properties Editor palette is docked on the right side. 3. A property in the Properties Editor palette is being edited that uses a textbox mask. 4. The developer selects the text and continues the selection off the right edge of the browser. 5. The developer releases the mouse off the right edge of the browser. <p>Workaround Avoid the previously listed steps.</p>

Defect #	Summary/Workaround (Sheet 3 of 4)
1-79MFUX	<p>Summary [Author Time] Firefox determines the user home from the install directory and is case sensitive, even though the file system is case insensitive.</p> <p>Workaround If you change the browser URL from C: to c:, you need to specify the user home directory again.</p>
1-7A9DSG	<p>Summary [Author Time] Firefox can't open non-XML files encoded as UTF-16.</p> <p>Workaround Use the General Interface Builder IDE, Internet Explorer, or a text editor to convert to 8-bit encoding before opening in Firefox.</p>
1-74L8YJ	<p>Summary [Author Time] General Interface Builder error sound does not play properly in Firefox 1.5.x.</p> <p>Workaround Download and install a plug-in for .wav files.</p>
1-7AP5XP	<p>Summary [Author Time] In Firefox, moving the mouse over a context menu may cause an error message in the System Log palette. This message can be ignored.</p>
1-7SPC40	<p>Summary [Author Time] When running in XHTML mode, API help throws an exception if the inline comments in the API documentation are not XHTML-compliant. For example, API help for <code>jsx3.app.Server.getEnv()</code> triggers an exception.</p>
1-7T5QGX	<p>Summary [Author Time] Matrix controls created in the General Interface 3.2 Beta 2 release cause "no such method" or "function required" exceptions when loaded with General Interface 3.2. This is due to obfuscated variants in the serialization files.</p> <p>Workaround Check the serialization file for obfuscated variant attributes, such as <code>hk=" []"</code>. Remove these obfuscated variants manually in Source XML (Expert) view in General Interface Builder or use a text edit tool to search and remove all instances.</p>
1-70DGBK	<p>Summary [Author Time] XML Mapping Utility - While retrieving and parsing a remote WSDL document, if the developer makes a typo in the URL, General Interface Builder locks up until the server returns a "document not found" error.</p> <p>Workaround Do not make typos in the URL field or work from a local copy of the WSDL.</p>

Defect #	Summary/Workaround (Sheet 4 of 4)
1-7C0PJX	<p>Summary [Author Time] XML Mapping Utility - Nodes are missing in the XML Mapping Utility. This occurs when a WSDL references a schema, which in turn references another schema. The XML Mapping Utility does not support this feature and the behavior is as expected. The <code>xsd:extension</code> element will not properly parse if the element is based upon an <code>xsd:extension</code> element that is based upon yet another <code>xsd:extension</code> element. The workaround is to manually create the necessary mapper nodes that were missed during the parse.</p> <p>Workaround Manually create the necessary mapper nodes that were missed during the parse.</p>
1-754023	<p>Summary [Author Time] Cursor in text tabs bleeds through a dialog in front of it.</p>
1-6E1BI5	<p>Summary [Author Time] General Interface Builder - When working in editable text tabs (JavaScript, XML, XSL, etc.), the type-ahead context menu (activated with Ctrl+spacebar) becomes disabled under certain circumstances.</p> <p>Workaround Click anywhere in the edit window to reset the Ctrl+spacebar listener.</p>
1-6THXVP	<p>Summary [Author Time] Submenus and menus are not aware enough to always cascade in the most optimal direction. Long menus, particularly those with long submenus, at the bottom of the screen will be rendered with scrollbars.</p> <p>Workaround Be sure to keep menus short enough and away from the bottom of the screen.</p>
1-6VYSPR	<p>Summary [Author Time] When General Interface Builder is launched from the HTA file, if Windows XPSP2 goes to sleep, IE will hang when the user re-logs into Windows.</p> <p>Workaround Launch General Interface Builder from <code>GI_Builder.html</code>.</p>
1-6Y33HD	<p>Summary [Author Time] The Debugger will occasionally evaluate a commented line.</p>

