G2 Bundle

Release Notes
Version 8.4 Rev. 2

G2 PLATFORM

gensym

G2 Bundle Release Notes, Version 8.4 Rev. 2

 June 2009

The information in this publication is subject to change without notice and does not represent a commitment by Gensym Corporation.

Although this software has been extensively tested, Gensym cannot guarantee error-free performance in all applications. Accordingly, use of the software is at the customer's sole risk.

Contents Summary

Contents

Preface

Describes the audience and conventions that these release notes use.

Introduction

These release notes introduce the new features and changes in the G2 Bundle Version 8.3. The documentation that ships with the bundle corresponds with the G2 Bundle Version 8.3 Rev. 0. These release notes include the following parts and chapters:

- **Part I** provides:

  - An overview of the G2 Bundle, a summary of new features, and important upgrade instructions.

  - G2 Bundle Component Availability describes the availability of each component on the various platforms.

- **Part II** provides a summary of new features and changes for the G2 core environment in:

  - G2 Version 8.3 Rev. 1

  - G2 Version 8.3 Rev. 0

- **Part III** provides a summary of new features and changes for the Telewindows client in:
  - Telewindows Version 8.3 Rev. 1
  - Telewindows Version 8.3 Rev. 0
- **Part IV** describes new features and changes to G2 utilities:
  - G2 Developer's Utilities
  - G2 Utilities
- **Part V** describes new features and changes to G2 bridges:
  - G2 ActiveXLink
  - G2 CORBALink
  - G2 Database Bridges
  - G2 Gateway
  - G2-HLA Bridge
  - G2 JavaLink
  - G2 Java Mail Bridge
  - G2 Java Socket Manager
  - G2 JMSLink
  - G2-OPC Client Bridge
  - G2-PI Bridge
  - G2-SNMP Bridge
  - G2 WebLink
- **Part VI** describes new features and changes to G2 applications:
  - G2 Diagnostic Assistant (GDA)
  - G2 e-SCOR
  - G2 ReThink
  - G2 Optegrity
  - G2 Integrity
  - G2 SymCure
  - G2 NeurOn-Line

Audience

These release notes are written for G2 users with familiarity with G2 Version 8.x.

Conventions

This guide uses the following typographic conventions and conventions for defining system procedures.

Typographic

Convention Examples	Description
g2-window, g2-window-1, ws-top-level, sys-mod	User-defined and system-defined G2 class names, instance names, workspace names, and module names
history-keeping-spec, temperature	User-defined and system-defined G2 attribute names
true, 1.234, ok, "Burlington, MA"	G2 attribute values and values specified or viewed through dialogs
Main Menu > Start KB Workspace > New Object create subworkspace Start Procedure	G2 menu choices and button labels
conclude that the x of y ...	Text of G2 procedures, methods, functions, formulas, and expressions
new-argument	User-specified values in syntax descriptions
text-string	Return values of G2 procedures and methods in syntax descriptions
File Name, OK, Apply, Cancel, General, Edit Scroll Area	GUIDE and native dialog fields, button labels, tabs, and titles

Convention Examples	Description
File > Save Properties	GMS and native menu choices
workspace	Glossary terms
`c:\Program Files\Gensym\`	Windows pathnames
`/usr/gensym/g2/kbs`	UNIX pathnames
`spreadsh.kb`	File names
`g2 -kb top.kb`	Operating system commands
`public void main()` `gsi_start`	Java, C and all other external code

Note Syntax conventions are fully described in the *G2 Reference Manual*.

Procedure Signatures

A procedure signature is a complete syntactic summary of a procedure or method. A procedure signature shows values supplied by the user in *italics*, and the value (if any) returned by the procedure <u>underlined</u>. Each value is followed by its type:

g2-clone-and-transfer-objects
    (*list*: class item-list, *to-workspace*: class kb-workspace,
    delta-x: integer, *delta-y*: integer)
    –> <u>*transferred-items*</u>: g2-list

Related Documentation

G2 Core Technology

- *G2 Bundle Release Notes*

- *Getting Started with G2 Tutorials*

- *G2 Reference Manual*

- *G2 Language Reference Card*

- *G2 Developer's Guide*

- *G2 System Procedures Reference Manual*

- *G2 System Procedures Reference Card*

- *G2 Class Reference Manual*

- *Telewindows User's Guide*

- *G2 Gateway Bridge Developer's Guide*

G2 Utilities

- *G2 ProTools User's Guide*

- *G2 Foundation Resources User's Guide*

- *G2 Menu System User's Guide*

- *G2 XL Spreadsheet User's Guide*

- *G2 Dynamic Displays User's Guide*

- *G2 Developer's Interface User's Guide*

- *G2 OnLine Documentation Developer's Guide*

- *G2 OnLine Documentation User's Guide*

- *G2 GUIDE User's Guide*

- *G2 GUIDE/UIL Procedures Reference Manual*

G2 Developers' Utilities

- *Business Process Management System User's Guide*

- *Business Rules Management System User's Guide*

- *G2 Reporting Engine User's Guide*

- *G2 Web User's Guide*

- *G2 Event and Data Processing User's Guide*

- *G2 Run-Time Library User's Guide*

- *G2 Event Manager User's Guide*

- *G2 Dialog Utility User's Guide*

- *G2 Data Source Manager User's Guide*

- *G2 Data Point Manager User's Guide*

- *G2 Engineering Unit Conversion User's Guide*

- *G2 Error Handling Foundation User's Guide*

- *G2 Relation Browser User's Guide*

Bridges and External Systems

- *G2 ActiveXLink User's Guide*
- *G2 CORBALink User's Guide*
- *G2 Database Bridge User's Guide*
- *G2-ODBC Bridge Release Notes*
- *G2-Oracle Bridge Release Notes*
- *G2-Sybase Bridge Release Notes*
- *G2 JMail Bridge User's Guide*
- *G2 Java Socket Manager User's Guide*
- *G2 JMSLink User's Guide*
- *G2-OPC Client Bridge User's Guide*
- *G2 PI Bridge User's Guide*
- *G2-SNMP Bridge User's Guide*
- *G2-HLA Bridge User's Guide*
- *G2 WebLink User's Guide*

G2 JavaLink

- *G2 JavaLink User's Guide*
- *G2 DownloadInterfaces User's Guide*
- *G2 Bean Builder User's Guide*

G2 Diagnostic Assistant

- *GDA User's Guide*
- *GDA Reference Manual*
- *GDA API Reference*

Customer Support Services

You can obtain help with this or any Gensym product from Gensym Customer Support. Help is available online, by telephone, by fax, and by email.

To obtain customer support online:

➔ Access G2 HelpLink at `www.gensym-support.com`.

You will be asked to log in to an existing account or create a new account if necessary. G2 HelpLink allows you to:

* Register your question with Customer Support by creating an Issue.

* Query, link to, and review existing issues.

* Share issues with other users in your group.

* Query for Bugs, Suggestions, and Resolutions.

To obtain customer support by telephone, fax, or email:

➔ Use the following numbers and addresses:

	Americas	**Europe, Middle-East, Africa (EMEA)**
Phone	(781) 265-7301	+31-71-5682622
Fax	(781) 265-7255	+31-71-5682621
Email	`service@gensym.com`	`service-ema@gensym.com`

Part I

Introduction

G2 Bundle Overview

Describes the software and documentation that make up the G2 Bundle, provides a summary of new features in the G2 Bundle, and describes how to upgrade your KBs to the new version.

gensym

Introduction

The G2 Bundle provides a basic suite of products for developing and deploying intelligent, knowledge-based, real-time applications that can communicate with dynamic external systems.

The bundle consists of the G2 executable, Telewindows, G2 utilities, G2 Developer's utilities, G2 bridges and integration tools, and G2 applications. For the current version numbers, see the `readme-g2.html` file. For information on

platform compatibility for each of the G2 components, see Chapter 2, "G2 Bundle Component Availability" on page 35.

The G2 Executable

The G2 executable is Gensym's core product. It is an object-oriented, graphical environment for building and deploying expert applications, and is the foundation for Gensym's layered products.

The executable is `g2\g2.exe` on Windows and `g2/g2` on UNIX.

We recommend that you use the following batch file and shell script for launching G2:

Windows:	`g2\StartServer.bat`
UNIX:	`g2/StartServer`

Telewindows

Telewindows allows you to open multiple local and remote windows into a running G2 to view and work with the contents of the current knowledge base. Your G2 application can offer each Telewindows user a distinct view of the knowledge base depending on such factors as functional privilege levels and natural-language interfaces. The executables are `g2\twng.exe` (Telewindows Next Generation) and `g2\tw.exe` (Telewindows) on Windows, and `g2/tw` on UNIX.

G2 Utilities

The following G2 Utility applications are part of the G2 Bundle. All of the knowledge bases except G2 ProTools are located in `g2\kbs\utils\` on Windows and `g2/kbs/utils/` on UNIX.

- **G2 ProTools** provides advanced G2 developer tools for speeding up development, testing, debugging, documenting, and deployment. The top-level knowledge base is located in `protools\kbs\protools.kb` on Windows and `protools/kbs/protools.kb` on UNIX.

- **G2 Dynamic Displays (GDD)** gives you the ability to add dynamically generated graphical attribute readouts to icons in your applications. The top-level knowledge bases are `gdddev.kb` and `gddlib.kb`.

- **G2 Developer's Interface (GDI)** enables you to develop a consistent Microsoft Windows-like graphical user interface for G2. The top-level knowledge base is `g2cuidev.kb.`

- **G2 Foundation Resources (GFR)** provides you tools for implementing multiple-module knowledge bases. The top-level knowledge base is `gfr.kb`.

- **G2 Menu System (GMS)** enables you to create menus similar to the menus on PCs. The top-level knowledge base is `gms.kb`.

- **G2 OnLine Documentation (GOLD)** enables you to distribute and access online documentation from within your G2 application. The top-level knowledge base is `goldui.kb`.

- **G2 GUIDE (GUIDE)** enables you to create graphical user interfaces for G2 applications. The top-level knowledge base is `guide.kb`.

- **G2 GUIDE/User Interface Library (GUIDE/UIL)** provides an application programmer's interface (API) to procedures that perform basic operations on the dialogs and to UIL controls that you use with GUIDE to create a user interface. The top-level knowledge base is `uil.kb`.

- **G2 XL Spreadsheet (GXL)** allows you to create, display, and edit tabular data in G2 in a familiar spreadsheet style. The top-level knowledge base is `gxl.kb`.

G2 Developer's Utilities

The G2 Developer's Utilities consist of a set of modules that provide a consistent development framework for building G2 decision management applications. These modules are located in the `g2i` directory.

- **Business Process Management System (BPMS)** provides a user interface, classes, methods, and built-in services that are based on the G2 Graphical Language (G2GL).

- **G2 Business Rules and Management System (BRMS)** provides a mechanism for easily editing, organizing, analyzing, and executing complex business rules.

- **G2 Web Services (GWEB)** defines out-of-the-box Web pages and SOAP services, as well as classes and APIs enabling G2 to implement an HTTP server and serve HTML pages, XML structures, SOAP services, and files.

- **G2 Reporting and Processing Engine (GRPE)** provides a consistent approach for defining reports and charts, collecting values, displaying tabular values in reports, and charting those values.

- **G2 Event and Data Processing (GEDP)** is a multi-purpose graphical language composed of graphical blocks that can be connected together to express a flow of data, perform calculations, execute functions, generate messages, and events (`gedp.kb`).

- **G2 Event Manager (GEVM)** provides tools that support highly scalable, distributed operator-advisory applications by providing an event "black

board" and alarm management capabilities, as well as associated message queues, message browsers, and logging (`gevm.kb`, `gqs.kb`, and `glf.kb`).

- **G2 Run-Time Library (GRTL)** provides a wide variety of development tools for the runtime environment. These include support for object models, which includes object keys, event notification, and support for localization, configuration files, command-line options, publish/subscribe, XML, and a variety of general runtime utilities (`grtl.kb`).

- **G2 Dialog Utility (GDU)** extends the custom Windows dialog functionality that G2 provides to enable the rapid building and deployment of native Windows dialogs (`gdu.kb`). This module also includes the **G2 Dialog Conversion Utility (GDUC)**, which generates custom Windows dialog specifications from GUIDE/UIL dialogs (`gduc.kb`) and the **G2 Dialog Configuration Editor (GDUE)**, which provides a native Windows editor for a native Windows dialog specification (`gdue.kb`).

- **G2 Data Source Manager (GDSM)** provides tools for managing network connections and for pooling connections to improve throughput in large-scale applications, including UIL and native configuration dialogs (`gdsm.kb`).

- **G2 Data Point Manager (GDPM)** provides functionality to configure, log, replay, and simulate datapoints, typically related to external sensors such as temperature, pressure, and flow. These external values are represented in GDPM as external datapoints and obtain their values typically via an OPC or PI interface and bridge (`gdpm.kb`).

- **G2 Engineering Unit Conversion (GEUC)** provides a way of specifying the engineering units for entering and displaying values, as well as a large number of synonyms for those conversions in both the English and metric systems (`geuc.kb`).

- **G2 Error Handling Foundation (GERR)** provides tools for error handling (`gerr.kb`).

- **G2 Relation Browser (GRLB)** provides tools for displaying related items in a graphical layout (`grlb.kb`).

G2 Bridges and Integration

The G2 Bundle includes the following bridges and integration tools:

- **G2 Gateway (GSI)** lets you develop two-way communication bridges between your G2 applications and dynamic external systems such as database management systems, C/C++ programs, and external simulation software. G2 Gateway gives your G2 application the real-time data it needs, and the ability to update the state of external systems.

    On UNIX, G2 Gateway is located in the `gsi` subdirectory.

On Windows, the G2 Bundle ships with two G2 Gateway directories, one called `gsi-intc`, which contains the GSI libraries and examples compiled with the Intel compiler, and the other called `gsi-msvc`, which contains the GSI libraries and examples compiled with Microsoft Visual Studio.

On HP-UX, G2 Gateway provides a separate component called G2 Gateway (GSI) for HP-UX Itanium, which is available on HP platforms only. This component installs into the `gsi-itanium` directory in the G2 bundle installation directory, a parallel directory to the `gsi` directory. The GSI libraries and examples in the `gsi-itanium` directory have been compiled on an HP Itanium machine running HP-UX 11.23, and are in the native ELF format. All other files in the G2 bundle on the HP platform are compiled and linked on a PA-RISC machine running HP-UX 11.00 and will run in compatibility mode on the HP Itanium machine under HP-UX 11.23.

- **G2 ActiveXLink** provides connectivity via ActiveX controls between G2 and a COM-compliant application running under Windows. G2 ActiveXLink integrates G2 applications with Microsoft Windows applications such as Microsoft Word, Excel, Visual Basic clients, and Explorer Web browsers.

  The sample files are located in the `activex` directory. The G2 ActiveXLink executable (`g2com.kb`) is located in the `g2\kbs\utils` directory on Windows and `g2/kbs/utils` on UNIX.

- **G2 JavaLink** lets you write G2 bridges entirely in Java. G2 JavaLink integrates G2 applications with other systems on the Internet or an organization's intranet. The components are located in the `javalink` directory.

- **G2 WebLink** lets you access G2 applications, using a Web browser via HTTP. The files are located in the `gw` directory.

- **G2 JMail Bridge** lets you send email messages from G2, using JMail. The files are located in the `jmail` directory.

- **G2-HLA Bridge** provides an interface to the Modeling and Simulation (M & S) High Level Architecture (HLA). The files are located in the `hls` directory.

- **Bridges**: G2 ships with a choice of two bridges. Bridge options include:

  - **G2-Oracle Bridge** (`oracle` directory)

  - **G2-Sybase Bridge** (`sybase` directory)

  - **G2-ODBC Bridge** (`odbc` directory)

  - **G2-OPC Client Bridge** (OLE for Process Control) (`opcclient` directory)

  - **G2 CORBALink** (Common Object Request Broker Architecture) (`corbalink` directory)

  - **G2-PI Bridge** (`pi` directory)

  - **G2 JMSLink** (`jms` directory)

- **G2 Java Socket Manager** (`sockman` directory)

- **G2 SNMP Bridge (**`snmp` directory)

G2 Applications

The following application product is part of the G2 bundle:

> **G2 Diagnostic Assistant (GDA)** is a visual programming environment for creating diagnostic, alarm-management, and advanced control applications that maintain the availability and performance of continuous and batch production processes. The files are located in the `gda` directory.

The following application products are available for purchase:

- **G2 e-SCOR** is a dynamic decision support and management tool for supply chains. You use it to design and analyze your current supply chain; to model alternative supply chains; and to perform "what-if" analysis to monitor, compare, and report on the performance of various alternative supply chains.

- **G2 ReThink** is a graphical simulation, analysis, and automation tool that enables decision-makers within complex organizations to visualize how their business process works, to analyze its performance, and to deploy it in real time.

- **G2 Optegrity** is an extensible software platform that is used to build abnormal condition management applications for process manufacturing industries. Optegrity applications ensure sustained operational performance and continuous availability of production assets. Its applications detect and resolve abnormal process conditions early—before they disrupt productivity, and weaken product quality and profits.

- **G2 Integrity** is an extensible software platform that is used to build network, system, service, and application management applications.

- **G2 SymCure** is a development and deployment environment for building and implementing fault management applications that automate real-time fault isolation, testing, repair, and availability management tasks of large-scale operations.

- **G2 NeurOn-Line** enables process-manufacturing companies in a variety of industries to realize substantial economic benefits from improved efficiency and product quality. Applications of the NOL platform include inferential measurements of product quality, model-based control, and process fault detection.

Summary of Features in the G2 Bundle Version 8.x

The primary focus of the G2 Bundle Version 8.x is to provide:

- Enhanced support for a Windows user interface, both for embedded Telewindows in COM-compliant, end-user applications, as well as for stand-alone Telewindows.

- Enhanced G2 server capabilities for application developers.

- Enhanced integration with other external standards such as Java and HTML.

G2 Bundle Version 8.3 Rev. 1

- **G2 Server**

  - G2 Graphical Language (G2GL) has a minor change.

  - The Dialog API has changes and enhancements in these areas:

    - Grid-View

    - Removing the Icon on MDI Child Windows

  - The G2 Bundle has changes on these platforms:

    - Windows Vista

    - HP Itanium

    - Linux

- **Bridges and Integration**

  - G2 Database Bridges modification to `g2-database.kb`.

  - G2 Gateway support on Linux.

G2 Bundle Version 8.3 Rev. 0

- **Stand-alone Telewindows**

  - Removed Support for Multiwindow Mode.

  - Text Editor enhancements.

- **G2 Server**
  - The Dialog API has changes and enhancements in these areas:
    - Custom Windows Dialogs Only Supported in Telewindows Next Generation
    - Custom Dialogs Use Windows Display Properties Fonts.
    - Calendar control.
    - Color-Picker control.
    - Combo-box control.
    - Duration control.
    - Grid-View control.
    - Image control.
    - Masked-Edit control.
    - Push-Button control.
    - Spinner control.
    - Tab-Frame control.
    - Tabular-View control.
    - Time-of-Day control.
    - Toggle-Button control.
    - Tree-View-Combo-Box control.
    - Programmatically Accepting Custom Dialogs.
    - Generic Dialog Callback.
    - Dialog Demo.
  - The Windows user interface has enhancements in these areas:
    - Chart View.
    - HTML View.
    - Property Grid.
    - Shortcut Bar.
    - Listbar-Style Shortcut Bar.
    - Status Bar.
    - Tree View.

- – Workspace.

- – Choosing Directories.

- – G2 has these platform-independent user interface enhancements:

  - – Mouse Hover Events.

  - – Custom Grammar in the G2 Text Editor.

  - – Programmatically Inserting Text into the G2 Text Editor.

  - – Visible Grid on Workspaces.

  - – Interactively Resizing Objects and Changing Connection Vertices.

  - – Font Changes on Texts.

- – G2 has these enhancements in the area of computational capabilities:

  - – Date and Time Formats for Attributes.

  - – Version Control Enhancements.

  - – Locking Mechanism for Objects.

  - – Summing Values in Histories.

  - – Referencing a Time Interval Ending with the Collection Time.

  - – Referring to the Current System Time.

  - – Confirming Run State Changes.

  - – Error Handling.

- – G2 provides new system procedures for:

  - – Clearing Histories.

  - – Getting Highly Precise Timestamps.

  - – Pinging a Network Host.

  - – Tracing a Network Host.

  - – Getting the IP Address of the G2 Server.

  - – Getting the G2 Process ID.

  - – Using the Explanation Facilities Programmatically.

  - – Generating a List of System-Defined Attributes.

  - – Saving and Loading KBs.

  - – Reading and Writing Bytes from and to a File.

  - – Upper Limit on UNIX Time Increased.

- – G2 provides new system procedures for networking and integration in these areas:
    - – Web Services.
    - – SOAP.
    - – HTTP.
    - – Network Socket Communication.
- – The G2 Bundle provides these changes:
    - – Platform Support.
    - – G2 Version 8.3 Requires New OK File.
    - – G2 OK File User Names and Passwords.
    - – Support for Secure Communication on UNIX.
    - – Log Command-Line Option on UNIX.
    - – Exit-on-Abort Command-Line Option.
    - – Windows Help.

- **Utilities**
    - – G2 Developer's Utilities
        - – Introduction to the G2 Developer's Utilities.
        - – Application Framework: G2 Developer's Utilities.
        - – Documentation.
    - – G2 Utilities
        - – No new features or changes.

- **Bridges and Integration**
    - – G2 ActiveXLink
        - – No new features or changes.
    - – G2 CORBALink
        - – No new features or changes.
    - – G2 Database Bridges
        - – Required Modules for G2-Database Bridges.
        - – "Smart Fetch".

– G2-Oracle Bridge supports the Oracle 10g server.

– G2-ODBC Bridge supports DNS Configuration via an RPC and Stored Procedures Returning Record Sets.

– G2 Gateway

  – Support for Secure Communication

  – G2 Gateway and Visual Studio 2005

  – G2 Gateway on HP-UX Itanium

– G2-HLA Bridge

  – Modeling and Simulation (M & S) High Level Architecture (HLA).

– G2 JavaLink

  – Support for secure communication.

– G2 Java Mail Bridge

  – Pinging the Bridge.

  – Changing the SNMP Port Number.

  – Sending MIME Email.

  – Disconnecting without Killing the Bridge Process.

  – Additional Documentation.

– G2 Java Socket Manager

  – Minor change in the `gsockdemo.kb` demo.

– G2 JMSLink

  – No new features or changes.

– G2-OPC Client Bridge

  – Complete rewrite of the bridge with most features implemented.

– G2-PI Bridge

  – Change in the shipping KBs and required modules of `g2-pi.kb`.

– G2-SNMP Bridge

  – New bridge product located in `snmp` directory.

– G2 WebLink

  – No new features or changes.

- **Applications**

    – G2 Diagnostic Assistant (GDA)

    – G2 e-SCOR

    – G2 ReThink

    – G2 Optegrity

    – G2 Integrity

    – G2 SymCure

    – G2 NeurOn-Line

Supported Platforms

Note The operating system requirements on some UNIX platforms have changed for G2 Version 8.3 Rev. 0. Please review the minimum requirements carefully.

G2 is available for the following platforms and operating systems (minimum requirements):

- **Windows** XP Pro, 2000, and 2003 running on a Pentium 4 computer.

- **Sun Solaris** 2.6 running on a Sun Sparc computer.

- **HP-UX** 11.0 or 11i V2 running on an HP9000 computer with the PA-RISC 2.0 architecture, and HP-UX 11.23 running on an HP computer with the Itanium artchitecture.

- **AIX** 5L V5.2 running on an IBM computer.

- **HP Tru64 UNIX** 5.1A (formerly known as DEC OSF/1) running on an HP Alpha computer

- **Red Hat Linux or Red Hat Enterprise Linux** running Linux kernel 2.4.26 or greater

The following platforms and operating systems are no longer supported: Windows NT 4.0 and IBM AIX 4.3.3.

For detailed information about specific G2 Bundle component availability on each of the platforms, see Chapter 2, "G2 Bundle Component Availability" on page 35.

Authorizing and Installing Your G2 Bundle Components

The G2 Bundle installation process automatically installs the required `g2.ok` (OK) files. Your product authorizations are enabled automatically when you enter your 25-character license key from the CD case during installation.

Note Before installing G2 products on any supported platform, carefully read the installation instructions on the CD liner for your platform and refer to the `readme-g2.html` file located at the top-level directory of the G2 Bundle CD.

These documents contain important platform-specific information and instructions on what to do before installing when you have a previous installation of the G2 Bundle on your machine.

Please note that the hardcopy version of the `readme-g2.html` file that is included with the software is the most up-to-date.

If you have questions about your installation, contact Gensym Customer Support at:

- `www.gensym-support.com`
- `support@gensym.com`
- 1-781-265-7301 (Americas) or +31-71-5682622 (EMEA)

Upgrading Your KBs

This section describes how to upgrade G2 Version 7.x or Version 6.x applications to G2 Version 8.3 Rev. 1. It also describes how to upgrade G2 Version 5.1 Rev. 0 through Version 5.2 Rev. 1 applications directly to G2 Version 8.3 Rev. 1.

Note Normally, G2 supports loading KBs from two release versions prior to the current release. For G2 Version 8.0 Rev. 0, this restriction has been relaxed to include KBs saved in G2 Version 5.1 Rev. 0 or higher.

Unlike upgrading from G2 Version 5.x to G2 Version 6.x, upgrading from G2 Version 6.x or 7.x to G2 Version 8.3 Rev. 1 does not require that you recompile your KBs.

Upgrading from Version 5.1 or 5.2 to Version 8.3, however, does require that you recompile your KBs. The instructions are the same as the previously published instructions for upgrading G2 5.x applications to G2 6.x.

Caution	You cannot load a KB that has been saved in G2 Version 8.3 into any earlier version of G2.

Throughout, the existing version of G2 from which you will be upgrading is referred to as G2 Version 7.x; however, keep in mind that the instructions apply to G2 Version 6.x KBs as well.

To obtain information about upgrading from any prior release of G2, please contact Gensym's Order Services department at 1-781-265-7106 (Americas) or +31-71-5682622 (EMEA) and request the G2 release notes for all releases prior to G2 Version 8.3.

Loading Your KB

If you are upgrading from G2 Version 7.x to G2 Version 8.3, you can simply load your G2 7.x KB directly into G2 Version 8.3 by automatically resolving all conflicts. You do not need to recompile your KBs.

If you are upgrading from G2 Version 5.1 or 5.2 directly to G2 Version 8.3, you must load your KBs into G2 Version 8.3 by automatically resolving conflicts, then you must recompile your KBs. See "Recompiling Your KB When Upgrading from G2 Version 5.1 or 5.2 to G2 Version 8.3" on page 16 and "Handling Text-Stripped KBs When Upgrading from G2 Version 5.1 or 5.2 to G2 Version 8.3" on page 17.

If your application requires any G2 utilities, such as GUIDE/UIL, GMS, GFR, GXL, GOLD, GDI, or GDD, you must copy the G2 Version 8.3 utilities into your application directory before you load, or use the new directory location of the G2 utilities in a module search path.

To load your KB:

➔   Choose Main Menu > Load KB and ensure that the automatically resolve conflicts option is selected, the default.

You are now finished upgrading your KBs from G2 Version 7.x to G2 Version 8.3.

Recompiling Your KB When Upgrading from G2 Version 5.1 or 5.2 to G2 Version 8.3

If you are upgrading from G2 Version 5.1 or 5.2 directly to G2 Version 8.3, you must recompile your KBs. When you load a G2 Version 5.1 or 5.2 KB into G2 Version 8.3, G2 automatically changes to incomplete the status of all items that need to be recompiled, such as rules, procedures, and methods. When the KB is loaded, G2 automatically launches the Inspect facility, which runs the recompile every item whose status is incomplete command. Thus, all such items are automatically recompiled during the load process for KBs running under a Development license.

If, during the recompile, G2 is unable to parse the text of an item, G2 places and ellipsis (...) in the text of the item, then displays the text editor with the offending text so you can change it.

If, during the recompile, G2 can parse the text but creates notes for the item, G2 displays a logbook message about the item.

After the load and recompile completes, we recommend that you use the Inspect facility to search for every item in your KB that has notes so you can correct any errors.

Handling Text-Stripped KBs When Upgrading from G2 Version 5.1 or 5.2 to G2 Version 8.3

If you are upgrading from G2 Version 5.1 or 5.2 directly to G2 Version 8.3, you must recompile all KBs built with earlier revisions. Since text-stripped KBs cannot be recompiled, you must obtain new versions of these KBs from your supplier or recompile the KBs from a version that contains source code. This includes third-party software that has been text-stripped.

If you need to upgrade an application that requires G2 Version 6.x or 7.x utilities, which are also text-stripped, you must use the Version 8.3 versions of the utilities.

Upgrade Issues Related to the Standard User Interface

When loading your existing KB into G2 Version 7.0 and higher, you may find that various aspects of the new user interface interfere with the operation of your KB's own user interface, or that users can access parts of your KB that were formerly hidden.

Note This section refers to user interface issues for G2 Version 7.0 and higher.

If you do not require any of the new user interface features; if all you need is the ability to configure commands for left, middle, and right mouse buttons and the mouse wheel; or if you simply prefer the classic user interface, then the easiest approach is to start G2 or Telewindows with the command line option `-ui classic`. This option disables all features of the new user interface except for the new item-configuration grammar.

If you require other features of the new user interface, then you cannot use the `-ui classic` option. The following sections provide some background information and workarounds for some of the problems that might arise when running with the default `-ui standard` or `-ui multiwindow` user interface.

Terminology

The term "selecting" is overloaded in Version 7.0. In previous releases of G2, "selecting" appeared only in item configurations, and referred to pressing and releasing a mouse button over an item without moving the mouse in between. This interpretation of selection is still valid for item configurations in Version 7.0. The documentation refers to this type selection as "sense one."

In G2 Version 7.0, there is an additional, unrelated meaning of "selecting," which means to cause an item to become a member of the set of selected objects in a window. This meaning of selection is bound to the gesture of pressing a mouse button, that is, the selection happens when the button is pressed rather than waiting for the button to be released. You can refer to this type of selection as "sense two."

Security

Running G2 Version 7.0 with the standard user interface provides some new user-interface gestures for invoking commands. Because G2 item configuration generally restrict only the "binding" of a command to a gesture, the new gestures are not necessarily affected by existing item restrictions. As a result, it might be possible for users to access parts of your KB in Version 7.0 that were not accessible in previous releases of G2.

For example, in previous releases of G2, an item configuration of the form selecting any FOO does nothing would effectively prevent access to the item's table, because it would prevent access to the item's menu.

However, in Version 7.0, pressing a mouse button on the item selects the item, in sense two. Once the item is selected, pressing the Enter key pops up that item's table. This behavior occurs even if you have an item restriction such as menu choices for FOO exclude: table.

Usability

The selection user interface might interfere with various user interface elements in your KB. For example, the action of selecting an item, in sense two, is bound to the gesture of pressing the left mouse button. An item configuration such as selecting any item does nothing does not trigger when you press the left mouse button, because "selecting" in this context, sense one, means pressing and releasing the mouse button without moving the mouse. This gesture is also called "clicking" the mouse.

As a result, "selecting" an item in sense one, that is, pressing and releasing a mouse button over the item, no longer does nothing. Instead, it "selects" the item, in sense two.

Once the item is selected, the user can access various commands via keyboard gestures, as described above. Also, the selected item is highlighted with a green border, by default, which might be undesirable.

Include versus Exclude Item Restrictions for Non-Menu Choices

You may find that some of the items in your existing KB are selectable in Version 7.0 and some are not, without any clear reason for the difference. The difference can be explained by the style of item restrictions you are using on non-menu choices.

Recall that an item restriction that uses include specifies the entire list of operations that are permitted on an item. For example:

> non-menu choices for any item include: move-object

G2 Version 7.0 introduces a new non-menu choice, select-object, which determines whether an item is selectable, in sense 2. Since the restriction above uses include and does not mention select-object, selecting the item, in sense two, is implicitly not permitted. Thus, items covered by this restriction are not selectable in Version 7.0.

On the other hand, your KB might be using exclude item restrictions, such as:

> non-menu choices for any item exclude: click-to-edit

This restriction disables just the click-to-edit operation, which means the select-object operation is implicitly permitted. Thus, items covered by this restriction are selectable in Version 7.0

Workarounds

Here is a list of potential problems and their workarounds. Each workaround is generally in the form of a clause you can add to an appropriate item configuration. If you want the workaround to apply to your entire KB, add the clause to the KB Configuration system table.

- Disable selection of a particular item, class of item, entire workspace, and so on. This restriction is the primary way to selectively disable selection, in sense two:

> non-menu choices for item exclude: select-object

- Disable the selection rubber band on a workspace:

> non-menu choices for this workspace exclude: select-area

- Restore workspace dragging with the left mouse button:

> pressing the left mouse button on any kb-workspace implies move

- Disable moving a workspace with the mouse wheel:

  rolling the mouse wheel forwards does nothing;
  rolling the mouse wheel backwards does nothing

- Hide the menu bar (this workaround requires sys-mod):

  start g2-nms-hide-menu-bar(g2-window)

- Hide the status bar:

  Choose View > Status Bar from the top-level menu bar. Note that
  G2 Version 7.0 does not support hiding the status bar programatically.

- Disable various new keystrokes, such as Return and Tab:

  typing return does nothing;
  typing tab does nothing;

- Remove the green highlight box while still allowing selection, in sense two.
  Note that this approach removes user feedback, so it is probably undesirable.

  In the Drawing Parameters system table, change primary-selection-color
  and secondary-selection-color to transparent.

- An item that you want to be selectable in sense two is not. The item probably
  has an active include restriction for non-menu choices.

  Add select-object to the list of non-menu choices included for that item.

Determining Client Features

Due to the recent UI enhancements, various Telewindows clients now may have
vastly differing UI capabilities. Depending on how it is deployed, a G2 server
may need to handle connections from any Telewindows version, adapting its UI
to suit. Alternatively, it may require a specific set of Telewindows features and
choose to refuse connections when they are not present.

The following attributes and system procedures may be helpful in determining a
client's capabilities.

G2-Window Attributes

- g2-window-ui-style: classic, standard, multiwindow

  – classic: No selection UI, no standard Windows menus, dialogs, controls,
    etc.

  – standard: Selection UI, standard Windows menus and dialogs

  – multiwindow: Selection UI, standard Windows menus, dialogs, scroll
    bars, workspaces as MDI child windows.

- g2-window-operating-system-type = unix, win32

   Currently, only Windows clients support standard Windows menus and controls.

- g2-window-is-embedded: true or false

   An embedded client window means it is a Web browser plug-in, an ActiveX control, or workspace view. Embedded clients do not permit standard Windows menu bars. If an embedded client also has the `multiwindow` UI style, then it is capable of displaying workspace views.

System Procedures

g2-nms-is-supported(g2-window)

g2-nms-version()

g2-ui-dialog-is-supported(type, g2-window)

Note that the dialog type custom is supported only in Telewindows Version 8.3 Rev. 0 or higher running on Windows, so that can be used as a test for a Version 8.3 client.

Toolbars

In Telewindows Version 8.0 Rev. 0, standard Windows toolbars are supported only by Telewindows Next Generation (`twng.exe`). There is no system procedure to test directly whether a client is `twng` or whether it supports toolbars, but you can effectively test this by attempting to create a toolbar and immediately deleting it if successful. The following G2 procedure is an example of this:

```
client-supports-toolbars(Win: class g2-window) = (truth-value)
tb: integer;
e: class error;
begin
    begin
        tb = call g2-nms-create-toolbar("Dummy", the symbol dummy, structure(),
            Win);
        end on error (e)
        delete e;
        return false;
    end;
    call g2-nms-delete-control(tb, Win);
    return true;
end
```

Example: Rejecting Connections from Certain Clients

Here is an example rule and pair of procedures that you can use to reject connections from any Telewindows except `twng` Version 8.0 Rev. 0 or higher.

Rule

whenever the g2-connection-status of any g2-window Win receives a value
then start check-client(Win)

Procedure

```
{ Reject connections if we have a reason. }
check-client(Win: class g2-window)
reason: text;
begin
    { Do nothing if called on an unconnected window. }
    if (the g2-connection-status of Win is not CONNECTED) then return;

    reason = call reason-to-reject-client(Win);
    if(reason = "") then
        post "Accepted connection from [the g2-window-remote-host-name of Win]."
    else
    begin
        post "Rejected [reason] connection from [the g2-window-remote-host-name of
            Win].";
        call g2-system-command(the symbol
            CLOSE-TELEWINDOWS-CONNECTION, Win, Win, the symbol NONE)
    end;
end
```

Procedure

```
{ Return the reason to reject this connection, or "" to permit it. }
reason-to-reject-client(Win: class g2-window) = (text)
torf: truth-value;
begin

    if (the g2-window-ui-style of Win is not MULTIWINDOW) then
        return "[the g2-window-ui-style of Win]";

    if (the g2-window-is-embedded of Win) then
        return "embedded";

    torf = call g2-ui-dialog-is-supported(the symbol CUSTOM, Win);
    if (torf is false) then
        return "pre-8.0r0";

    torf = call client-supports-toolbars (Win);
    if (torf is false) then
        return "non-TWNG";

    return "";
end
```

Known Limitations

The following are known limitations of Telewindows Version 8.0 Rev. 0 and higher:

- Displaying G2 or Telewindows on a Windows platform may cause an abort when the available memory for graphical objects is exhausted. In the Task Manager, this task is called Paged Kernel Memory. Displaying or zooming in on a very large icon can cause this situation to occur. Work around: Use Telewindows for display. A Telewindows abort should not affect the G2 server. Adding more physical RAM to your machine might also help.

- You can no longer set the g2-window-specific-language of a g2-window to no-value.

- GMS allows you to vary the size and colors of a menu item icon, via the gms-inline-icon-description attribute, effectively using an instance icon instead of a class icon. NMS permits only class icons, so this feature of GMS is not supported in native GMS when viewed through Telewindows.

Migrating Applications to a Standard Windows User Interface

This section describes general practices for migrating G2 classic applications to standard Telewindows on Windows platforms to take advantage of the G2 7.x and G2. 8.x standard user interface features.

Automatic Migration Capabilities

Many of the standard Windows user interface features are automatically available simply by viewing your application through Telewindows on a Windows platform. These features include:

- A multiple document interface (MDI) application in which workspaces appear in their own windows with scrollbars.

- Standard Windows developer menu bar, popups, selection-style user interface, and keystroke commands and mouse gestures.

- Standard file and print dialogs.

Migrating End User Applications

Migrating end user applications that include menus and dialogs requires varying amounts of additional effort, depending on how these applications have been implemented:

- If your application uses Gensym Menu System (GMS) menus, simply viewing the application through Telewindows on a Windows platform renders these menus, using standard Windows menus. This capability is part of the Native Menu System (NMS).

- You can use the Native Menu System (NMS) API to create standard Windows menus that support these features: menu bars, submenus, popup menus, dynamic menus, alternate menu bars, localization, checked and radio menu choices, menu groupings and separators, icons, color, character underling, enabling and disabling menu choices, and callbacks.

- You can use G2 system procedures to create standard Windows dialogs. G2 supports built-in dialog types—basic, query, notification, delay notification, file, and print—as well as custom dialogs, which you can build using standard Windows controls. It also supports a grid view control for creating custom message browsers.

- You can use G2 system procedures to create standard Windows toolbars that support these features: tool tips, dockable toolbars, and floating toolbars.

Using G2 Developer Utilities

In addition to the automatic and Native Menu System (NMS) capabilities, you can use the G2 Developer's Utilities modules to support these user interface capabilities:

- Operator message and event management, using message browsers and message queues.

  See the *G2 Event Manager User's Guide*.

- A comprehensive object model that provides:

  - Default implementations for various methods for items, including constructors, destructors, and initialization methods, as well as procedures for cloning, creating, and deleting items.

  - Object keys that provide a universal naming/ID system.

  - Object properties that provide virtual attributes for items and the ability to serialize and de-serialize objects for integration with XML, for example.

  - A standard event notification model, which you can define on a class or instance level.

  - Object repositories for dynamically created objects.

- Datapoints that get their values from source datapoints, such as external DCS systems.

- Domain objects and process maps used to represent external processes graphically and hierarchically.

- Schemes used to implement logic used to validate datapoints, monitor situations, detect events, perform diagnostics, or perform actions.

  See Part IV "Gensym Runtime Library Object Model" in the *G2 Run-Time Library User's Guide*.

- Numerous user interface features, including: posting standard Windows or UIL dialogs based on the window type, posting operator messages, creating palette groups, defining user preferences, and managing connections.

  See Part III "Gensym Runtime Library User Interface Operations" in the *G2 Run-Time Library User's Guide*.

- Numerous additional application features, including: localization, application configuration, command-line options, time and text manipulation, file and directory management, module management, performance metrics, and list, vector, and sequence manipulation,

  See Part II "Gensym Runtime Library Utilities" in the *G2 Run-Time Library User's Guide*.

Specific Notes on Migrating Applications

Here are some specific notes on various aspects of migrating existing applications to use standard Windows capabilities:

- Do not mix GMS menus and menus created using the Native Menu System (NMS) API. You must use one technique or the other, but not both.

- If your application needs to support both classic G2 and standard Telewindows dialogs, use the g2-ui-dialog-is-supported system procedure to determine whether the g2-window instance that you pass as an argument to the API procedure that post dialogs is for the server or for a Telewindows client. You can also use the GRTL API procedures related to dialogs, which post the appropriate type of dialog automatically, based on the type of window.

- If your application uses gms-enable-entry and gms-disable-entry for enabling and disabling GMS menu items, based on a selection, you need to test whether the selected item is a kb-workspace in the server. This is because GMS uses kb-workspaces as menus, which are not selected in the Telewindows client. Thus, if you try to use any of the selection APIs, the selected items will return an empty sequence. Also, there are cases where the selected workspace is not a kb-workspace, such as the Message Board. You

can use the following code to get a handle on the selected workspace, if one exists:

```
if the selected-workspace Wksp of Win exists and Wksp is a kb-workspace
then begin
    …
end;
```

Online Documentation

The G2 Bundle is distributed with online documentation in the form of PDF and HTML files. You can access the HTML files directly from G2, using GOLD. For information on using GOLD to access online documentation, see the *G2 OnLine Documentation User's Guide*.

On Windows platforms, a subset of the G2 Bundle documentation is also distributed in the form of Windows HTML Help, which you can access directly through Telewindows.

The online documentation distributed with the G2 Bundle Version 8.3 corresponds with the current shipping version. These release notes provide a summary of the new features and changes in the G2 Bundle Version 8.3.

The default installation directory for all documentation files is:

* Windows: `C:\Program Files\Gensym\g2-8.3r1\doc\`

* UNIX: `/usr/gensym/g2-8.3r1/doc/`

The `readme-g2.html` file provides links to the PDF versions of all online documentation.

On Windows platforms, you can access Windows HTML Help directly from the Telewindows Help menu. The online help includes G2 core and G2 utility documentation in Windows help format.

Also on Windows platforms, you can access PDF and HTML versions of the documentation by choosing Programs > Gensym G2 8.3r1 > Documentation or Documentation (HTML), respectively, from the Start menu, which is organized into the following submenus:

* Bridges

* G2

* G2 Developer's Utilities

* G2 JavaLink

* GDA

* Utilities

Note Due to a display bug in earlier versions of Adobe Reader, we recommend that you use Adobe Acrobat Reader Version 5.05 or later for viewing PDF files.

The *G2 Bundle Release Notes* (this document) and readme files for the G2 Bundle and for G2 JavaLink are also available.

Online Documentation Location

The PDF and HTML documentation is installed under the following directories, by default:

Directory Name (Windows Style)	Contents
`\g2-8.3r1\readme-g2.html`	Readme file for the G2 Bundle
`\g2-8.3r1\doc`	*G2 Bundle Release Notes*
`\g2-8.3r1\doc\g2`	*Getting Started with G2 Tutorials* *G2 Reference Manual* *G2 System Procedures Reference Manual* *G2 System Procedures Reference Card* *G2 Developer's Guide* *Telewindows User's Guide* *G2 Class Reference Manual* *G2 Gateway Bridge Developer's Guide* *G2 Language Reference Card*
`\g2-8.3r1\doc\utilities`	*G2 ProTools User's Guide* *G2 Dynamic Displays User's Guide* *G2 Developer's Interface User's Guide* *G2 Foundation Resources User's Guide* *G2 Menu System User's Guide* *G2 OnLine Documentation User's Guide* *G2 OnLine Documentation Developer's Guide* *G2 GUIDE User's Guide* *G2 GUIDE/UIL Procedures Reference Manual* *G2 XL Spreadsheet User's Guide*

Directory Name (Windows Style)	Contents
\g2-8.3r1\doc\g2i	*Business Process Management System User's Guide* *Business Rules Management System User's Guide* *G2 Reporting Engine User's Guide* *G2 Web User's Guide* *G2 Event and Data Processing User's Guide* *G2 Run-Time Library User's Guide* *G2 Event Manager User's Guide* *G2 Dialog Utility User's Guide* *G2 Data Source Manager User's Guide* *G2 Data Point Manager User's Guide* *G2 Engineering Unit Conversion User's Guide* *G2 Error Handling Foundation User's Guide* *G2 Relation Browser User's Guide*
\g2-8.3r1\doc\javalink\ docs\guides	*G2 JavaLink User's Guide* *G2 Bean Builder User's Guide* *G2 DownloadInterfaces User's Guide*
\g2-8.3r1\doc\javalink \docs\api\index.html	G2 JavaLink API documentation as Javadoc
\g2-8.3r1\javalink\ readme-javalink.html	G2 JavaLink readme file
\g2-8.3r1\doc\gda\	*GDA User's Guide* *GDA Reference Manual* *GDA API Reference*
\g2-8.3r1\doc\bridges	*G2 ActiveXLink User's Guide* *G2 CORBALink User's Guide* *G2 Database Bridge User's Guide* *G2-ODBC Bridge Release Notes* *G2-Oracle Bridge Release Notes* *G2-Sybase Bridge Release Notes* *G2-HLA Bridge User's Guide* *G2 JMail Bridge User's Guide* *G2 Java Socket Manager User's Guide* *G2 JMSLink User's Guide* *G2-OPC Client Bridge User's Guide* *G2-PI Bridge User's Guide* *G2-SNMP Bridge User's Guide* *G2 WebLink User's Guide*

Windows Help

In addition to the PDF and HTML files, the G2 Bundle provides Windows help
(.chm) files for all the available online documentation, as follows:

Getting Started

G2 Bundle Release Notes
Getting Started with G2 Tutorials

G2 Core

G2 Reference Manual
G2 System Procedures Reference Manual
G2 System Procedures Reference Card
G2 Developer's Guide
Telewindows User's Guide
G2 Class Reference Manual
G2 Language Reference Card

G2 Utilities

G2 Dynamic Displays User's Guide
G2 Developer's Interface User's Guide
G2 Foundation Resources User's Guide
G2 Menu System User's Guide
G2 OnLine Documentation User's Guide
G2 OnLine Documentation Developer's Guide
G2 ProTools User's Guide
G2 GUIDE User's Guide
G2 GUIDE/UIL Procedures Reference Manual
G2 XL Spreadsheet User's Guide

G2 Developer's Utilities

Business Process Management System User's Guide
Business Rules Management System User's Guide
G2 Reporting Engine User's Guide
G2 Web User's Guide
G2 Event and Data Processing User's Guide
G2 Run-Time Library User's Guide
G2 Event Manager User's Guide
G2 Dialog Utility User's Guide
G2 Data Source Manager User's Guide
G2 Data Point Manager User's Guide
G2 Engineering Unit Conversion User's Guide
G2 Error Handling Foundation User's Guide
G2 Relation Browser User's Guide

G2 Bridges and Integration

G2 ActiveXLink User's Guide
G2 CORBALink User's Guide
G2 Database Bridge User's Guide
G2-ODBC Bridge Release Notes
G2-Oracle Bridge Release Notes
G2-Sybase Bridge Release Notes
G2-HLA Bridge User's Guide
G2 JMail Bridge User's Guide
G2 Java Socket Manager User's Guide
G2 JMSLink User's Guide
G2-OPC Client Bridge User's Guide
G2-PI Bridge User's Guide
G2-SNMP Bridge User's Guide
G2 WebLink User's Guide

You access Windows HTML Help through the standard Telewindows interface on a Windows machine. To access online help, choose Help > Help Topics from the top-level menu bar. The online help looks like this:

Help

Hide   Back   Print   Options

Contents | Index | Search

- G2 Online Help
  - G2 Online Manuals
  - For More Information
  - Getting Started
    - G2 Bundle Release Notes
    - Getting Started with G2 Tutorials
  - G2 Core
    - G2 Reference Manual
    - G2 System Procedures
    - G2 System Procedures Reference Card
    - G2 Developer's Guide
    - Telewindows
    - G2 Class Reference
    - G2 Language Reference
  - G2 Utilities
    - G2 ProTools
    - G2 Dynamic Displays (GDD)
    - G2 Developer's Interface (GDI)
    - G2 Foundation Resources (GFR)
    - G2 Menu System (GMS)
    - G2 Online Documentation (GOLD)
    - G2 Online Documentation Developer's Guide (GOLD)
    - G2 User Interface Development Environment (GUIDE)
    - G2 User Interface Development Environment/User Interface
    - G2 XL Spreadsheet (GXL)
  - G2 Developers' Utilities
    - Business Process Management System (BPMS)
    - Business Rules Management System (BRMS)
    - G2 Reporting Engine (GRPE)
    - G2 Web (GWEB)
    - G2 Event and Data Processing (GEDP)
    - G2 Run-Time Library (GRTL)
    - G2 Event Manager (GEVM)
    - G2 Dialog Utility (GDU)
    - G2 Data Source Manager (GDSM)
    - G2 Data Point Manager (GDPM)
    - G2 Error Handling Foundation (GERR)
    - G2 Engineering Unit Conversion (GEUC)
    - G2 Relation Browser (GRLB)
  - G2 Bridges and Integration
    - G2 ActiveXLink
    - G2 CORBALink
    - G2-Database Bridge
    - G2 Gateway
    - G2 HLA
    - G2 JavaLink
    - G2 JMail Bridge
    - G2 JMSLink
    - G2 Java Socket Manager
    - G2-ODBC Bridge
    - G2 OPCLink
    - G2-Oracle Bridge
    - G2-PI Bridge
    - G2-SNMP Bridge
    - G2-Sybase Bridge
    - G2 WebLink

G2 Online Help

To access online help, click a topic on the Contents tab, click a topic on the Index tab, enter a keyword on the Search tab, or click a topic below:

Getting Started

G2 Bundle Release Notes provides a description of the G2 Bundle, and the new features and changes in the current release.

Getting Started with G2 Tutorials provide a step-by-step tutorial for new users of G2. The tutorials provide simple, online applications that you can use to begin working with G2.

G2 Core

G2 Reference Manual provides a comprehensive reference for creating, testing, and deploying G2 applications. It describes the developer's environment and other global components, knowledge representation, computational capabilities, user interface components, editors and facilities, debugging and optimization, application deployment, networking and interfacing, launching a G2 process, keyboard shortcuts, and more.

G2 System Procedures provides API procedures that you can use to develop G2 applications. Many categories of system procedures exist, such as file and directory operations, connection operations, history clearing operations, inspect operations, KB and module operations, object passing operations, user interface operations, and more.

G2 System Procedures Reference Card provides for a summary of all G2 system procedures.

G2 Developer's Guide provides a broad overview of G2's features and capabilities. This guide discusses many aspects of how to develop successful G2 applications. The scope of this online guide covers issues that span the entire software development lifecycle from the planning phase, through the implementation phase, to the deployment phase. It emphasizes generic standards and guidelines for developing reusable modules within large organizations and domain-specific toolkits.

Telewindows allows multiple users to access a G2 knowledge base remotely through a client. On Windows platforms, the Telewindows client provides a standard Windows user interface for G2 developers and end users.

G2 Class Reference provides a complete reference of every attribute of every G2 class, for use in conjunction with the attribute access facility. This reference is for advanced G2 users only.

G2 Language Reference provides a syntax summary for the G2 language, including actions, procedure syntax and statements, rules syntax, generic references, expressions, functions, and Inspect syntax.

G2 Utilities

G2 Dynamic Displays (GDD) gives you the ability to add dynamically generated graphical attribute readouts to icons in your applications.

Master.chm file, shown above, is the top-level help file from which you access all available Windows help. The .chm files are located in the g2 directory.

Online Documentation through GOLD

The HTML files that make up the GOLD book objects are installed in `html` directories under each product directory in the `doc` directory. For example, the HTML files for the *G2 Reference Manual* are installed in `\doc\g2\g2refman\html` on Windows and `/doc/g2/g2refman/html` on UNIX. Each `html` directory has a file named `titlepag.htm`, which displays the title page of the book and provides access to the table of contents and index.

To access the online documentation through GOLD, you must load the various modules that provide GOLD book objects, then configure your machine for online help, as described in detail in the *G2 OnLine Documentation User's Guide*. The default path name for each book object points to the installation directories shown above for each online book; therefore, unless you install the G2 Bundle in a directory other than the default, you should not have to configure the default path names for the GOLD books.

G2 Version 8.3 provides GOLD book objects for all available online documentation. By loading `starter.kb`, located in the `g2\kbs\utils` directory on Windows and in the `g2/kbs/utils` directory on UNIX, you have access to all online books defined for G2, GOLD, GFR, GUIDE, GDI, all the bridges, all the G2 Developer's Utilities, and G2 JavaLink.

To access online books for the utilities that are not included when you load `starter.kb`, namely GDD, GMS, and GXL, you can load the starter module, then merge the gddroot, gms, and/or gxl modules into your application. To access the online books for GDA, you can simply load the gda module, which requires the goldui module. If you do not want to provide access to all of the online books in the starter module, you can simply load the goldui module and merge the other modules into that module.

Note   To access any online documentation through GOLD, the goldui module must always be loaded.

This table defines the online books that are available in each module, where the indented modules are required modules of the **starter** module:

Module	Available Online Books
starter	*Getting Started with G2 Tutorials*
	Telewindows User's Guide
	G2 ProTools User's Guide
	G2 ActiveXLink User's Guide
	G2 CORBALink User's Guide
	G2 Database Bridge User's Guide
	G2-ODBC Bridge Release Notes
	G2-Oracle Bridge Release Notes
	G2-Sybase Bridge Release Notes
	G2-HLA Bridge User's Guide
	G2 JMail Bridge User's Guide
	G2 JMSLink User's Guide
	G2 Java Socket Manager User's Guide
	G2-OPC Client Bridge User's Guide
	G2-PI Bridge User's Guide
	G2-SNMP Bridge User's Guide
	G2 WebLink User's Guide
	G2 JavaLink User's Guide
	G2 Bean Builder User's Guide
	G2 DownloadInterfaces User's Guide
	Business Process Management System User's Guide
	Business Rules Management System User's Guide
	G2 Reporting Engine User's Guide
	G2 Web User's Guide
	G2 Event and Data Processing User's Guide
	G2 Run-Time Library User's Guide
	G2 Event Manager User's Guide
	G2 Dialog Utility User's Guide
	G2 Data Source Manager User's Guide
	G2 Data Point Manager User's Guide
	G2 Engineering Unit Conversion User's Guide
	G2 Error Handling Foundation User's Guide
	G2 Relation Browser User's Guide
goldui	*G2 OnLine Documentation User's Guide*

Module	Available Online Books
gold	*G2 Reference Manual* *G2 Developer's Guide* *G2 Class Reference Manual* *G2 System Procedures Reference Manual* *G2 System Procedures Reference Card* *G2 Gateway Bridge Developer's Guide* *G2 OnLine Documentation Developer's Guide*
gfr	*G2 Foundation Resources User's Guide*
uillib	*G2 GUIDE User's Guide* *G2 GUIDE/UIL Procedures Reference Manual* *G2 Developer's Interface User's Guide*
gddroot	*G2 Dynamic Displays User's Guide*
gms	*G2 Menu System User's Guide*
gxl	*G2 XL Spreadsheet User's Guide*
gda	*GDA User's Guide* and *Reference Manual* *GDA API Reference*

Online Search and Context-Sensitive Help

All of the online books support an online search capability through the GOLD Help dialog, as described in the *G2 OnLine Documentation User's Guide*.

Various books also support context-sensitive help (F1), including:

- *G2 Reference Manual*

- *G2 System Procedures Reference Manual*

- All G2 utilities guides, except *G2 GUIDE/UIL Procedures Reference Manual* and *G2 ProTools User's Guide*.

- *GDA User's Guide*, *GDA Reference Manual*, and *GDA API Reference*

Online Documentation through HelpLink

The HTML files are also available for viewing directly on HelpLink. To access these files, visit www.gensym-support.com, then choose Download Product Information and Software > On-Line Documentation > G2 Bundle.

G2 Bundle
Component Availability

Describes the availability of each G2 component on each of the supported platforms.

Introduction

The following table presents the availability of each G2 Bundle component on each operating system that G2 supports. It also discusses issues with upgrading operating systems. In all cases, the operating system version refers to the specified version or higher, except where noted for specific components.

For system requirements, see the `readme-g2.html` file.

G2 Version 8.3 Rev. 1 Bundle Component Availability						
	Windows Vista/XP Pro/ 2003/2000 (a)	**Sun Solaris 2.6 (b)**	**IBM AIX 5L V5.2**	**HP-UX 11.0, 11i V2, or 11.23 (c)**	**Red Hat Linux (d)**	**HP Tru64 UNIX 5.1A**
G2 Version 8.3 Rev. 1 including G2 utilities	Available	Available	Available	Available	Available	Available
Telewindows Version 8.3 Rev. 1	Available	Available	Available	Available	Available	Available
G2 Gateway (GSI) Version 8.3 Rev. 1	Available	Available	Available	Available	Available (e)	Available
G2 Developer's Utilities Version 2.3 Rev. 1	Available	Available	Available	Available	Available	Available
G2 ActiveXLink Version 8.3 Rev. 1	Available	Not Available	Not Available	Not Available	Not Available	Not Available
G2 CORBALink Bridge Version 8.3 Rev. 1	Available	Available	Available	Available	Available	Not Available
G2-HLA Bridge Version 8.3 Rev. 1	Available	Not Available	Not Available	Not Available	Not Available	Not Available
G2 JavaLink Version 8.3 Rev. 1	Available	Available	Available	Available	Available	Available
G2 Java Mail Bridge Version 2.3 Rev. 1	Available	Available	Available	Available	Available	Available
G2 Java Socket Manager Version 4.2 Rev. 1	Available	Available	Available	Available	Available	Available
G2 JMSLink Version 2.3 Rev. 1	Available (f)	Not Available	Not Available	Available (f)	Available (f)	Not Available
G2-ODBC Bridge Version 8.3 Rev. 1	Available	Not Available	Not Available	Not Available	Not Available	Not Available

G2 Version 8.3 Rev. 1 Bundle Component Availability					
	Windows Vista/XP Pro/ 2003/2000 (a)	**Sun Solaris 2.6 (b)**	**IBM AIX 5L V5.2**	**HP-UX 11.0, 11i V2, or 11.23 (c)**	**Red Hat Linux (d)**	**HP Tru64 UNIX 5.1A**
G2-OPC Client Bridge Version 8.3 Rev. 1	Available	Not Available	Not Available	Not Available	Not Available	Not Available
G2-Oracle Bridge Version 8.3 Rev. 1 for Oracle 7.3 client	Available	Available	Available	Available (g)	Not Available	Available
G2-Oracle Bridge Version 8.3 Rev. 1 for Oracle 8.0 client	Available	Available	Available	Available (g)	Available	Available
G2-Oracle Bridge Version 8.3 Rev. 1 for Oracle 8.1 client	Available	Available	Available	Available (g)	Available	Available
G2-Oracle Bridge Version 8.3 Rev. 1 for Oracle 9i with the Oracle 9.0 client	Available	Available on Solaris 2.8	Not Available	Available (g)	Available	Available
G2-Oracle Bridge Version 8.3 Rev. 1 for Oracle 8i or 9i with the Oracle 9.2 client	Available	Available on Solaris 2.8	Available	Available (g)	Available	Available
G2-Oracle Bridge Version 8.3 Rev. 1 for Oracle 10g with the Oracle 10g client	Available	Available on Solaris 2.8	Available	Available (g)	Available	Available
G2-PI Bridge Version 8.3 Rev. 1	Available	Not Available	Not Available	Not Available	Not Available	Not Available
G2-SNMP Bridge Version 4.0 Rev. 1	Available	Available	Not Available	Not Available	Not Available	Not Available

G2 Version 8.3 Rev. 1 Bundle Component Availability						
	Windows Vista/XP Pro/ 2003/2000 (a)	**Sun Solaris 2.6 (b)**	**IBM AIX 5L V5.2**	**HP-UX 11.0, 11i V2, or 11.23 (c)**	**Red Hat Linux (d)**	**HP Tru64 UNIX 5.1A**
G2-Sybase Bridge Version 8.3 Rev. 1 for Sybase 12.5 libraries	Available	Available	Available	Available (g)	Not Available	Available
G2-Sybase Bridge Version 8.3 Rev. 1 for Sybase 11.1 libraries	Available	Available	Available	Available (g)	Not Available	Available
G2 WebLink Version 8.3 Rev. 1	Available	Available	Available	Available	Available	Available
G2 Diagnostic Assistant (GDA) Version 5.1 Rev. 1	Available	Available	Available	Available	Available	Available
G2 e-SCOR Version 5.1 Rev. 1	Available	Not Available	Not Available	Not Available	Not Available	Not Available
G2 ReThink Version 5.1 Rev. 1	Available	Not Available	Not Available	Not Available	Not Available	Not Available
G2 Optegrity Version 5.1 Rev. 1	Available	Not Available	Not Available	Not Available	Not Available	Not Available
G2 Integrity Version 5.0 Rev. 1	Available	Available	Not Available	Available	Not Available	Not Available
G2 SymCure Version 5.1 Rev. 1	Available	Not Available	Not Available	Not Available	Not Available	Not Available
G2 NeurOn-Line Version 5.1 Rev. 1	Available	Not Available	Not Available	Not Available	Not Available	Not Available

(a) Requires Pentium 4 computer.

(b) Requires Sparc computer.

(c) HP-UX 11.0 and HP-UX 11i V2 both run on the PA-RISC 2.0 architecture; HP-UX 11.23 requires the Itanium architecture.

(d) Supports Red Hat Linux or Red Hat Enterprise Linux running Linux 7.3 kernel 2.4.26 or greater.

(e) The G2 Bundle ships with an additional G2 Gateway directory called `gsi-redhat4`, which contains the GSI libraries and examples compiled with Redhat Enterprise Linux 4 with kernel 2.6.9-55.EL.

(f) G2 JMSLink works with JBoss 3.2.6 on Linux and HP; Open JMS 0.7.6.1 on Windows, Linux, and HP PA-RISC; and Java 2 Platform, Enterprise Edition (J2EE) 1.3.1 on Linux.

(g) The database bridges for HP Itanium are produced on PA-RISC and run in emulation mode on the Itanium.

Upgrading Your Operating System

Gensym software rarely encounters problems when moved to a newer version of an operating system. Maintaining upward compatibility is a high priority for operating system vendors. It has been our experience that vendors essentially always meet this goal.

When incompatibilities are introduced in operating system upgrades, Gensym is usually notified by the operating system supplier, because it is considered a serious concern. If this occurs, Gensym customers affected by the problem are notified through Gensym field offices.

Part II

G2

Chapter 3, 4   G2 Version Update Summary

List of Changes – 84R2

Issues Fixed

Issue no	Issue Description
Gensym-304	ODBC bridge aborts when a sql query using a bind variable prefix is passed to the procedure db-sql-function, and no bind variable is defined. The abort is fixed by implementation of error handling.
Gensym-46	Grid performance seems slow when initially adding rows to the grid whene there is a large number of rows (>1000). Two new APIs g2-ui-grid-view-insert-rows and g2-ui-grid-view-update-rows in a grid-view are provided.
Gensym-10379	G2 database bridge based on OLEDB technology for Microsoft SQL Server 2005 and 2008 is being released.
Gensym-17415, Gensym-45535, Gensym-30331	Saving operation on a kb fails in 84R1 when a backup kb is present at the saving location. This issue is fixed.
Gensym-45539	G2 aborts when specific application that runs on 83R1 is loaded into 84R1. This issue is fixed.
Gensym-107	G2 aborts when tabular-function-of-1-arg is called using a sequence or structure. This issue is fixed
Gensym-98	The port number is changed to 20000 on clicking user menu choice connect in G2 database interface. This issue is fixed
Gensym-34713	TWNG ignores "rolling the mouse wheel {forward / backward} and does nothing" when displaying html-view.
Gensym-99, Gensym-18859	Errors in UIL on loading the application due to improper error handling. This issue is fixed.
HQ-5747651 / Gensymh-5357	It was not possible to launch a native editor as modal. This issue is fixed.
HQ-5346283 / Gensymh-1398	G2 becomes sluggish when switched to fast as possible mode after real time mode. This issue is fixed.

Gensym-41	ReThink ignores needs-all-inputs attribute setting of blocks. This issue is fixed.
Gensym-40	The bpr-insert-block doesn't insert items into the container attribute of the container object in ReThink. This issue is fixed.
Gensym-45	Field level validation is called twice if we create a native text box using gdu-add-text-box-control() and specify a validation procedure. This problem is fixed.
Gensym-46712	Developer code was left over in sys-mod.kb. This has been resolved.
Gensym-34	Native dialog always appears in primary display even when TWNG is used in dual-display environment. This issue is fixed.
Gensym-85	The changes to GRTL-MODULE-SETTINGS object get overwritten after KB start/restart. This issue is fixed
HQ-5142516	G2 aborts on concluding the value of a nested structure that is an attr of a class. This issue is fixed.
HQ-5749759 / Gensymh-2440, Gensym-12521	Memory is consumed by messages and workspaces created by g2-save-module API. A new API "g2-reclaim-save-module-memory" is introduced to reclaim the consumed memory.
Gensym-196	Installation issues with OPC link in 83R1. This issue is fixed.
Gensym-117	Whenever we edit a method using native editor only the top signature is animated in blue. The issue is fixed.
Gensym-46704	G2 Server aborts on loading cdgmodguide.kb. This issue has been fixed.
HQ-5383513	When an object is moved across the screen it leaves "tracks" or shadows. This issue is fixed.
HQ-5341671	Memory Leaks in G2GL while compiling procedures. This issue is fixed.
HQ-5578136 / Gensym-138	If a workspace view currently has scroll bars, then moving the workspace via the "show workspace" action to a position completely outside the view may shrink the view to zero size, leaving only a title bar. This issue is fixed.
Gensym-77	Memory leaks are seen in Opegrity 5.1 if a csv file containing data points is replayed. This issue is fixed.
Gensym-12523	G2 Server internal object/item relations management has been improved to provide better runtime performance for any KB application using G2 relations.

HQ-5097830 / Gensymh-930	Uninterrupted procedure execution time limit exceeded in g2-write-line. This issue is fixed. A new option is added to "System Tables" -> "Miscellaneous Parameters" named by "Calculate time of called procedures" which will allow switch off the fix.

Features Implemented

Gensym-118	A call back is required that gets called as soon as the selection is made on combobox. This feature is implemented.
Gensym-10797	The requirement is to allow running procedure on the LHS of a rule. The feature is implemented
Gensym-6154	A new API also include a sequence of structures that contains the "starting" and "ending" positions of the affected text supporting g2-get-items-affected-by-attribute-name-change" and "g2-rename-class-or-attribute" is required. This request has been implemented.
Gensym-42	A method to specify the background color of the text box was required. 5 New APIs are added in gdu module for this purpose.
HQ-5736245 / Gensymh-23911	Support for integer 32 bit and 64 bit signed and unsigned values is to be provided. This feature request is implemented.
HQ-4778853	Feature which allows G2 user authentication based on OS standard like LDAP or ActiveDirectory is implemented.
Gensym-64	Ability to configure all the attributes of G2-Database-Interface in the table of GDSM-Database-Connection-Pool. This feature has been implemented.
HQ-5758971 / Gensymh-3386	This feature is to provide a new API to have callback when g2 aborts. The server should run with command line "-use-g2-abort-callback". It tells server that should call g2-abort-callback instead of standard mechanism error notification.

G2 bundle 8.4r1 – change list

Issue ID	Description of change
HQ-5696106	Special characters like "@' were not visible after editing and saving text. These characters are now visible.
HQ-5677168	The Validation procedure used to be called to validate the contents of a text box created by the GDU API but not the focus is changed. A new Custom Dialog Control Behavior named focus-changed-custom-validate-control-value-procedure is added for gdu-add-detail-button-control API
HQ-5737893/ Gensym101,Gensym-102	G2 and Twng used to freeze when adding to the order of 100,000 nodes and deleting them. The problem is fixed by optimizing G2 and adding a faster tree control in twng.
Gensym-54	There was hang observed in the application which uses G2Com. The hang is fixed in the current release.
HQ-5263399 /HQ-5268570	On rotating (90 degrees or -90 degrees) "text in icon" objects upon a workspace and typing Ctrl+W (Twenty percent wider) or Ctrl+N (Twenty percent narrower) several times, the texts in icon were disordered. They are now displayed correctly.
HQ-5662669	The attribute "multi-axes-proportions" did not work in Chart Views. It now works as documented.
HQ-5476610/HQ5459372	New feature: Ability to change the font size of all application menus (View->Menu size)
HQ-5469715/HQ5468600	TWNG "Window" menu used to show only nine opened windows. New feature: Added a new sub-menu "Window ->More Windows" which would hold all other windows menu items but these nine.
HQ-5677976	When File->Save was chosen on a running kb only top level module could be saved. New feature: The "Save" command now allows saving all user changes in all modules on a running kb as well by means of a new check-box in the Save dialog (TWNG only).
HQ-5684960	New feature: Added two new duration controls to the UI controls. New duration controls have the format "hh:mm:ss" - gdu-demo.kb contains usage examples.
HQ-5684961	On creating a method from "Fault Model" toolbox in Symcure gave "inconsistencies detected" message on operator logbook. This no longer occurs.
HQ-5304677	New feature: Half-width Katakana font encoding is now supported.
HQ-5589246/ HQ-5591641	New feature: Added the value-checking expressions from the G2 expression language to the G2GL expression language.
HQ-5721792	The Y axes of multiple charts could not be aligned in the native chart view. Multiple charts now have aligned axes.
HQ-5737001	Under certain circumstances, having a user with a password that never expires might result in an invalid OK file. This problem is now resolved.
HQ-5730937	When g2-ui-launch-editor is used to open a G2 editor and If you start typing a word and click "control-spacebar", you get a pop-up with the entire list of completed keywords even ones that have been excluded. This problem is fixed in this release.
HQ-5613073/HQ5617619	System procedure g2-nms-dismiss() did not dismiss opened menu as it was supposed to do. This has now been fixed.

Versata Confidential.

HQ-5466592/HQ5468546	G2 displayed the "tile-vertically" menu for TWNG incorrectly in non-English locales. This has now been fixed.
HQ-5714215	Changing the gsi-interface-name of this gsi-data-service from one valid interface-name to another valid interface-name (or even if we just reconclude it) caused the gsi-data-service to be is dropped from the item-list. This no longer occurs.
HQ-5696305	NOL - Twng might emit an error while making data-set permanent. This error has been fixed.
HQ-5726693	When the workspace-scaling of the printing-detail of the system-table printer-setup is set to something else then 100, then the output from g2-ui-printworkspace() to a jpeg file was garbled. All scales are now correctly supported in the output.
HQ-5722050/HQ5638016	G2 Rename file did longer allow drive changes. A new API invocation now supports moving files across drives.
HQ-5747814	New feature: The support for word wrapping is added for Native grid and tabular view.
HQ-5639667	Too many messages to the queues in GEVM causes G2 to slow down almost to a halt. GEVM is now much more performant.
HQ-5685830	GEVM messages could cause memory leak while inserting messages. This has now been fixed.
HQ-5685832	A G2 abort could be caused by too many gevm messages. This has now been fixed.
HQ-5737181	Memory leak could occur when using hash tables. These leaks have been stopped.
HQ-5717416	Memory leaks could occur in the TWNG under certain circumstances. These leaks have been stopped.
HQ-5389688	Checkmarks on GMS menus did not work in TW and TWNG. They did not show up in TW clients, but they still worked in the G2 server. They now appear in TW & TWNG as well.
HQ-5615265	A GSI memory leak could occur while using gsi.dll – this has now been resolved.
HQ-5700108	New feature: A new ODBC bridge supporting UNICODE characters is added in this release. This new bridge will replace the existing ODBC bridge starting with G2 bundle 8.4r2
HQ-5647093	Symcure events could become inferred true after specified true. This has now been fixed.
HQ-5722053	New feature: Independence of effects support with output fraction 1.0 on NM-NM events
HQ-5685019/HQ5621951	New feature: Unicode support for SymCure CDG import-export feature
HQ-5732321	TWNG aborted when pie chart windows was continuously created and destroyed in charts.kb. This problem has been fixed.
Gensym-209	New feature: A new Oracle 11 DB bridge is now included in the G2 bundle
HQ-5727990	When opening two or more connections to a java socket, disconnecting one will stop data flow in all connections. This has now been fixed.
HQ-5696305	NOL: TWNG could emit an error while making data-set permanent when count increases more then 100. This has now been fixed.
HQ-5699700	Anytime TWNG is showing a G2 workspace, the number of GDI processes grows. This leak has been stopped.
HQ-5732318/HQ5737358	Menubar reappeared even though it was hidden. This has been fixed.
HQ-568807	Objects created could be lost after saving and reloading the kb in Optegrity. This has been fixed.

Versata Confidential.

HQ-5726443	New feature: The support for printing native dialog is added in this release. Support for certain controls will be added in G2 bundle 8.4r2
Gensym-71	If Two dialogs are displayed one after the other in TW, closing the second one before the first could cause an error. This has now been fixed.
HQ-5513188/HQ5518534	After hiding the workspace of a trend-chart, data-window-time-span of which were several weeks or more and redisplaying it, the early part of plots could be lost. This has now been fixed.
HQ-5712791	g2-ui-get-selected-window-handle procedure signalled an error when local window was selected. This error has been replaced by the message "This operation cannot be performed on a g2 local window"
HQ-5314826/HQ5323849	New feature: Scenario manager now allows working with scripts of up to 2MB in size.
HQ-5323850 / HQ-5736251	New feature: Limit for text values extended to 2MB
HQ-5753994	Repeatedly calling G2-send-soap-request could cause a segmentation fault. This has now been fixed.
Gensym-49	New feature: SUN JRE 6 is now supported

Issues Addressed in G28.4R0

Issues Fixed: Below are the Issues Ids / Feature Requests that are fixed for this release.

Issue no	Issue title
HQ-5630818	Telewindows did not start from the command line if the command line length is too large. This issue is fixed.
HQ-5644947	The Order Selection 1-2 and 3-4 for D1 and D2 do not work even if sort direction and sort criteria are set. Build Selection 1-2 and 3-4 for M1, M2, M3 and Mb do not work too. These blocks always select orders in LIFO regardless of actual settings of the Order/Build Selection. This issue is fixed
HQ-5636201 HQ-5636382	State transition of NM-NM event with option is not consistent. Events were changing to Inferred true/false during propagation even after specifying the value of event. This Issue is fixed.
HQ-5695007	When using a dialogs with Icon button in TWNG, the CHM help file is appearing at the back the TWNG. This should appear as the first item. This Issue is fixed.
HQ-5632446	Delay has been observed while connecting TWNG, due to the native log-book. This issue is fixed.
HQ-5681538	User-defined procedures and methods can't be used. This bug is specific to Optegrity product. This issue is fixed
HQ-5677006	If item configuration (not the main menu restrictions as explained in the bug HQ-5573456) is changed and saves, then the kb is reloaded; The changes were not included in the application. This issue is fixed
HQ-5476196	Inconsistencies in the protols call tree and inspect call tree function where some of the called procedures were missing from the call tree.This issue has been fixed
HQ-5685158	"Independent Of Effects" option for "AND" and "N/M" events, Where setting this value from the event properties make the event independent of its effects.
HQ-5713451	Sometimes NM-NM event with 'Independent Of Effects' option on is upstream inferred false when its effects are set false. (SymcureUserGuide.pdf, Page No – 215)
HQ-5673853	A suggestion for adding the SUCI-Add menu-choice on table of item "go-to-item or show-item". The menu choice has been added.
HQ-5696803	In SymCure,a global variable is needed that returns true if the cgd engine is online or returns false if the cdg engine is offline. This suggestion is included.
HQ-5685961	This is to add topological sorting feature for the generic fault models into the SymCure Application. Currently topological sorting exists only for the specific fault models. This issue is fixed and the topological sortign is added for the Generic Fault

	Models.
HQ-5685871	NM-NM event sensitivity has been made more flexible. Now we can specify delay required in propagating the value.
HQ-5747639	G2 was aborting on application reset. G2 is not aborting now. This issue is fixed
HQ-5748740	Protools could not find unused variable if appropriate call is commented. This issue has been fixed
HQ-3606250	Intriguingly evil abort of G2. The G2 system bug has been fixed.
HQ-3599784	Instance blows stack aborts on save / Class-definition, which inherits from own instance, blows stack aborts on save. G2 is not aborting now. This issue is fixed.
HQ-3449469	G2 abort after loading snapshot. The abort is fixed
HQ-5467919	The goal of this project is investigate and fix the specified bug in the GRTL knowledge base (KB) file. It concerns connecting to the G2 Server when KB is not fully loaded yet. This connection makes the loaded application main menus (File, Edit) empty. This bug is fixed
HQ-5620430	Inconsistencies in GDA application menu, where "appliacation" was not shown in Twng.This issue has been fixed
HQ-5537477	There are several instances of the Error Queue come up on application startup, apparently one per error message logged. Each view shows the same list of error messages. It is also true for custom queues. This issue is fixed.
HQ-5589450	Restricting access to a root menu template based on user mode was not working. This issue is fixed.
HQ-5617174	API "send-fault-model-event" has been published.
HQ-5190424	Java link used to work only with Java (JDK) 1.4. No Javalink is enhanced to run with Java (JDK) 1.5
HQ-5674916	Alt Keys were failing with Caps Lock on. This issue is fixed

Issues Addressed in G28.3R3

Issues fixed: Below are the Issues Ids / Feature Requests that are fixed for this patch.

Issue no	Issue title
Hq-5608585	Making a matrix permanent could result in an error. This has now been fixed.
Hq-5505895	A new control action sort-rows has been added to re-sort rows of tabular views.
Hq-5632610	New entries would always be added to the end of GEVM queues. An attribute sort-views-upon-update has now been added to make this behavior configurable.
Hq-5682094	Under certain circumstances, SymCure event propagation might incorrectly mark events suspect. This has now been fixed.
Hq-5499854	The message queue popup menu and toolbar might not be updated on user-mode change. This has now been fixed.
Hq-5682092	Under certain circumstances, SymCure events might spuriously change from inferred to specified. This has now been fixed.
Hq-5615265	Using GSI via .Net interop might result in a memory leak. This has now been fixed.
Hq-5621386	When a tree node is deleted, its children might survive. This has now been fixed.
Hq-5621389	Under certain circumstances, the color of tree-view icons may not be updated. This has now been fixed.
Hq-5622064 (Bug HQ-5622513)	Under certain circumstances, string comparison inside a G2GL switch-fork condition could be case-sensitive. This has now been fixed.
Hq-5621004	Under certain circumstances, G2 might generate unreadable PostScript files. This has now been fixed.
Hq-5632008 (Bug HQ-5641399)	G2 could leak memory when applying the function symbol. This has now been fixed.
Hq-5686220	The system limit of 130,000 native icons might be spuriously exceeded. This has now been fixed.
Hq-5699700	The number of GDI processes might unduly grow in certain circumstances. This has now been fixed.
Hq-5570524 (Bug HQ-5575594)	Under certain circumstances, a tree view control might degrade in performance monotonically. This has now been fixed.
Hq-5647736	A g2-window object might incorrectly persist after unexpected termination of TWNG. This has now been fixed.
Hq-5647541	TWNG might crash after several million continuous property changes to a tree view. This has now been fixed.
Hq-5656948	TWNG might leak memory when showing and hiding workspaces. This has now been fixed.
Hq-5577406	TWNG might leak tooltips when associated tree view nodes are deleted. This has now been fixed.
Hq-5656952	Under certain circumstances, combo box callbacks might be incorrectly invoked. This has now been fixed.
Hq-5698037	Creation and deletion of combo boxes might leak memory. This has now been fixed.
Hq-5698060	The system limit of 130,000 native icons might be spuriously exceeded. This has now been fixed.
Hq-5702427	Under certain circumstances, icons might be updated with incorrect

	images. This has now been fixed.
Hq-5629528	The call g2-ui-print-workspace did not respect the JPEG quality parameter. This has now been fixed.
Hq-5637522	The CDG module might intermittently display a spurious message. This has now been fixed.
Hq-5685963	The CDG module might intermittently display a spurious message. This has now been fixed.
Hq-5683231	Certain toolbar operations might leak memory. This has now been fixed.
Hq-5595405	It was not possible to determine programmatically whether a class can be instantiated. This can now be done by reading the attribute instantiate.
Hq-5679116	G2, when invoked in classic UI mode, did not support certain UI configuration options. This has now been fixed.
Hq-5259563	Under certain circumstances, deleting all workspaces of a module can cause G2 to abort. This has now been fixed.
Hq-5641329	Certain control keys may not be recognized with the caps lock on. This has now been fixed
Hq-5483955	Under certain circumstances, Protools may not find a called method. This has now been fixed.
Hq-5077052	Under certain circumstances, a Rethink report may be updated only when deleted. This has now been fixed.
Hq-3177181	Under certain circumstances, a Rethink object cannot be loaded from file. This has now been fixed.
Hq-5593312	Under certain circumstances, viewing a disabled class definition might cause an error. This has now been fixed.
Hq-5519377	The HLA bridge did not display its version. This has now been fixed.
HQ-5392920	Protools: Selecting the launch procedure menu choice of a procedure, does not do type checking to be performed for arguments of type truth-value. The result defaults to False. This is fixed now.
HQ-5570526	Gnne version incorrect in g2 83r1 bundle. GNNE version is showing as 51r0, but NOL and NOL Studio are 51r1. This is fixed now.
HQ-5627280	The environment variable G2_SERVER was not mentioned in the grpe manual. The G2 Reporting Engine user's guide has now been updated – Chapter 5 (page 57) – with this information.
HQ-5609289	Terminating the g2-opc client window generates an "end now" button. This is fixed now.
HQ-5604040	Hla bridge connection broken. The G2-HLA Bridge user's guide has been updated – Chapter 1 (page 2) – with setup instructions.
HQ-3033037	TW aborts because of running out of pixmaps when it is repeatedly rerouted. This is fixed now.
HQ-5362738	In the data-server-aliases of data-server-parameters, adding "none" to the list of aliases causes g2 to abort. This is fixed.
HQ-5345421	Trend-chart aborts g2 when lower-bound > upper-bound. This is fixed.
HQ-4655290	Certain operations involving selection UI and inspect might cause an abort. This is fixed.
HQ-4548609	G2 or TW displaying on a Windows platform may abort when the available memory for graphical objects is exhausted. This is Fixed now.
HQ-3477026	Abort with bad value for identifying attributes of interface. This is Fixed now.

HQ-3322650	When we put the image-definition on the sub workspace of the media-bin, g2 aborts. This is fixed now.
HQ-2880891	Rule compilation might abort g2. This is fixed now.
HQ-5332405	Rethink: system attribute named class does not display in block input. This is fixed now.
HQ-5517326	Accelerator labels are not displayed in native menu of gms-choice-template. This is Fixed now.
HQ-1050868	If the Maximum Activities of a block are specified and work objects arrive at the block simultaneously, ReThink ignores the specification of Maximum Activities. This is fixed now.
HQ-5601884	Gedp-quotient-block might not be connected to an upstream block. This is fixed.
HQ-5282778	G2 abort when using trend-chart. This is fixed.
HQ-5566882	Bug: protools call-tree doesn't recognize // as a comment. This is fixed and now it recognizes // as comment
HQ-5567729	Integrity: no indication of failure creating domain object from ping manager dialog. This is fixed.
HQ-5566571	Inconsistencies when loading the NOL module. This is fixed now.
HQ-5620431	Incomplete items when loading gda-tank demo. This is fixed now.

G2 Version 8.3 Rev. 1

Describes the new features and changes in G2 Version 8.3 Rev. 1.

Introduction

This chapter describes the new features and changes in G2 Version 8.3 Rev. 1, which ships with the G2 Bundle Version 8.3 Rev. 1.

Information on bug fixes and known bugs is available separately as Technical Bulletins through HelpLink at `www.gensym-support.com`. For information on bugs in G2 Version 8.3 Rev. 1, choose Technical Bulletins for G2, specify Starting From Version as 8.3 Rev. 0 and Ending At Version as 8.3 Rev. 1, choose the bugs and suggestions that either Remain Open or Were Fixed, then click Execute.

G2 Graphical Language (G2GL)

When a G2GL process calls a G2 procedure or invokes a remote Web service, and that procedure signals an error, a G2GL fault is signaled with the error description and backtrace as the fault description. Previously, the error was posted to the operator logbook and the G2GL process continued without signaling a fault.

Custom Dialogs

Grid-View

The grid view control now supports multi-line input text, using the carriage return and line feed characters. Previously, using these characters did not place text on multiple lines.

Removing the Icon on MDI Child Windows

You can now specify a blank icon on the title bar of an MDI child view by using this syntax:

g2-ui-modify-view (*handle*, structure(icon: the symbol NONE), *win*)

where *handle* is the handle of the MDI child window.

Previously, specifying an icon of none did not remove the icon.

G2 Bundle

Windows Vista

G2 is now supported on the Windows Vista operating system.

HP Itanium

As of G2 Version 8.3 Rev. 1, G2, Telewindows, G2 Gateway, and G2 JavaLink are all supported on the HP Itanium platform under HP-UX 11.23. Other bridges and layered products have been compiled and linked on a PA-RISC machine running HP-UX 11.00 and run in compatibility mode on the HP Itanium machine under HP-UX 11.23. Thus, the `gsi-itanium` directory no longer exists.

Linux

On Linux, the G2 Bundle ships with an additional G2 Gateway directory called `gsi-redhat4`, which contains the GSI libraries and examples compiled with Redhat Enterprise Linux 4 with kernel 2.6.9-55.EL. Otherwise, in G2 Version 8.3 Rev. 1, all other G2 components are compiled with Linux 7.3 kernel 2.4.26.

In G2 Version 8.4 Rev. 0, all G2 components will be compiled with Redhat Enterprise Linux 4 with kernel 2.6.9-55.EL, which will become the minimum supported version.

G2 Version 8.3 Rev. 0

Describes the new features and changes in G2 Version 8.3 Rev. 0.

gensym

Introduction

This chapter describes the new features and changes in G2 Version 8.3 Rev. 0, which ships with the G2 Bundle Version 8.3 Rev. 0.

Information on bug fixes and known bugs is available separately as Technical Bulletins through HelpLink at `www.gensym-support.com`. For information on bugs in G2 Version 8.3 Rev. 0, choose Technical Bulletins for G2, specify Starting From Version as 8.2 Rev. 4 and Ending At Version as 8.3 Rev. 0, choose the bugs and suggestions that either Remain Open or Were Fixed, then click Execute.

Telewindows Next Generation Only

This section lists the features of G2 that are supported in Telewindows Next Generation (twng.exe) only and describes other differences.

These features are only supported in Telewindows Next Generation:

- Windows views:
  - Chart views
  - HTML views
  - Tree view
  - Shortcut bar
  - Status bar
  - Property grid
  - Tabbed MDI mode
  - Native logbook
  - Native message board
- Custom Windows dialogs
- Toolbars

G2 Graphical Language (G2GL)

New G2GL Icons

G2GL provides new icons and connections for all the G2GL activities, as well as icons for G2GL processes. Here are the new icons:
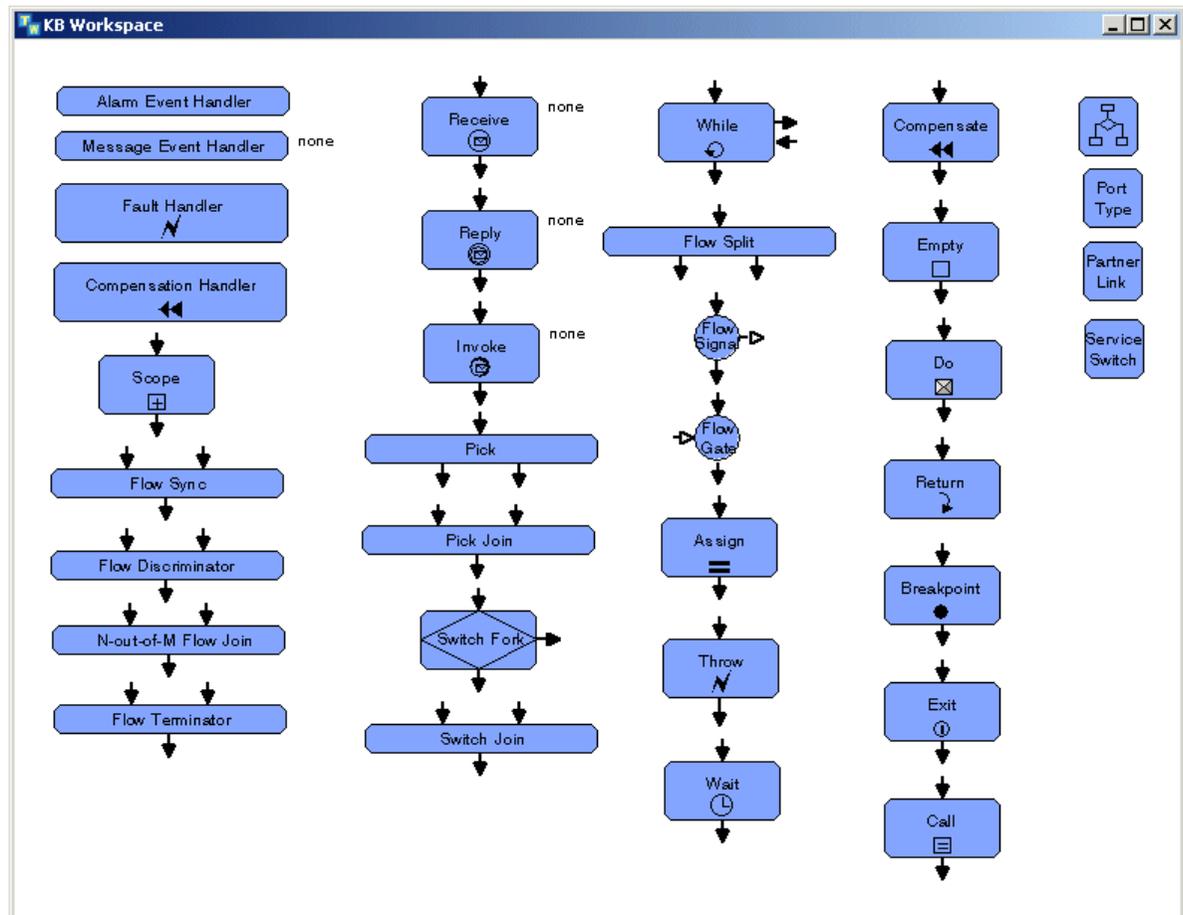
Compensation

The compensation mechanism in G2GL supports transaction-based computing. A Compensation Handler on a Scope body specifies how to compensate for (for example, undo) the work done by that Scope in case a fault is signaled in a parent Scope after the Scope has completed successfully. A Scope completes successfully if it finishes executing without signaling a fault. Even if a signaled fault is handled

by a Fault Handler in that Scope, the Scope is not considered to have completed successfully.

When a Compensation Handler is invoked, the activities on its body are executed. All variables in the handler's Scope and its ancestor Scopes are temporarily restored to their values at the time when the Scope completed successfully.

A Compensation Handler can be invoked in one of two ways:

- Explicitly by a Compensate activity in a Fault Handler or Compensation Handler in the parent Scope.

- Implicitly if there is no applicable Fault Handler or no Compensation Handler in the parent Scope.

Explicit Compensation

By default, a Compensate activity in a Fault Handler or Compensation Handler body invokes the Compensation Handlers for all Scopes that completed successfully in the same body as the handler, in reverse order of their completion. Alternatively, you can configure the scope-name-for-compensate-activity to be the name of a specific Scope in the same body as the handler; the Compensate activity will then only invoke the Compensation Handler on the body of that Scope.

If the named Scope did not complete successfully or if its Compensation Handler was already invoked, then its Compensation Handler is not invoked, and the Compensate activity does nothing.

If the named Scope completed successfully multiple times because it was inside a While loop, its Compensation Handler is invoked once for each completion, in reverse order.

Implicit Compensation

If there is no applicable Fault Handler in a Scope when a fault is signaled, all Compensation Handlers in Scopes that have completed successfully in that Scope are invoked, in reverse order of their completion, before the fault is propagated to the parent Scope. If there is no Compensation Handler in a Scope that is being compensated, all Compensation Handlers in Scopes that have completed successfully in that Scope are invoked, in reverse order of completion. If a Fault Handler or Compensation Handler has no Compensate activities, then the Scopes on the same body as the handler will not be compensated. In other words, implicit compensation only happens if there is no applicable Fault Handler when a fault is signaled or no Compensation Handler when a Scope is compensated.

Local Compensation Handler on an Invoke Activity

An Invoke activity may be connected to a Compensation Handler with a right-side output connection. This is equivalent to the Invoke activity being inside its own Scope with the Compensation Handler being on that Scope body.

Invoking Remote Web Service Operations

Partner link variables now have a default-value-for-g2gl-variable attribute, similar to local and argument variables. Its value can either be "local" (the default) to represent a link to another G2GL process in the same KB, or an endpoint reference specification, to represent a link to a remote Web service, with the following syntax:

> service-namespace: *service-namespace*, service-name: *service-name*,
> endpoint-name: *endpoint-name*

For more information about endpoint references, see "Web Services" on page 139.

An Invoke activity that uses a partner link variable containing an endpoint reference invokes an operation on the remote Web service specified by the endpoint reference. The `g2\kbs\utils\g2web.kb` module must be loaded to enable G2GL to communicate with remote Web services.

Note that the Invoke activity waits for the operation to complete, even in the case of one-way communication where there is no reply; for example, invoking an HTTP operation waits for the HTTP response before continuing, because HTTP is a synchronous protocol.

For an example, see the Shakespeare demo in `g2web-demo.kb` located in the `g2\kbs\demos` directory.

Invoking Instantiation Trigger Operations Programmatically

You can use the following system procedure to invoke an instantiation trigger operation in a G2GL process programmatically:

g2-invoke-g2gl-operation
  (*service-switch*: class g2gl-service-switch, *operation-name*: symbol,
  input-message: item-or-value)
  –> *output-message*: item-or-value, *reply-or-fault-name*: symbol

In order for a G2GL process to be available for instantiation this way, the name of the *service-switch* item must be in its names-of-g2gl-service-switches-for-instantiation list, or, if that is "none", the name of the *service-switch* item must be its name-of-g2gl-service-switch-for-connection. By default, every KB has a g2gl-service-switch item named g2gl-standard-service-switch, which is also the default name-of-g2gl-service-switch-for-connection for G2GL processes.

The *operation-name* is the name of the instantiation trigger operation. The *input-message* and *output-message* are Web service message structures, or, for backwards compatibility, instances of the class g2gl-message or a subclass. For a description of Web service message structures, see "G2GL Messages as Web Service Message Structures" on page 50.

The *reply-or-fault-name* is either the symbol reply if the reply was not a fault, or else the name of the fault.

Note   Currently, this system procedure can only invoke two-way synchronous operations. The system procedure always waits for a reply from the G2GL process. If the process exits without responding, a g2gl-fault error is signaled, whose fault-name is partner-has-terminated.

G2GL Messages as Web Service Message Structures

A G2GL message can be a Web service message structure, or, for backward compatibility, an instance of a subclass of g2gl-message. A Web service message structure is a structure whose attributes correspond to message parts. The value of a message part attribute is a text, an XML element value, or a sequence of texts and/or XML element values. For more information, see "Web Services" on page 139.

In addition, the g2gl-assignments attribute of the Assign activity handles expressions of the form "the *attribute* of *variable* = *value*", where the variable is either uninitialized or holds a structure. In either case, the variable is updated to hold a structure with the new attribute value included.

G2GL Expressions

Time Expressions

Previously, the expression the current time returned the current UNIX time, as a float, which is the number of seconds since January 1, 1970, which was inconsistent with the return value of this expression in G2, which returned the number of seconds, as an integer, since the KB was started. To correct this inconsistency, G2GL has two changes:

- G2GL supports the following G2 time expressions, which return the same values as in G2:

  – the current subsecond [real] time — Returns the number of seconds since the KB was started, as a float, using the G2 clock (the current subsecond time), or using the real-time clock (the current subsecond real time).

- the current {time | real time | year | month | day of the month | day of the week | hour | minute | second} — Returns an integer that represents the specified time increment.

- G2GL (and G2) supports the following new expressions:

  - the current system time — Returns the current UNIX time, as a float, which is the number of seconds since January 1, 1970, using the G2 clock.

  - the current system real time — Returns the current UNIX time, as a float, which is the number of seconds since January 1, 1970, using the real-time clock.

Note that the G2GL expressions used to specify deadlines, such as in the Wait activity, must use the new time expressions. For example, to wait 2 seconds with the type-of-g2gl-alarm-expression attribute being deadline-expression, the duration-or-deadline-expression attribute should be the current system time + 2 seconds, not the current time + 2 seconds as was previously the case.

Concatenation Operator

You can use square brackets in text expressions to insert the text value of a G2GL expression into a literal text. For example, in the g2gl-assignments attribute of the Assign activity, you could specify:

    local-text = "Random = [random(100)]"

Sequence Expressions

You can now access elements of a sequence by using expressions such as:

    my-sequence[0]

    my-sequence[1][2]

Previously, these expressions generated an error.

g2-call-g2gl-process-as-procedure Return Values

The g2-call-g2gl-process-as-procedure system procedure now returns a single value, the sequence of values returned by the g2gl-process, or signals an error if a fault is not handled at the top-level of the g2gl-process. The error is an instance of the new system error class g2gl-fault, which is a subclass of g2-error. The error-description has a text describing the fault, which is the same as the message shown on the breakpoint execution display. The error item has two additional attributes, fault-name, a symbol, and fault-data, which is "none" for all non-user faults and for user faults that don't include fault data.

Assigning Sequences and Structures to G2 Item Attributes

The G2GL Assign activity allows you to assign a sequence or structure value to a G2 item attribute. Previously, attempting to assign a sequence or structure to a G2 item attribute generated an error.

Calling Uncompiled G2GL Processes

The g2-call-g2gl-process-as-procedure system procedure first compiles the specified process if it has not already been successfully compiled, then calls it. Previously, calling an uncompiled G2GL process using this procedure generated an error.

Compilation Postings

When the same error or warning occurs more than once in a G2GL process, for example, when referring to an undeclared variable, G2GL creates compilation errors and warnings for each occurrence of the error or warning. Previously, when the same error or warning occurs, it created a single compilation posting.

g2gl-credit-rating-example.kb

The g2gl-credit-rating-example.kb demo has been updated as follows:

- The demo uses Web service message structures as described in "G2GL Messages as Web Service Message Structures" on page 50.

- The Invoke and Reply activities use an uninitialized variable as its message variable, which is equivalent to sending a message that has no parts.

- The Asynchronous Communication and Synchronous Communication examples assign subclasses of g2gl-message.

- The icons and connections have been updated as described in "New G2GL Icons" on page 47.

Custom Dialogs

Custom Windows Dialogs Only Supported in Telewindows Next Generation

As of G2 Version 8.3 Rev. 0, custom Windows dialogs are only supported in Telewindows Next Generation Version 8.3 Rev. 0 (twng.exe). Previously, certain Windows dialog features were supported in Telewindows (tw.exe), while other features were only supported in Telewindows Next Generation.

Note that if you are running an earlier version of G2, such as G2 Version 8.2, and you attempt to display a custom dialog in Telewindows Version 8.3 (tw.exe) that contains only control types that were supported in earlier versions of Telewindows, the dialog simply does not display and no error is reported. Therefore, if you have an application that uses custom dialogs running in an earlier version of G2, you must use either Telewindows Next Generation Version 8.3 or a pre-8.3 version of Telewindows (tw.exe) in order to see the dialogs.

Custom Dialogs Use Windows Display Properties Fonts

Custom dialogs obey the Windows desktop font preferences, which you specify by choosing Start > Settings > Control Panel > Display and clicking the Appearance tab. In particular, dialogs use the Message Box font, which you specify by clicking the Advanced button.

Calendar

The calendar control supports modifying the control via g2-ui-modify-custom-dialog. Although this feature was documented, it was not previously implemented. The calendar control supports these control actions: replace, enable, disable, hide, and show.

Color-Picker

The color-picker control supports the replace control action via g2-ui-modify-custom-dialog.

Combo-box

A combo-box control whose list-box-style is simple or dropdown allows you to select text in the current selection, using the text-selection attribute in the control-value, as follows:

Attribute Name	Type	Required	Default	Description
control-value	structure	yes	N/A	A structure that specifies a sequence of text values in the combo box, the text of the initially selected value, the selected text, and the width of the combo-box:

structure
(text-sequence:
  sequence (*text*[, ...]) | ( ),
selected: *text*,
text-selection: *index* |
   sequence (*index*, *index*),
dropdown-width: *integer*)

In the control-value, specify text-selection as an integer to set the initial cursor position at the specified index or as a sequence to set the initial selection between the specified indices, where *index* is a zero-based integer index.

The selection is returned as the text-selection attribute of the control-value, as either an integer or a sequence.

Duration

The duration control provides a tooltip when hovering the mouse over each field to indicate the units: weeks, days, hours, minutes, and seconds.

The duration control supports modifying the control via the g2-ui-modify-custom-dialog system procedure.

Grid-View

Specifying Dropdown List Style for Combo-Box Cells

Grid-view combo-box cells support a new attribute in their *cell-value-specification*, list-box-style, which is a symbol with values:

- dropdown — Allows entering any value via an edit box, the default.

- dropdownlist — Allows only selections from the dropdown list.

Specifying Bold Font for Cells in a Grid-View

The grid-view control *cell-setting-spec* defines a new attribute, bold, which is a truth-value. When true, the cell's text is bold. You can also specify the attribute in the *column-spec* to specify that all text in a column should be bold.

Color-Picker in Grid-View

The color-picker in a grid-view now looks like the color-picker control that appears in a dialog with a color swatch.

Displaying Animated Icons

The grid-view control now supports displaying icons based on G2 items whose icons update dynamically as the G2 icon changes.

In addition, the scaling and clipping behavior of icons in grid views has changed to be more intuitive, as follows:

- If an icon is too large for its cell, the portion that fits is now drawn, rather than drawing nothing at all.

- If an icon is smaller than 32x32 pixels, then it is no longer scaled up to that size, rather it is drawn at its actual size.

Getting Updates when the Selection Changes

The dialog-generic-callback for a custom dialog, if specified, is called whenever the set of selected cells changes in a grid-view on the dialog, with the following arguments:

- *event*: SELECTION-CHANGED

- *control-id*: integer

- *info*: structure
    (row: *integer*,
     column: *integer*,
     selection: *sequence*)

  where:

  – row and column describe a particular cell in the selection, either the cell with the focus, or if no cell has the focus, the top-left-most cell in the selection. The value for row and column are -1 if the selection is empty.

– selection is a sequence of structures representing the rectangular regions of contiguously selected cells in the unsorted grid, where each structure has this syntax:

structure
(minrow: *integer*,
maxrow: *integer*,
mincol: *integer*,
maxcol: *integer*)

- *user-data*: value

Getting Updates When the User Clicks the Mouse or Presses a Key

The dialog-generic-callback for a custom dialog, if specified, is called whenever the user clicks the mouse or presses a key in the grid-view on the dialog, with the following arguments:

- *event*: LEFT-CLICK, MIDDLE-CLICK, RIGHT-CLICK, KEY-PRESS, or any combination of modifier keys or DOUBLE combined with LEFT-CLICK, MIDDLE-CLICK, or RIGHT-CLICK, for example: SHIFT+LEFT-CLICK, CONTROL+RIGHT-CLICK, ALT+LEFT-CLICK, DOUBLE+CONTROL+LEFT-CLICK.

- *control-id*: integer

- *info*: structure
(x: *integer*,
y: *integer*,
selected-cells: *sequence*,
key: *symbol*,
row: *integer*,
column: *integer*)

where:

– x and y are the x-y coordinates of the mouse click in the grid view.

– selected-cells is a sequence of structures representing the row and column of each selected cell, where each structure has this syntax:

structure
(row: *integer*,
column: *integer*)

– key is the key that was pressed, if any.

– row and column describe a particular cell in the selection, either the cell with the focus, or if no cell has the focus, the top-left-most cell in the selection. The value for row and column are -1 if the selection is empty.

- *user-data*: value

Modifying Grid-View Dialog Cells

The documentation for g2-ui-grid-view-replace-cell states that the *cell-spec* structure allows you to replace the text-value of a text-box control only. You can now replace all grid view attributes of all control types, using the following syntax for *cell-spec*:

    structure
    (cell-type: the symbol *cell-type*,
    cell-value: structure ([*attribute*: *value*] ...))

For example, to replace a cell with a text-box cell with a given text and background color, the *cell-spec* would look like this:

    structure
    (cell-type: the symbol text-box,
    cell-value:
        structure (text-value: "New Text", background-color: the symbol blue))

For a description of this system procedure and the attributes of each cell type that you can modify, see the description of grid-view in Chapter 47 "Windows Dialogs" in the *G2 Reference Manual*.

Note that the g2-ui-grid-view-replace-cell system procedure replaces the entire cell contents with the specified *cell-spec*, which requires that you specify the entire cell contents to replace.

You can use the following new system procedure to modify only the specified attributes of a cell, leaving the unspecified attributes unchanged:

g2-ui-grid-view-modify-cell
    (*dialog*: integer, *control*: value, *row*: integer,
    column: integer, *cell-spec*: structure, *window*: class g2-window)

Inserting and Deleting Columns

g2-ui-grid-view-insert-column
    (*dialog*: integer, *control*: value, *column*: integer, *column-spec*: structure,
    win: class g2-window)

Inserts a column into the grid-view specified by *control*, which is the control-id of the control, in *dialog*. The *column* is the *column-id* of the new column, after it has been inserted. The *column* argument must not be greater than the number of columns in the grid after the new column has been added. For example, if the grid has columns 0 - 4 (5 columns), the *column* argument can be 0 - 5, inclusive. A column number of -1 means to insert as the last column.

For a description of *column-spec*, see "grid-view" in Chapter 43, "Custom Windows Dialogs" in the *G2 Reference Manual*. The *column-spec* must specify the same number of rows as the existing columns in the grid.

g2-ui-grid-view-delete-column
    (*dialog*: integer, *control*: value, *column*: integer, *win*: class g2-window)

Deletes a column from the grid-view specified by *control*, which is the control-id of the control, in *dialog*. The *column* is the *column-id* of the column to delete, which must be a valid column in the grid. A column number of -1 means to delete the last column.

Specifying Day/Time Format for Time-of-Day and Calendar Controls

You can add the attribute calendar-format to the *cell-value* of a calendar control in a grid-view, which should be one of the symbols MM-DD-YYYY (the default), DD-MM-YYYY, and YYYY-MM-DD.

Creating a Grid-View with a Calendar Control

If you create a grid-view with a calendar control without specifying the default date, G2 uses "the current date" to create the control. Note that if G2 and Telewindows are in different time zones, and if the dialog is created at a time of day when the date is different in the two time zones, the control uses the date in G2's time zone. Previously, the control used the date in Telewindows' time zone.

Sorting Numerical Grid View Columns

In a grid-view, clicking the column header of a column that contains exclusively cells of type integer or quantity sorts the rows numerically. Previously, clicking the column header only sorted rows alphabetically.

Image

The image control now supports the replace control action via g2-ui-modify-custom-dialog. Although this feature was documented, it was not previously implemented. The image control supports these control actions: add, replace, enable, disable, hide, and show.

Masked-Edit

The masked-edit control now supports modifying the control via g2-ui-modify-custom-dialog. Although this feature was documented, it was not previously implemented. The masked-edit control supports these control actions: add, replace, enable, disable, hide, and show.

The masked-edit control allows you to specify the text color, background color, and initially selected text in the control-value specification, as follows:

Attribute Name	Type	Required	Default	Description
control-value	structure	no	structure ( )	A structure that specifies the current value, mask, literal text for the masked edit box, text color, background color, and initial selection for the masked edit box: structure (current-value: *text*, mask: *text*, literal: *text*, text-color: *color*, background-color: *color*, selection: *index* | sequence (*index, index*))

Push-Button

For consistency, the control-value accepts the icon attribute in addition to the icon-name attribute. However, note that the icon-name attribute is deprecated. You should use the icon attribute instead.

Spinner

Precision, Default Current-Value, Default High-Value, and Increment

The spinner control allows you to specify floating point numbers, as well as integers. The spinner uses decimal or scientific notation if any of the input values (current-value, low-value high-value, or increment) are in decimal or scientific notation; otherwise, it uses integers.

You control the number of digits to the right and left of the decimal point in the control-value by using the precision attribute.

The default value for the high-value and current-value attributes have changed. They used to both be 0.

Here is the specification for the control-value:

| Attribute Name	Type	Required	Default	Description
control-value	structure	yes	structure (current-value: 50, low-value: 0, high-value: 100, increment: 1)	A structure that specifies the default value, low and high values, value for incrementing the spinner, and precision:
				structure (current-value: *quantity*, low-value: *quantity*, high-value: *quantity*, increment: *quantity*, precision: *ddd.dddd-format*)

The precision is specified as a *ddd.dddd-format* expression, which is a symbol that determines the number of decimal places to the right and left of the decimal point. For example, ddd.dd formats a floating point number to the hundredths decimal place.

The minimum value for increment is .0001.

For example:

```
structure (control-type: the symbol spinner,
      control-id: the symbol my-spinner,
      height: 15,
      width: 70,
      left: 100,
      top: 10,
      response-action: the symbol respond,
      control-value: structure (current-value: 95.5,
            low-value: 0.5,
            high-value: 300.5,
            increment: 1.0,
            precision: the symbol dd.ddd))
```
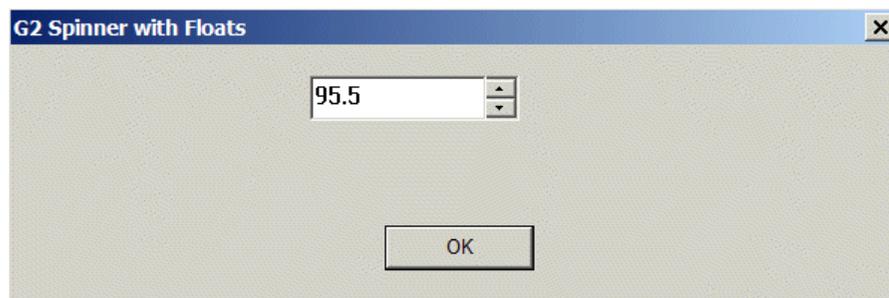
Modifying Spinner Control

The spinner control now supports the replace control action via g2-ui-modify-custom-dialog. Although this feature was documented, it was not previously implemented. The spinner control supports these control actions: add, replace, enable, disable, hide, and show. You can modify all attributes in the control-value structure.
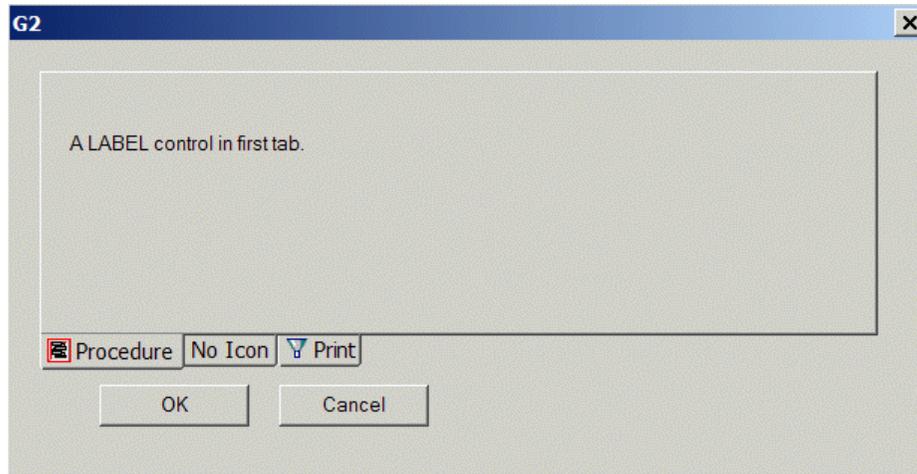
Tab-Frame

Specifying Icons and Tab Position

You can specify icons in place of labels for each tab in a tab-frame control, and you can specify the position of the tabs in the control-value structure, as follows:

Attribute Name	Type	Required	Default	Description
control-value	structure	yes	N/A	A structure that specifies the tab labels, tab icons, initially selected tab, and tab position:
				structure (tab-labels: sequence (*text*,...), tab-icons: sequence (*item-or-symbol*, ...), selected-tab: *text*, tab-position: top | left | right | bottom)

In the control-value structure:

- tab-labels is a sequence of labels for each tab, each of which must be unique. If the sequence is empty, a tab-frame with no tabs is created.

- tab-icons is a sequence of icons, where the number of elements in the sequence must match the number of elements in the tab-labels sequence. To specify no icon for a particular tab, use false. The icon can be a G2 class name, an item, or a built-in GMS icon. For a list of built-in GMS icons, see the description of the image control in Chapter 43, "Custom Windows Dialogs" in the *G2 Reference Manual*.

- selected-tab is the initially selected tab.

- tab-position is the position of the tabs in the overall frame, where the default value is top.

Here is a tab-frame in a dialog with the tabs at the bottom and icons specified for two of the three tabs:

Here is the procedure used to create the dialog:

```
create-tab-frame(side: symbol, Win: class g2-window)
components: sequence = sequence(
  structure (control-type: the symbol push-button,
    control-id: 1, height: 14, width: 50, left: 30, top: 115,
    response-action: the symbol ok,
    control-value: structure (text-value: "OK")),
  structure (control-type: the symbol push-button,
    control-id: 2, height: 14, width: 50, left: 90, top: 115,
    response-action: the symbol cancel,
    control-value: structure (text-value: "Cancel")),
  structure (control-type: the symbol tab-frame,
    control-id: 3, height: 100, width: 280, left: 10, top: 10,
    response-action: the symbol ignore,
    control-value: structure (tab-labels: sequence ("Procedure", "No Icon", "Print"),
                              tab-icons: sequence (this procedure, false, the symbol
                                  GMS-FUNNEL-ICON),
                              tab-position: side)),
  structure (control-type: the symbol label,
    control-id: the symbol L1,  parent-control-id: 3, parent-control-text: "Procedure",
    control-value: structure (text-value: "A LABEL control in first tab."),
    height: 40,  width: 250,  left: 20,  top: 30),
    structure (control-type: the symbol label,
    control-id: the symbol L2,  parent-control-id: 3,   parent-control-text: "No Icon",
    control-value: structure (text-value: "A LABEL control in second tab."),
    height: 40,  width: 250,  left: 20,  top: 30),
  structure (control-type: the symbol label,
    control-id: the symbol L3,  parent-control-id: 3, parent-control-text: "Print",
    control-value: structure (text-value: "A LABEL control in third tab."),
    height: 40,  width: 250,  left: 20,  top: 30));
```

```
begin
    call g2-ui-post-custom-dialog(structure(dialog-width: 310, dialog-height: 160,
components: components), false, Win);
    end
```

Modifying the Selected Tab

You can modify the selected tab of the tab-frame, using the following symbol for control-action in the *modify-specification* in g2-ui-modify-custom-dialog:

Control Action Symbol	Description	Available for Controls	Required Control Value
selected-tab	Changes the selected tab.	tab-frame	A text of the selected tab.

For example:

```
call g2-ui-modify-custom-dialog
        (dialog-id,
        sequence
            (structure
                (control-action: the symbol selected-tab,
                control-id: the symbol my-tab-frame,
                control-value: "Tab2")),
        window);
```

Dialog-Dismissed Callback Return Value

When a custom dialog with a tab-frame is dismissed, the value returned by the dialog-dismissed-callback procedure was formatted incorrectly with an extra level in the structure for the tab-labels attribute, as follows:

```
#5   2:55:15 p.m.   callback values:
  sequence (structure (CONTROL-ID: the
  symbol MY-OK-BUTTON,
    CONTROL-VALUE: structure (TEXT-
      VALUE: "OK")),
  structure (CONTROL-ID: the symbol MY-
    SPINNER,
    CONTROL-VALUE: structure
      (CURRENT-VALUE: 20,
      LOW-VALUE: -50,
      HIGH-VALUE: 50,
      INCREMENT: 5)),
  structure (CONTROL-ID: the symbol
    THIS-TAB-FRAME,
    CONTROL-VALUE: structure (TAB-
      LABELS: structure (TAB-LABELS:
      sequence ("Tab1",
        "Tab2")),
      SELECTED-TAB: "Tab1")))
```

The return value now looks like this:

```
#12   2:52:18 p.m.   callback values:
  sequence (structure (CONTROL-ID: the
  symbol MY-OK-BUTTON,
    CONTROL-VALUE: structure (TEXT-
      VALUE: "OK")),
  structure (CONTROL-ID: the symbol MY-
    SPINNER,
    CONTROL-VALUE: structure
      (CURRENT-VALUE: 20,
      LOW-VALUE: -50.0,
      HIGH-VALUE: 50.0,
      INCREMENT: 5.0)),
  structure (CONTROL-ID: the symbol
    THIS-TAB-FRAME,
    CONTROL-VALUE: structure (TAB-
      LABELS: sequence ("Tab1",
      "Tab2"),
      SELECTED-TAB: "Tab1")))
```

Tabular-View

Removing Columns Dynamically

The g2-ui-modify-dialog system procedure allows the remove-columns control action, applicable only to tabular views, for dynamically removing columns. The control-value is a sequence of integer column numbers, where -1 means the last column.

Symbolic Logical-IDs in Row Structures

The logical-id of the rows structure of the tabular-view control can be a symbol, text, or float, as well as an integer. Also, the logical-id is now optional and defaults to an integer starting at 1.

Sorting

In the specification of a tabular-view, when allow-sort-rows is true, clicking the column header of a column that contains cells of type integer or quantity sorts the rows numerically.

Alignment and Icons

You can specify the alignment of the cell contents of each column in a tabular view.

You can specify icons in cells instead of text in the value specification of a row specification of a tabular view. You can also specify an icon in the column header, which appears to the right of any text in the column header. Note that if a column header has an icon, the sorting arrows are not drawn.

Here is the *column-structure* with the new attributes:

```
structure
(text-value: text,
width: integer-or-symbol,
icon: item-or-symbol,
alignment: left | right | center)
```

Here is the *value-specification* of the *row-specification* with the new attributes:

```
structure
(text-value: text,
icon: item-or-symbol)
```

The icon can be a G2 class name, an item, or a built-in GMS icon. For a list of built-in GMS icons, see the description of the image control in Chapter 43, "Custom Windows Dialogs" in the *G2 Reference Manual*.

Here is an example of a tabular view with different cell alignment:

```
structure (control-type: the symbol tabular-view,
    control-id: the symbol the-tabular-view,
    height: 200,
    width: 210,
    left: 10,
    top: 10,
    text-color: the symbol blue,
    background-color: the symbol yellow,
    allow-sort-rows: true,
    row-height: 25,
    single-selection: true,
    response-action: the symbol ignore,
    control-value: structure (columns: sequence (structure (text-value: "Center Alignment",
        width: 80,
        alignment: the symbol center),
      structure (text-value: "No Alignment",
        width: 50),
      structure (text-value: "Right Alignment",
        width: 50,
        alignment: the symbol right)),
    rows: sequence (structure (logical-id: 0,
      row-values: sequence (structure (text-value: "Value4"),
        structure (text-value: "Value2"),
        structure (text-value: "-1245.90"))),
      structure (logical-id: 1,
        row-values: sequence (structure (text-value: "Override! Right"),
          structure (text-value: "Value6"),
          structure (text-value: "3.14")))))))
```

Here is the resulting tabular view:

Center Alignment	No Alignment	Right Alignm...
Value4	Value2	-1245.90
Override! Right	Value5	3.14

Adding Columns Dynamically

A new dialog control action for modifying a tabular-view control has been added called add-columns, which inserts a new column at given column number. The action takes as a control-value a sequence of column structures with this syntax:

    structure
    (column: *integer*,
    text-value: *text*,
    icon: *item-or-symbol*,
    width: *integer*,
    row-values: *sequence*,
    alignment: left | right | center)

where:

- column — The column number to add.

- text-value —The column header text.

- icon — An icon to replace the column header text. The icon can be a G2 class name, an item, or a built-in GMS icon. For a list of built-in GMS icons, see the description of the image control in Chapter 43, "Custom Windows Dialogs" in the *G2 Reference Manual*.

- width — The width of the column in the tabular view. The default width is the width of the widest value in the column (text-width); thus, this attribute is optional. The value can be an integer for the width of each column, specified in dialog units, or one of these symbols:

    – header-width — The width of the column text.

    – text-width — The width of the widest value of the cells in the column.

- row-values — A sequence of cell structures used to populate the new column, where each structure has this syntax:

    structure
    (logical-id: [*integer* | *float* | *symbol* | *text*],
    text-color: *symbol*,
    background-color: *symbol*,
    text-value: *text*
    icon: *item-or-symbol*)

If row-values is omitted, all cells in the new column are blank.

This procedure adds a column to a tabular view:

```
add-column(dialog: value, pos: value, icon: item-or-value, win: class g2-window)
tag: integer = the current time;
begin
    call g2-ui-modify-custom-dialog(dialog,
        sequence
            (structure
                (control-action: the symbol add-columns,
                control-id: the symbol table,
                control-value:
                    sequence
                        (structure
                            (column: pos,
                            icon: icon,
                            row-values: sequence
                                (structure (logical-id: 0, text-value: "X-[tag]"),
                                structure (logical-id: 1, text-value: "Y-[tag]"),
                                structure (logical-id: 2, text-value: "Z-[tag]")))))),
        Win);
end
```

This code adds a column in position 1 with an icon in the column header:

```
start add-column(dialog-handle, 1, the symbol GMS-UNDO-ICON, this window)
```

Getting Updates When the User Clicks the Mouse or Presses a Key

The dialog-generic-callback for a custom dialog, if specified, is called whenever the user clicks the mouse or presses a key in the tabular-view on the dialog, with the following arguments:

- *event*: LEFT-CLICK, MIDDLE-CLICK, RIGHT-CLICK, KEY-PRESS, or any combination of modifier keys or DOUBLE combined with LEFT-CLICK, MIDDLE-CLICK, or RIGHT-CLICK, for example: SHIFT+LEFT-CLICK, CONTROL+RIGHT-CLICK, ALT+LEFT-CLICK, DOUBLE+CONTROL+LEFT-CLICK.

- *control-id*: integer

- *info*: structure
      (x: *integer*,
      y: *integer*,
      selected-rows: *sequence*,
      key: *symbol*,
      row: *integer*)

where:

- – x and y are the x-y coordinates of the mouse click in the tabular view.
- – selected-rows is a sequence of the logical-id values of the selected rows in the tabular view.
- – key is the key that was pressed, if any.
- – row is the logical-id of the row with the focus, or if no row has the focus, the top-most row in the selection. The value for row is -1 if the selection is empty.

- *user-data*: value

Time-of-Day

The time-of-day control now supports modifying the control via g2-ui-modify-custom-dialog. Although this feature was documented, it was not previously implemented. The time-of-day control supports these control actions: replace, enable, disable, hide, and show.

Toggle-Button

The toggle-button control provides a button whose state toggles between a selected and unselected state. It can contain text or an icon.

Here is a toggle button that shows two different states:

  selected

  unselected

Attribute Name	Type	Required	Default	Description
control-value	structure	yes	N/A	A structure that specifies a text value for the label, or an icon and text value to use as a tool tip, and whether the button is initially selected: structure (text-value: *text*, icon: *item-or-symbol*, selected: *truth-value*)

The icon can be a G2 class name, an item, or a built-in GMS icon. For a list of built-in GMS icons, see the description of the image control in Chapter 43, "Custom Windows Dialogs" in the *G2 Reference Manual*.

Note The control-value also accepts the icon-name attribute; however, this attribute is deprecated. You should use the icon attribute instead.

You can modify all attributes of the control-value by using the replace action. You can also modify the control by using the check and uncheck actions to toggle the selection state of the button.

Here is an example:

```
structure (control-type: the symbol toggle-button,
control-id: the symbol my-toggle-button,
height: 14,
width: 50,
left: 125,
top: 25,
response-action: the symbol respond,
control-value:
    structure (icon: the symbol gms-replace-icon,
            text-value: "Toggle Button",
            selected: true))
```

Tree-View-Combo-Box

Controlling the Size of the Tree-View-Combo-Box

You can now control the size of the tree-view-combo-box, both initially and when you modify the control.

Here are the new specific attributes for the tree-view-combo-box (the control-value has not changed):

Attribute Name	Type	Required	Default	Description
dropped-height	integer	no	false	The height of the tree view combo box when it is expanded.
dropped-width	integer	no	false	The width of the tree view combo box when it is expanded.

To modify the height and width, use the following symbols for control-action in the *modify-specification* in g2-ui-modify-custom-dialog:

Control Action Symbol	Description	Available for Controls	Required Control Value
dropped-height	Changes the height.	tree-view-combo-box	The height, in pixels, as an integer.
dropped-width	Change the width.	tree-view-combo-box	The width, in pixels, as an integer.

For example, here is the specification for the initial control height and width:

```
structure (control-type: the symbol tree-view-combo-box,
     control-id: the symbol my-combo-tree-view,
     control-value: structure (tree-layout: sequence (
         structure (item-or-name: "animal",
             children: sequence (
                 structure (item-or-name: " mammal",
                     children: sequence (structure (item-or-name: "dog"),
                                          structure (item-or-name: "cat"),
                                          structure (item-or-name: "monkey"),
                                          structure (item-or-name: "platypus"))),
                 structure (item-or-name: "amphibian",
                     children: sequence (structure (item-or-name: "lizard"),
                                          structure (item-or-name: "alligator"),
                                          structure (item-or-name: "dinosaur")))))))),
     response-action: the symbol respond,
     dropped-width: 300,
     dropped-height: 300,
     height: 15,
     width: 100,
     left: 25,
     top: 30)
```

Here is the specification for modifying the control height and width:

```
sequence
    (structure
        (control-action: the symbol dropped-height,
         control-id: the symbol my-combo-tree-view,
         control-value: 500),
    structure
        (control-action: the symbol dropped-width,
         control-id: the symbol my-combo-tree-view,
         control-value: 500))
```

Modifying a Tree-View-Combo-Box

The tree-view-combo-box control now supports the replace control action via g2-ui-modify-custom-dialog. Although this feature was documented, it was not previously implemented.

Workspace

The Native Menu System (NMS) API allows you to create popup menus on workspace controls. Previously, when Telewindows Next Generation displayed a workspace control, attempts to use the NMS API failed.

Note that NMS does not support menu bars or tool bars on workspace controls

Also, the system procedure g2-nms-is-supported has been changed. It now returns two truth-values. The first is true if NMS menu bars are supported, and the second is true if NMS popup menus are supported.

Programmatically Accepting Custom Dialogs

g2-ui-accept-custom-dialog
(*dialog-id*: integer, *win*: class g2-window)

Simulates the user clicking the OK button in a custom dialog displayed on a window. The *dialog-was-accepted* argument in the dialog-dismissed-callback is true.

Generic Dialog Callback

Custom dialogs define the dialog-generic-callback attribute in the dialog component structure, which specifies a procedure to call when a dialog update occurs. It is called when the user:

- Edits a control value in the dialog.

- Dismisses the dialog.

- Changes the selected cells in a grid-view.

- Clicks the mouse or presses a key in a grid-view or tabular-view.

The syntax for the generic dialog callback is:

my-dialog-generic-callback
(*event*: symbol, *win*: class g2-window, *dialog-id*: integer, *item*: value, *info*: structure, *user-data*: value)

Argument	Description
event	The event that occurred. The options are: • All controls: USER-EDIT and DISMISSED • grid-view: SELECTION-CHANGED • grid-view or tabular-view: LEFT-CLICK, MIDDLE-CLICK, RIGHT-CLICK, KEY-PRESS, or any combination of modifier keys or DOUBLE combined with LEFT-CLICK, MIDDLE-CLICK, or RIGHT-CLICK, for example: SHIFT+LEFT-CLICK, CONTROL+RIGHT-CLICK, ALT+LEFT-CLICK, DOUBLE+CONTROL+LEFT-CLICK.
win	The window on which the dialog is displayed.
dialog-id	The dialog handle that is returned by the g2-ui-post-custom-dialog system procedure.
item	For the DISMISSED event, true if the user clicked the OK button, or false if the user clicked the Cancel button. For all other events, the The control-id of the dialog component structure for the updated component.
info	A structure, which depends on the type of event. For information on the USER-EDIT and DISMISSED events, see below. For information on the mouse and key click events, see "Grid-View" on page 54 and "Tabular-View" on page 64.
user-data	Any user-defined value.

The *info* structure for the USER-EDIT event has this syntax:

```
structure
(control-id: control-id,
control-value: new-value,
all-control-values: sequence)
```

The *info* structure for the DISMISSED event has this syntax:

```
structure
(dialog-ok: truth-value,
all-control-values: sequence)
```

Dialog Demo

The `dialogs-demo.kb` has the following enhancements:

- Added Edit Alert button and dialog.

- Modified relative location of the dialogs.

- Alert Viewer appears in an MDI child window.

- Delete button only enabled when items are available.

Windows User Interface

Chart View

Allowing Sequence of Structures for Options

The g2-ui-create-chart-view and g2-ui-modify-chart-view system procedures
permit either a structure or a sequence of structures for the *options* and *properties*
arguments, respectively. A sequence of structures are guaranteed to be applied in
order.

Graph Annotation Types

The graph annotation types such as PEGAT-ARROW-SMALL, as documented in
the help file by Gigasoft, are now supported. For example:

```
start g2-ui-create-chart-view
("Title", the symbol NONE, structure
    (ydata: sequence(1,3,1),
    show-annotations: true,
    graph-annotation-x: sequence(1.0, 3.0),
    graph-annotation-y: sequence(2.8, 1.5),
    graph-annotation-type: sequence(the symbol PEGAT-THICKSOLIDLINE,
        the symbol PEGAT-ARROW-LARGE),
    graph-annotation-text: sequence("Start", "End")),
this window)
```
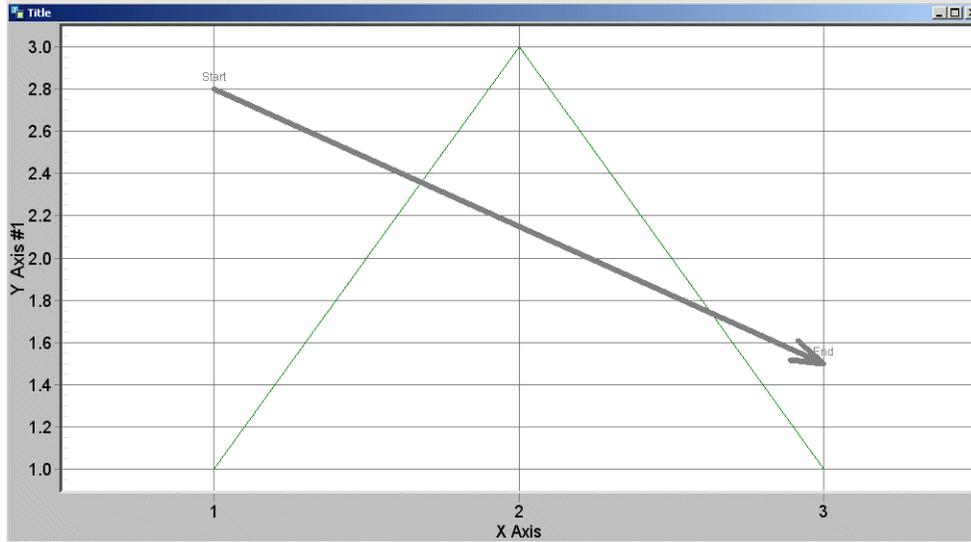
2D Contour Charts

2D contour plotting mode of charts, which you get by specifying a plotting-method of pegpm-contourlines or pegm-contourcolors for chart of type scigraph or sgraph now works.

The charts.kb sample KB provides examples.

Here is a 2D contour lines chart:
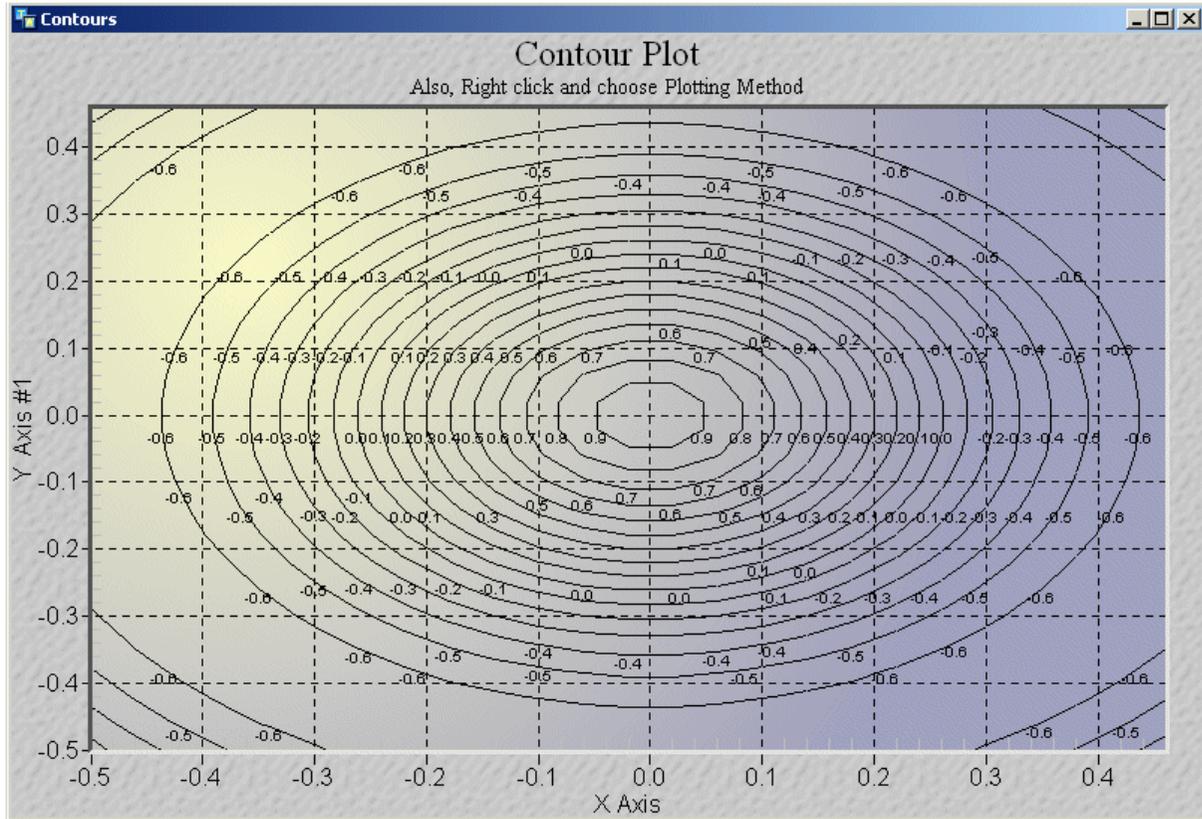
Here is a 2D contour color chart:
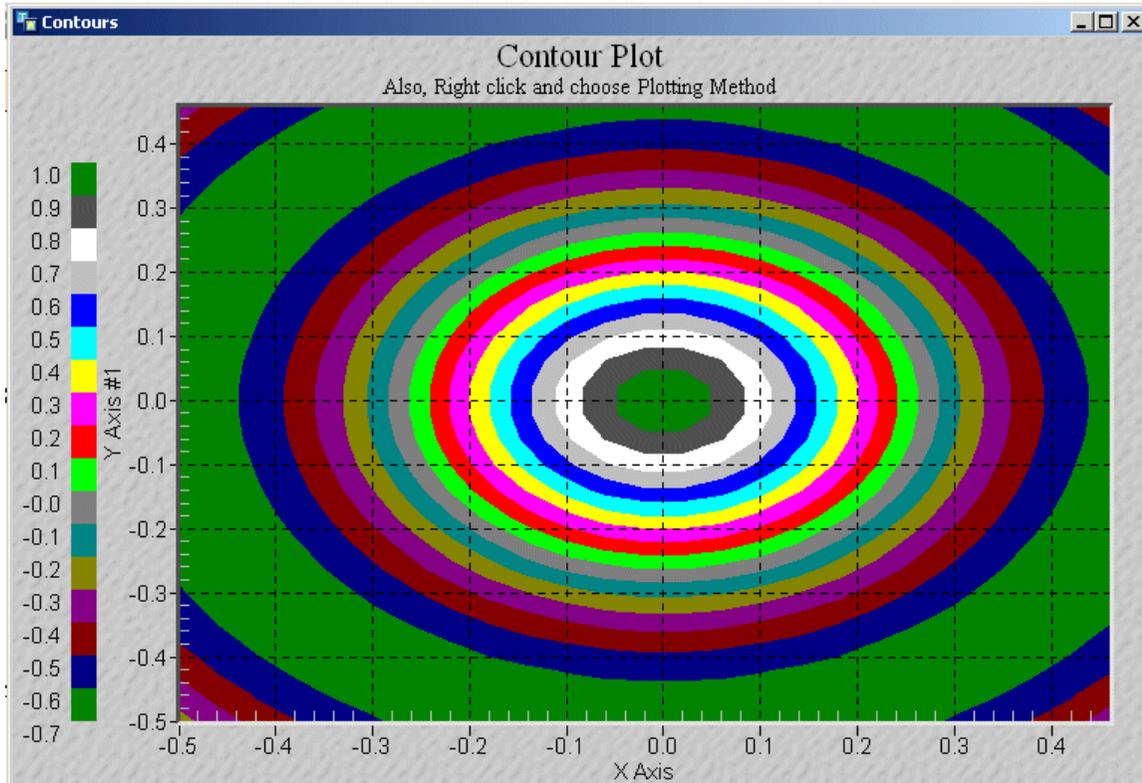
2D and 3D Charts in Floating and Dockable Panes and as Dialog Controls

2D and 3D charts can be displayed in dockable or floating panes, as well as within a custom dialog. Previously, they could only be displayed in MDI child windows.

To support this, the g2-ui-create-chart-view system procedure now allows the container attribute of the *options* structure to be a dialog handle, and the neighbor attribute to be a control-id for a control in a custom dialog. The chart view is placed in the dialog, replacing the existing control, which must be a label. Also, the dock attribute must be the symbol within.

For the complete syntax, see g2-ui-create-chart-view in Chapter 33 "User Interface Operations" in the *G2 System Procedures Reference Manual*.

Rendering 2D and 3D Charts to Image Files

You can render 2D and 3D charts as image files by using the export action in the g2-ui-manage-chart-view system procedure, whose syntax is:

g2-ui-manage-chart-view
    (*action*: symbol, *handle*: integer, *arg*: value, *win*: class g2-window)

When *action* is export, *arg* can be a text or a structure, as follows:

- text — The pathname of a file on the client.

  or

- structure
  (pathname: *text*,
   width: *integer*,
   height: *integer*,
   format: *symbol*)

  where:

  - pathname — The pathname of a file on the client.

  - width — The width, in pixels, of the exported image. The default is the width of the view.

  - height — The height, in pixels, of the exported image. The default is the height of the view.

  - format — The format of the image file, which is one of these symbols: JPEG, PNG, BMP, or WMF. The default is computed from the file type of the pathname.

For the complete syntax, see g2-ui-manage-chart-view in Chapter 33 "User Interface Operations" in the *G2 System Procedures Reference Manual*.

More Efficient Real-Time Updates

You can update a set of points in a chart view by using the partial-update action in the g2-ui-manage-chart-view system procedure, whose syntax is:

g2-ui-manage-chart-view
  (*action*: symbol, *handle*: integer, *arg*: value, *win*: class g2-window)

When *action* is partial-update, *arg* is a structure with the following syntax:

structure
(start: *integer*
 count: *integer*)

where:

- start is the index of the first data point to update. The default value is 0.

- count is the number of data points to update. The default is 1.

The partial-update action redraws the given set of points, for all subsets, on the chart view. It does not update any other properties of the chart view; you should still perform a complete update periodically by using the update action. For details, see "Chart Views" on page 509.

HTML View

As of G2 Version 8.2 Rev. 1, HTML views can be displayed as dialog controls within a custom dialog. However, the release notes did not include this feature.

Specifically, the g2-ui-create-html-view system procedure allows the container attribute in the *options* structure argument to be a dialog handle and the neighbor attribute to be a control-id of a control in a custom dialog. When container is a dialog handle, the dock attribute must be the symbol within.

Property Grid

The g2-ui-manage-property-grid system procedure supports additional values for the *action* argument:

- clear — Clears the property grid (*arg* is ignored).

- populate — Populates the property grid, where *arg* is the complete new contents of the property grid, in the same format as the contents option of g2-ui-create-property-grid.

For details, see "Property Grid Views" in Chapter 33, "User Interface Operations" in the *G2 System Procedures Reference Manual*.

Shortcut Bar

As of G2 Version 8.2 Rev. 1, shortcut bars can be displayed as dialog controls within a custom dialog. However, the release notes did not include this feature.

Specifically, the g2-ui-create-shortcut-bar system procedures allows the container attribute in the *options* structure argument to be a dialog handle and the neighbor attribute to be a control-id of a control in a custom dialog. When container is a dialog handle, the dock attribute must be the symbol within.

Listbar-Style Shortcut Bar

You can display an arbitrary view in the folders of a listbar-style shortcut bar, for example, a tree view or a dialog view. To support this feature, the container option of any of the g2-ui-create-*view* system procedures and the g2-ui-post-custom-dialog may now be the handle of a listbar-style shortcut bar, in which case the neighbor option is the number of the folder within the listbar into which to create the view.

In addition, you can create shortcut bars and listbars with no items initially, then add native views to those folders later.

The following examples show how to create an empty listbar and add views to the folders incrementally.

This procedures creates the listbar with three folders, two of which are initially empty:

```
f()=sequence(
    structure(label: "Folder Zero", icon: cp-1),
    structure(label: "Folder One", items: sequence(
        structure(label: "First", icon: the symbol GMS-SAVE-ICON),
        structure(label: "Second", icon: CP-1),
        structure(label: "Third", icon: the symbol SAX-PARSER)))),
    structure(label: "Folder Two", icon: the symbol GMS-PASTE-ICON))
```

The following button creates the listbar:


```
start g2-ui-create-shortcut-bar(f(), the symbol
    CB, structure(style: the symbol LISTBAR,
    title: "ListBar", dock: the symbol RIGHT), this
    window)
```

This button adds a tree-view to the first folder in the listbar:

start g2-ui-create-tree-view("Tree", the
    symbol CB, structure(height: 120, container:
    "ListBar", tree: sequence("AA", "BB", "CC")),
    this window)

This button adds a dialog to the first folder in the listbar:

start g2-ui-post-custom-
    dialog(structure(dialog-title: "Dialog", dialog-
    width: 100, dialog-height: 120, resizable: true,
    container: "ListBar", components: ok(),
    dialog-update-callback: cb-update), false,
    this window)
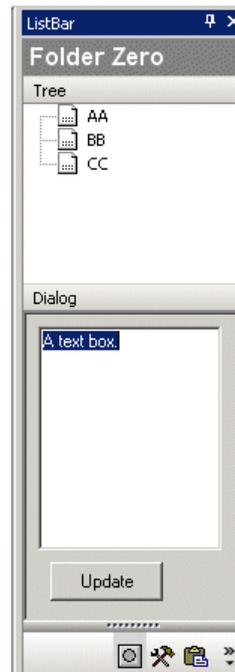
This button adds another dialog to the first folder in the listbar:
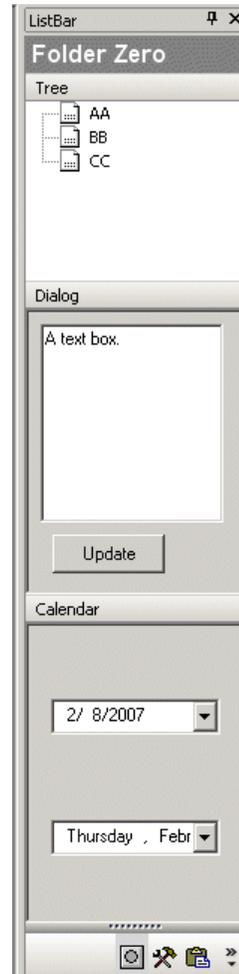
This button adds a property grid to the third folder in the listbar:
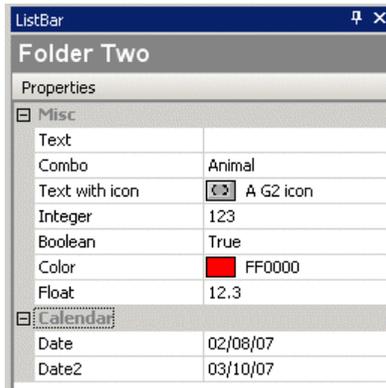
Status Bar

You can use the following system procedures to configure and modify the status bar of a Telewindows Next Generation window:

g2-ui-configure-status-bar
>   (*options*: structure, *win*: class g2-window)

>   Configures the status bar. *Options* is a structure with the following syntax:

>   structure
>   (visible: *truth-value*,
>   callback: *symbol-or-procedure*,
>   min-height: *integer*,
>   panes: *sequence*)

>   where:

>   • visible — Whether to show or hide the status bar. Set to false to hide the status bar.

>   • callback — A callback procedure for mouse clicks on the status bar, which has this syntax:

>   >   my-status-bar-callback
>   >   (*event*: symbol, *win*: class g2-window, *handle*: integer, *pane-id*: value, *info*: structure, *user-data*: value)

>   >   The options for *event* are LEFT-CLICK, MIDDLE-CLICK, RIGHT-CLICK, or any combination of these events plus modifiers, for example, CONTROL+LEFT-CLICK, SHIFT+RIGHT-CLICK, CONTROL+ALT+LEFT-CLICK, DOUBLE+LEFT-CLICK.

The *info* structure contains the mouse X and Y position.

The *handle* argument is not meaningful for status bars.

- min-height — The minimum height of the status bar, in pixels.

- panes — A sequence of panes in the status bar, where each pane is a structure with the following syntax:

```
structure
(id: integer | symbol | text,
text: text,
icon: item-or-symbol,
width: integer | symbol,
background-color: symbol | truth-value,
foreground-color: symbol | truth-value,
alignment: symbol,
tooltip: text,
visible: truth-value,
enabled: truth-value,
borders: truth-value,
user-data: item-or-value)
```

where:

- id — A user-supplied identifier for the pane, which is supplied to other actions. The ID is not required to be unique. The default value is the value of the text attribute. See description below.

- text — A text string to show in the pane. The default is "".

- icon — An icon to show in the pane. The default is false. The icon can be a G2 class name, an item, or a built-in GMS icon. For a list of built-in GMS icons, see the description of the image control in Chapter 43, "Custom Windows Dialogs" in the *G2 Reference Manual*.

- width — The width of the pane in pixels, or the symbol fit, which fits the width to the text. The default is the symbol fit.

- background-color — The background color. The default is false, which means transparent.

- foreground-color — The foreground color. The default is false, which means black.

- alignment — The alignment of the icon and text in the pane. The options are: left, center, or right. The default is the symbol left.

- tooltip — The tooltip to show when hovering the mouse over the pane. The default is "".

- visible — Whether the pane is visible. The default is true.

- enabled — Whether the pane is enabled. The default is true.

- borders — Whether the pane has borders. The default is true.

- user-data — Arbitrary user data attached to the pane, which is returned in the callback. The default is the symbol none.

A pane ID is the name of a pane in the status bar. It is useful to give panes names so they can be referenced without regard to their absolute position in the status bar, which may change as panes are added and removed.

A pane ID can be any integer, a symbol, or a text string. The following integer and symbol values are reserved for use by G2:

- 0 — The built-in documentation-line pane.

- -1 — Indicates beyond the last pane (for ADD-PANE only).

- g2-documentation-line — Another name for the documentation-line pane.

- g2-security-icon — The pane showing the padlock icon, for SSL secured connections.

- g2- — Any other symbol beginning with g2- is reserved for future use.

A pane ID is not required to be unique. In that case, the ID refers to some pane.

g2-ui-manage-status-bar
(*action*: symbol, *arg*: value, *win*: class g2-window)

Manages existing status bars, where *action* is one of these symbols with corresponding values for *arg*, where applicable:

- hide — Hides the status bar, which is the same as calling g2-ui-hide-status-bar.

- show — Shows the status bar, which is the same as calling g2-ui-show-status-bar.

- set-text — Sets the text of the documentation-line pane in the status bar, which is the same as calling g2-ui-set-status-bar-text. The *arg* is a text value.

- configure — Configures the status bar, which is the same as calling g2-ui-configure-status-bar. The *arg* is a structure, which is the same as the *options* argument to g2-ui-configure-status-bar.

- set-min-height — Sets the minimum height of the status bar, in pixels. The *arg* is an integer.

- set-callback — Sets the callback procedure. The *arg* is a symbol or procedure.

- **add-pane** — Adds a pane to the status bar. The *arg* is a structure, which is the same as the pane structure for g2-ui-configure-status-bar, with one additional attribute:

    - **position:** *integer | symbol | text* — The ID of the pane to insert before. The default is -1, which means add the pane on the end.

- **modify-pane** — Modifies the pane specified by the ID. The *arg* is a structure, which is the same as the pane structure for g2-ui-configure-status-bar.

- **remove-pane** — Removes a pane. The *arg* is the ID of the pane to remove or structure (id: *the-id-of-pane-to-remove*).

Here is an example of a status bar with multiple panes and the code used to create it:


```
start g2-ui-configure-status-bar
(structure
(callback: cb,
panes: sequence
    (structure(id: "X", text: "Hello", user-data: 123456),
    structure(id: "Y", text: "World"))), this window)
```

This example shows how to add a pane to the status bar:
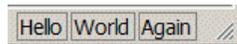

```
start g2-ui-manage-status-bar
(the symbol ADD-PANE, structure(id: "X", text: "Again"), this window)
```

This example shows how to modify a pane by adding an icon:
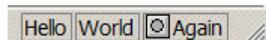

```
start g2-ui-manage-status-bar
(the symbol MODIFY-PANE, structure(id: "X", icon: cp-1), this window)
```

This example shows how to modify the background color of a pane:
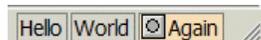

```
start g2-ui-manage-status-bar
(the symbol MODIFY-PANE, structure(id: "X", background-color: the symbol WHEAT),
this window)
```

Tree View

Right-Click Without Selection

The g2-ui-create-tree-view system procedure supports the following attribute in the *options* structure:

> right-click-selects-node — Whether right-clicking a node in the tree view automatically selects the node. The default is true. Set to false to disable automatic selection when right-clicking the node.

For a complete description of the system procedure, see "Tree Views" in Chapter 33, "User Interface Operations" in the *G2 System Procedures Reference Manual*.

Left-Click Events

The g2-ui-create-tree-view system procedure callback procedure supports new events:

- left-click, which invokes the callback when the user clicks the left mouse button a tree node.

- double+left-click, which invokes the callback when the user double-clicks the left mouse button.

- Modifier keys in combination with clicking the left mouse button, for example, shift+left-click.

For a complete description of the callback, see "Tree Views" in Chapter 33, "User Interface Operations" in the *G2 System Procedures Reference Manual*.

Choosing Directories

You can use the following system procedure to display a dialog for choosing a directory, as opposed to a file. For example, you would use this system procedure to allow users to choose a directory in which to store log files.

g2-ui-choose-directory
> (*pathname*: text, *options*: structure, *win*: class g2-window)
> –> *directory*: text

> Displays a dialog in *win* for choosing a directory. Returns the pathname of the chosen directory or "" if the dialog is cancelled.

> *Pathname* is the pathname of the initially selected directory in the dialog. Use "" to display the platform's default directory, which is generally the current directory.

Options is a structure with the following syntax:

```
structure
(caption: text,
ok-button-label: text,
cancel-button-label: text,
client-side: truth-value,
root: text)
```

where:

- caption — The text to display as a caption for the dialog. The default is "".

- ok-button-label — Alternative text to display on the OK button. The default is "OK".

- cancel-button-label — Alternative text to display on the Cancel button. The default is "Cancel".

- client-side — Whether to browse the client's file system. The default is false, which displays the server's file system in the dialog.

- root — Pathname of a directory to be considered the root directory, as a text. The user will only be allowed to select a subdirectory of this directory. Default value is "/" for G2 running on a UNIX machine or "X:\" for G2 running on a Windows machine, where "X" is the drive letter for the current default directory.

Platform-Independent UI Enhancements

Mouse Hover Events

G2 provides the following grammar in user interface clauses of item configurations to detect mouse hover events:

```
hovering [modifiers+] the mouse [over any class]
  {implies action | does nothing}
```

The action is executed if G2 receives a mouse hover event, which is sent whenever the mouse does not move more than a certain amount for a period of time, which is determined by the operating system. On Windows, the default is 4 pixels and 400 milliseconds, respectively.

For example:

```
configure the user interface as follows:
when in developer mode:
hovering the mouse over any connection-post implies umc
```

where umc is a user menu choice:
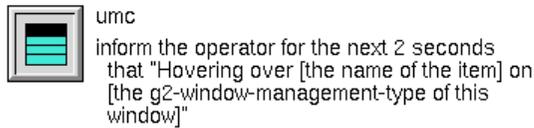
Note that the g2-last-input-event and g2-last-input-event-info system procedures ignore mouse hover events. At startup, the event returned is UNKNOWN.

Custom Grammar in the G2 Text Editor

The G2 Text Editor supports custom prompting and highlighting for custom grammar. To support this, the g2-ui-launch-editor system procedure supports the following attribute in the *options* structure:

- grammar is either a symbol naming a class or a structure with the following syntax:

      structure
      (class: *symbol*,
      attribute: *symbol*,
      defining-class: *symbol*)

Use the following new system procedure to launch the G2 Text Editor on a given text string:

g2-ui-edit-text
    (*text*: text, *options*: structure, *window*: class g2-window)
    –> *text*: text, *status*: symbol

Launches an editor on the given *text* string, waits for editing to finish, and returns the resulting text string and exit status. The *status* is either the symbol OK or the symbol CANCELLED.

The syntax for the *options* structure is the same as g2-ui-launch-editor:

structure
(attribute: *symbol*,
exclude: *sequence,*
preferred-editor: classic | native,
grammar: *symbol* | *structure*)

For more information, see g2-ui-launch-editor on page 539 in Chapter 33, "User Interface Operations" in the *G2 System Procedures Reference Manual*.

Programmatically Inserting Text into the G2 Text Editor

g2-ui-insert-text-into-current-editor
(*text*: text, *options*: structure, *window*: class g2-window)

Inserts *text* at the current cursor position in the current G2 Text Editor. Currently, no options are supported.

Visible Grid on Workspaces

The Drawing Parameters system table has a new attribute, alignment-grid, which controls a visible grid and a snap grid on KB workspaces.

The snap grid is disabled by default. To enable it, use the following grammar in the alignment-grid attribute:

grid [, line color: *color*] [, line pattern: *symbol*] [, snap to *grid*]

where *grid* can be either an integer, giving the spacing in workspace units for both X and Y, or a pair on integers (*integer*, *integer*) giving spacings for X and Y. For example:

100, line color: gray, line pattern: long dash, snap to 10

When given as a structure, the syntax is:

structure
(spacing: sequence(*integer*, *integer*),
line-color: *symbol*,
line-pattern: *symbol*,
snap-to: sequence(*integer*, *integer*))

The default value is:

structure
(spacing: sequence (50,50),
line-color: the symbol foreground,
line-pattern: the symbol coarse-dot)

The visible grid is invisible by default. To view the grid, do one of the following:

• Choose View > Toggle Visible Grid.

• Enter Ctrl + G with the mouse over a KB workspace.

• Execute the toggle-visible-grid system command, using the g2-system-command system procedure. For details, see g2-system-command in Chapter 33, "User Interface Operations" in the *G2 System Procedures Reference Manual*.

- Set the view-preferences of a KB workspace to visible-grid, or conclude the visible-grid attribute in the view-preferences structure of a KB workspace, for example:

    conclude that the view-preferences of this workspace = structure(visible-grid: true)

If the snap grid is enabled, and both a constrain moving ... item configuration and the snap grid apply to a particular item, then the item configuration takes precedence and the snap grid is ignored.

For example, here is a workspace whose alignment-grid is set to:

    100, line color: gray, line pattern: short dash, snap to 10
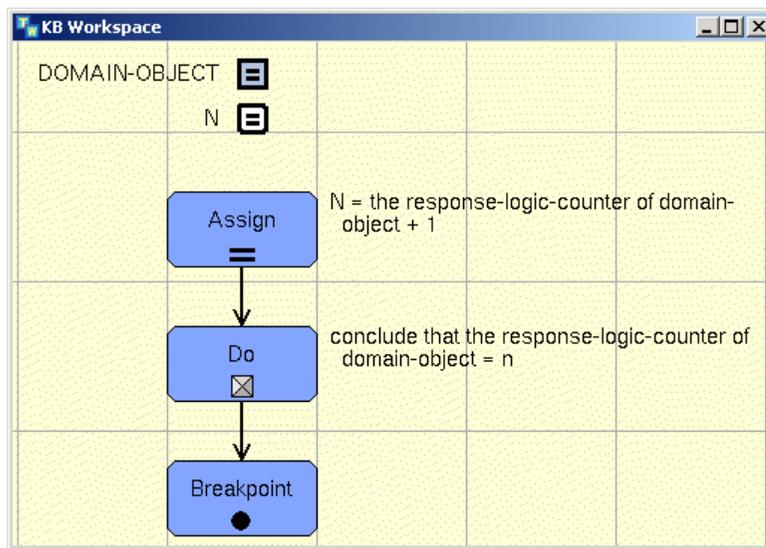
Interactively Resizing Objects and Changing Connection Vertices

You can interactively resize objects, using selection handles on the object. To support this feature, the Drawing Parameters system table defines the show-selection-handles attribute, which defaults to true. For example:
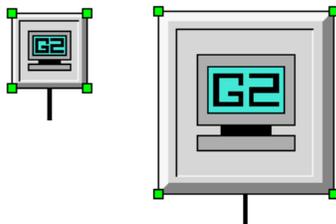
You can also interactively change the connection vertices of a connection, using handles on the connection. For example:
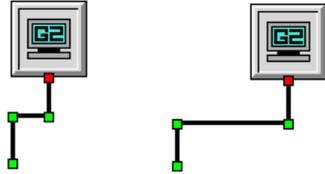
When show-selection-handles is true, the default, selection handles appear and the change size menu choice does not appear on items. When show-selection-handles is false, selection handles do not appear and the change size menu choice appears on items.

Font Changes on Texts

Changing the font on any text, such as free-text, borderless-free-text, or message, by choosing the font menu choice permanently changes the font. Previously, it only changed the font until the next KB reset.

Computational Capabilities

Date and Time Formats for Attributes

For attributes of type quantity, float, or integer, you can use the following date and time formatting statements in the class-specific-attributes of a class description and in the display-formats of a readout table:

    formatted as mm-dd-yyyy-hh-mm-ss
    formatted as dd-mm-yyyy-hh-mm-ss
    formatted as yyyy-mm-dd-hh-mm-ss
    formatted as mm-dd-yyyy-hh-mm-ss-am-pm
    formatted as mm-dd-yyyy-hh-mm-am-pm
    formatted as yyyy-mm-dd-hh-mm-ss-am-pm
    formatted as dd-mm-yyyy-hh-mm-ss-am-pm
    formatted as dd-mm-yyyy-hh-mm-am-pm
    formatted as yyyy-mm-dd-hh-mm-am-pm
    formatted as mm-dd-yyyy-hh-mm
    formatted as dd-mm-yyyy-hh-mm
    formatted as yyyy-mm-dd-hh-mm

formatted as mm-dd-yyyy
formatted as dd-mm-yyyy
formatted as yyyy-mm-dd
formatted as mm-yyyy
formatted as yyyy-mm
formatted as dd-hh-mm-ss as an interval
formatted as hh-mm-ss as an interval
formatted as hh-mm as an interval
formatted as mm-ss as an interval
formatted as hh.hh as an interval

The following table shows examples of all date and time formats:

a test	
Notes	OK
Item configuration	none
Names	none
Mm dd yyyy hh mm ss	02/28/2007 15:00:17
Dd mm yyyy hh mm ss	28/02/2007 14:44:20
Yyyy mm dd hh mm ss	2007/02/28 14:44:56
Mm dd yyyy hh mm ss am pm	02/28/2007 2:45:11 p.m.
Mm dd yyyy hh mm am pm	02/28/2007 3:00 p.m.
Yyyy mm dd hh mm ss am pm	2007/02/28 2:48:41 p.m.
Dd mm yyyy hh mm ss am pm	28/02/2007 2:43:43 p.m.
Dd mm yyyy hh mm am pm	28/02/2007 2:44 p.m.
Yyyy mm dd hh mm am pm	2007/02/28 5:12 p.m.
Mm dd yyyy hh mm	02/28/2007 14:43
Dd mm yyyy hh mm	28/02/2007 15:04
Yyyy mm dd hh mm	2007/02/28 14:43
Mm dd yyyy	03/01/2007
Dd mm yyyy	02/03/2007
Yyyy mm dd	2007/02/28
Mm yyyy	03/2007
Yyyy mm	2007/02
Dd hh mm ss as an interval	0:00:16:40
Hh mm ss as an interval	1:02:03
Hh mm as an interval	2:01
Mm ss as an interval	100:07
Hh.hh as an interval	1.33

Version Control Enhancements

G2 provides a comprehensive version control system, which leverages the G2 change log facility, to allow:

- Tagging attributes of G2 objects that support change logging (for example, procedures, rules, class definitions) within a module, as well as tagging all the attributes that support change logging of all items in a module.

- Reverting change-loggable attributes of individual items or of all items in a module to a previous revision. Note that this only works on items that still exist; G2 does not preserve the change log of deleted items.

- Deleting change log entries.

- Commenting change log entries.

- Performing a "diff" operation on two texts or two change log entries.

The system procedures are located under G2 Version Control in `sys-mod.kb`.

View Change Log

Each entry in the View Change Log command includes the following columns, some of which are new or renamed:

- Attribute — The name of the changed attribute.

- Revision — The revision number for the change.

- Value — The value of the attribute for that revision.

- Module Version — When change-logging is enabled on a particular module, each time the module is saved, it is given a unique version number. The module version and corresponding date and time of the save are visible in the Saving Parameters system table.

- Timestamp — The date and time of the edit.

- Author — The user name of the author.

- Tags — User-defined tags.

Previously, it included Attribute, Value, Version, Timestamp, and Author.

For example:

In addition:

- In earlier releases, the change log information showed the *previous* value of the edited attribute as the first entry; now, it shows the current value.

- When editing an item produces no changes to the item, G2 no longer adds an entry to the change log. Previously, G2 added an entry to the change log every time an attribute was edited, even if no change was made.

- The change-log and authors attribute is now defined for the action-button class; previously, it was not defined.

Change Logging System Procedures

Use the following system procedures for programmatically accessing the change log entry for an item.

g2-get-change-log-entry
    (*item*: class item, *attribute-name*: symbol, *identifier*: structure)
    –> *change-log-entry*: structure

Gets the change log entry for the *attribute-name* attribute of *item*. Use *identifier* to specify a revision, timestamp, or tag to further identify the change log entry to get, which is a structure with this syntax:

structure
(revision: *integer*,
timestamp: *structure*, {see *change-log* return value structure for syntax}
tag: *symbol*)

If *identifier* is empty, it gets the current revision. The structure should specify only one of these attributes.

The *change-log-entry* structure has this syntax:

```
structure
(attribute: symbol,
revision: integer,
comment: text,
tags: sequence, {a sequence of symbolic tags}
text-value: text,
module-version: integer,
timestamp: structure
    (year: integer,
    month: integer,
    date: integer,
    hours: integer,
    minutes: integer),
author: symbol)
```

Note that you cannot use a reserved word as a symbol in the tags sequence or the symbol none, which is not a reserved word.

g2-tag-change-log-entry
(*item*: class item, *attribute-name*: symbol, *identifier*: structure, *tag*: symbol)
−> *change-log*: structure

Adds a tag to the change log entry for the *attribute-name* attribute of *item*. Use *identifier* to specify a revision or timestamp to further identify the change log entry to tag, which is a structure with this syntax:

```
structure
(revision: integer,
timestamp: structure)
```

If *identifier* is empty, it tags the current revision. The structure should specify only one of these attributes.

The *change-log* structure has the same syntax as g2-get-change-log-entry or an empty structure if the tagging failed.

g2-delete-change-log-tag
(*item*: class item, *attribute-name*: symbol, *tag*: symbol)
−> *change-log*: structure

Deletes the value of the specified *tag* for the *attribute-name* attribute of *item*.

The *change-log* structure is the value of the change log entry after the tag has been deleted. It has the same syntax as g2-get-change-log-entry or an empty structure if the deletion failed.

g2-delete-change-log-entry
(*item*: class item, *attribute-name*: symbol, *identifier*: structure)
−> *result*: truth-value

Deletes a change log entry for the *attribute-name* attribute of *item* and returns true if the deletion was successful, false otherwise. Use *identifier* to specify a revision, timestamp, or tag to further identify the change log entry. See g2-get-change-log-entry for a description.

g2-set-change-log-entry-comment
(*item*: class item, *attribute-name*: symbol, *identifier*: structure, *comment*: text)

Sets the value of the comment attribute of the change log entry for the specified *attribute-name* attribute of *item*. Use *identifier* to specify a revision, timestamp, or tag to further identify the change log entry. See g2-get-change-log-entry for a description.

g2-get-change-log-entry-comment
(*item*: class item, *attribute-name*: symbol, *identifier*: structure)
−> *comment*: text

Returns the value of the comment attribute of the change log entry for the specified *attribute-name* attribute of *item*. Use *identifier* to specify a revision, timestamp, or tag to further identify the change log entry. See g2-get-change-log-entry for a description.

g2-revert-change-log-entry
(*item*: class item, *attribute-name*: symbol, *identifier*: structure)
−> *change-log*: structure

Reverts the value of the *attribute-name* attribute of *item* to the particular previous value specified by *identifier*. Use *identifier* to specify a revision, timestamp, or tag to further identify the change log entry. See g2-get-change-log-entry for a description.

The *change-log* structure has the same syntax as g2-get-change-log-entry or an empty structure if the revert failed.

g2-tag-module
(*module-name*: symbol, *identifier*: structure, *tag*: symbol)
−> *result*: truth-value

Sets the value of the tags attribute of the change log entry for every attribute that has a change log of every item in the specified *module-name* with *tag*, and returns true if the tagging was successful, false otherwise. If the tags attribute already has a value, this system procedure adds to the existing sequence. For example, if the value of the tags attribute is sequence(abc), calling g2-tag-module with xyz as the *tag* results in sequence(abc, xyz). Use *identifier* to specify a timestamp to further identify the change log entry. If *identifier* is empty, it tags the current revision. See g2-tag-change-log-entry for a description.

g2-revert-module
(*module-name*: symbol, *identifier*: structure)
–> <u>result</u>: truth-value

Reverts the change log entry for every attribute of every item in the specified *module-name* for which change logging is enabled, and returns true if the revert was successful, false otherwise. Use *identifier* to specify a revision, timestamp, or tag to identify the change log entry to which to revert. See g2-get-change-log-entry for a description.

g2-disable-change-logging-on-item
(*item*: class item)

Disables change logging on the specified *item* in a module. If change logging is already disabled on the module or on the item, this system procedure has no effect.

g2-enable-change-logging-on-item
(*item*: class item)

Enables change logging on the specified *item* in a module. If change logging is disabled on the module, this system procedure generates an error. If change logging is already enabled on the item, this system procedure has no effect.

Text "Diff" System Procedures

Use the following system procedures to perform a "diff" operation on two texts and change log entries. You specify an external program to perform the "diff" operations; G2 does not provide an internal capability for performing "diff" operations.

g2-set-external-diff-specification
(*spec*: structure)
–> <u>result</u>: truth-value

Change the command used for performing the "diff" operation when using g2-diff-texts. The *spec* structure has this syntax:

    structure
    (pathname: *text*,
    command-line-options: *text*)

where:

- pathname is the path to the "diff" program to use.

- command-line-options is a text string of command-line options for the "diff" program.

g2-diff-texts
(*text1:* text, *text2*: text)
–> *result*: structure

Returns a structure that describes the difference between two texts. The syntax for *result* is:

    structure
    (diff-output: *text*,
    diff-error: *text*)

where:

- diff-output describes the differences in the two input text or an empty text, if the diff fails.

- diff-error is the empty text if the "diff" succeeds, or the error text.

By default, the system procedure uses /usr/bin/diff on UNIX and C:\WINDOWS\system32\fc.exe on Windows. To change the "diff" command that g2-diff-texts uses, use g2-set-external-diff-specification.

g2-diff-change-log-entries
(*item*: class item, *attribute-name*: symbol, *identifier-1*: structure, *identifier-2*: structure)
–> *result*: structure

Returns a structure that describes the difference between two change log entries, where *change-log-spec-1* and *change-log-spec-2* are of the format returned by g2-get-change-log-entry. The change log specifications do not have to be of the same type, for example, one could be a timestamp and the other could be a tag.

This system procedure uses the same "diff" programs as g2-diff-texts.

The syntax for *result* is the same as for g2-diff-texts.

Examples

The following examples refer to the following item named my-umc, whose change log shows edits to the names, label, action, and applicable-class attributes of the item, including three revisions of the label attribute:

Attribute	Revision	Value	Module Version	Timestamp	Author	Tags
Label	2	post-hello-world	63	7 Feb 2007 3:09 p.m.	nrs	none
Applicable class	0	object	63	7 Feb 2007 3:09 p.m.	nrs	none
Action	0	post "Hello world"	63	7 Feb 2007 3:08 p.m.	nrs	none
Label	1	do-action	63	7 Feb 2007 3:08 p.m.	nrs	none
Label	0	do-something	63	7 Feb 2007 3:08 p.m.	nrs	none
Names	0	MY-UMC	63	7 Feb 2007 3:07 p.m.	nrs	none

Tagging Change Log Entries

Here is a generic procedure that tags the change log entry of an item with a given identifier:

```
tag-change-log-entry (item: class item, attribute-name: symbol, identifier: structure,
    new-tag: symbol)
resulting-struct: structure;
begin
    resulting-struct = call g2-tag-change-log-entry(item, attribute-name, identifier,
        new-tag);
    post "[resulting-struct]"
end
```

This action button tags the change log entry for the names attribute of my-umc with the given timestamp with the symbol G283R0:


```
start tag-change-log-entry(my-umc, the
    symbol names, structure(timestamp:
    structure(year: 2007, month: 2, date: 7,
    minutes: 7, hours: 12+3)), the symbol
    G283R0)
```

TAG-CHANGE-LOG-ENTRY

Here is the resulting change log and message board:

Attribute	Revision	Value	Module Version	Timestamp	Author	Tags
Label	2	post-hello-world	63	7 Feb 2007 3:09 p.m.	nrs	none
Applicable class	0	object	63	7 Feb 2007 3:09 p.m.	nrs	none
Action	0	post "Hello world"	63	7 Feb 2007 3:08 p.m.	nrs	none
Label	1	do-action	63	7 Feb 2007 3:08 p.m.	nrs	none
Label	0	do-something	63	7 Feb 2007 3:08 p.m.	nrs	none
Names	0	MY-UMC	63	7 Feb 2007 3:07 p.m.	nrs	G283R0

```
#86   3:15:51 p.m.   structure
  (ATTRIBUTE: the symbol NAMES,
  REVISION: 0,
  COMMENT: "",
  TAGS: sequence (the symbol G283R0),
  TEXT-VALUE: "MY-UMC",
  MODULE-VERSION: 63,
  TIMESTAMP: structure (YEAR: 2007,
    MONTH: 2,
    DATE: 7,
    HOURS: 15,
    MINUTES: 7),
  AUTHOR: the symbol NRS)
```

Getting Change Log Entries

Here is a generic procedure that posts the change log entry for an attribute of an item with a given identifier:

```
post-change-log-entry (item: class item, attribute-name: symbol, tag: structure)
result: structure;
begin
 result = call g2-get-change-log-entry (item, attribute-name, tag);
 post "[result]"
end
```

This action button gets the change log entry for the names attribute of my-umc tagged with the symbol G283R0:

Find By Tag   start post-change-log-entry(my-umc, the
symbol names, structure(tag: the symbol
G283R0))

POST-CHANGE-LOG-ENTRY

Here is the resulting message board:

```
#86   3:15:51 p.m.   structure
  (ATTRIBUTE: the symbol NAMES,
  REVISION: 0,
  COMMENT: "",
  TAGS: sequence (the symbol G283R0),
  TEXT-VALUE: "MY-UMC",
  MODULE-VERSION: 63,
  TIMESTAMP: structure (YEAR: 2007,
    MONTH: 2,
    DATE: 7,
    HOURS: 15,
    MINUTES: 7),
  AUTHOR: the symbol NRS)
```

This action button gets the change log entry for revision 2 of the label attribute of my-umc:

Find By Revision   start post-change-log-entry(my-umc, the
                        symbol label, structure(revision: 2))

Here is the resulting message board:

```
#89   3:21:00 p.m.   structure
  (ATTRIBUTE: the symbol LABEL,
  REVISION: 2,
  COMMENT: "",
  TAGS: sequence (),
  TEXT-VALUE: "post-hello-world",
  MODULE-VERSION: 63,
  TIMESTAMP: structure (YEAR: 2007,
    MONTH: 2,
    DATE: 7,
    HOURS: 15,
    MINUTES: 9),
  AUTHOR: the symbol NRS)
```

Deleting Change Log Entry Tags

Here is a generic procedure that deletes the change log tag for an attribute of an item:

```
delete-change-log-tag (item: class item, attribute-name: symbol, new-tag: symbol)
resulting-struct: structure;
begin
    resulting-struct = call g2-delete-change-log-tag(item, attribute-name, new-tag);
    post "[resulting-struct]"
end
```

This action button deletes the change log entry tag G283R0 for the names attribute of my-umc:

Delete Tag

Action   start delete-change-log-tag(my-umc, the
DELETE-CHANGE-LOG-TAG   symbol names, the symbol G283R0)

Here is the resulting change log and message board:

	Attribute	Revision	Value	Module Version	Timestamp	Author	Tags
	Label	2	post-hello-world	63	7 Feb 2007 3:09 p.m.	nrs	none
	Applicable class	0	object	63	7 Feb 2007 3:09 p.m.	nrs	none
	Action	0	post "Hello world"	63	7 Feb 2007 3:08 p.m.	nrs	none
	Label	1	do-action	63	7 Feb 2007 3:08 p.m.	nrs	none
	Label	0	do-something	63	7 Feb 2007 3:08 p.m.	nrs	none
	Names	0	MY-UMC	63	7 Feb 2007 3:07 p.m.	nrs	none

```
#100   4:09:18 p.m.   structure
  (ATTRIBUTE: the symbol NAMES,
  REVISION: 0,
  COMMENT: "",
  TAGS: sequence (),
  TEXT-VALUE: "MY-UMC",
  MODULE-VERSION: 63,
  TIMESTAMP: structure (YEAR: 2007,
    MONTH: 2,
    DATE: 7,
    HOURS: 15,
    MINUTES: 7),
  AUTHOR: the symbol NRS)
```

Deleting Change Log Entries

Here is a generic procedure that deletes a change log entry for an attribute of an item with a given identifier:

```
delete-change-log-entry (item: class item, attribute-name: symbol, identifier: structure)
succeeded: truth-value;
begin
    succeeded= call g2-delete-change-log-entry(item, attribute-name, identifier);
    if succeeded then
        post "deleting entry succeeded!"
    else
        post "deleting entry failed!"
end
```

This action button deletes revision 2 of the change log entry for the label attribute of my-umc:

start delete-change-log-entry(my-umc, the symbol label, structure(revision: 2))

DELETE-CHANGE-LOG-ENTRY

Here is the resulting change log:

Attribute	Revision	Value	Module Version	Timestamp	Author	Tags
Applicable class	0	object	63	7 Feb 2007 3:09 p.m.	nrs	none
Action	0	post "Hello world"	63	7 Feb 2007 3:08 p.m.	nrs	none
Label	1	do-action	63	7 Feb 2007 3:08 p.m.	nrs	none
Label	0	do-something	63	7 Feb 2007 3:08 p.m.	nrs	none
Names	0	MY-UMC	63	7 Feb 2007 3:07 p.m.	nrs	none

Commenting Change Log Entries

This action button adds a comment to revision 1 of the label attribute change log entry for my-umc:

start g2-set-change-log-entry-comment(my-umc, the symbol label, structure(revision: 1), "Changed label to do-action")

Here is a generic procedure that adds a comment to a change log entry for an attribute of an item with a given identifier:

```
post-change-log-entry-comment(item: class item, attribute-name: symbol,
    id: structure)
comment: text;
begin
    comment = call g2-get-change-log-entry-comment(item, attribute-name, id);
    post "[comment]"
end
```

This action button adds a comment to revision 1 of the change log entry for the label attribute of my-umc:

start post-change-log-entry-comment(my-umc, the symbol label, structure(revision: 1))

POST-CHANGE-LOG-ENTRY-COMMENT

Here is the resulting message board:

```
#96   3:28:12 p.m.   Changed label to do-
      action
```

Reverting Change Log Entries

Here is a generic procedure that reverts the change log entry for an attribute of an item with a given identifier:

```
revert-change-log-entry (item: class item, attribute-name: symbol, identifier: structure)
resulting-struct: structure;
begin
    resulting-struct = call g2-revert-change-log-entry(item, attribute-name, identifier);
    post "[resulting-struct]"
end
```

This action button reverts the label attribute of my-umc to revision 0:

Revert Attribute   start revert-change-log-entry(my-umc, the
                    symbol label, structure(revision: 0))

REVERT-CHANGE-LOG-ENTRY

Here is the resulting change log and message board, thereby adding a new entry to the change log:

Attribute	Revision	Value	Module Version	Timestamp	Author	Tags
Label	3	do-something	63	7 Feb 2007 3:34 p.m.	nrs	none
Applicable class	0	object	63	7 Feb 2007 3:09 p.m.	nrs	none
Action	0	post "Hello world"	63	7 Feb 2007 3:08 p.m.	nrs	none
Label	1	do-action	63	7 Feb 2007 3:08 p.m.	nrs	none
Label	0	do-something	63	7 Feb 2007 3:08 p.m.	nrs	none
Names	0	MY-UMC	63	7 Feb 2007 3:07 p.m.	nrs	none

```
#99   3:34:26 p.m.    structure
  (ATTRIBUTE: the symbol LABEL,
  REVISION: 0,
  COMMENT: "",
  TAGS: sequence (),
  TEXT-VALUE: "do-something",
  MODULE-VERSION: 63,
  TIMESTAMP: structure (YEAR: 2007,
    MONTH: 2,
    DATE: 7,
    HOURS: 15,
    MINUTES: 8),
  AUTHOR: the symbol NRS)
```

Tagging All Items in a Module

This action button tags the current version of all attributes of all items in the module named top with the symbol G283B0:

Tag Whole Module   start g2-tag-module(the symbol top,
                              structure(), the symbol G283R0)

Here is the resulting change log for my-umc:

Attribute	Revision	Value	Module Version	Timestamp	Author	Tags
Label	3	do-something	63	7 Feb 2007 3:34 p.m.	nrs	G283R0
Applicable class	0	object	63	7 Feb 2007 3:09 p.m.	nrs	G283R0
Action	0	post "Hello world"	63	7 Feb 2007 3:08 p.m.	nrs	G283R0
Label	1	do-action	63	7 Feb 2007 3:08 p.m.	nrs	none
Label	0	do-something	63	7 Feb 2007 3:08 p.m.	nrs	none
Names	0	MY-UMC	63	7 Feb 2007 3:07 p.m.	nrs	G283R0

Performing a "Diff" on Two Texts

This action button performs a "diff" on the change log for put-up-text-box-dialog:

Do Diff Test

start do-diff-test(put-up-text-box-dialog, this
  window)

This procedure calls spawn-diff, which performs a "diff", and show-results, which displays the results in a text box within a custom dialog:

```
do-diff-test (item-to-diff: class item, g2-win: class g2-window)
whole-diff-text: text;
begin
    whole-diff-text = call spawn-diff(item-to-diff);
    call show-results(whole-diff-text, g2-win)
end
```

This procedure calls g2-diff-texts on revision 0 and revision 1 of an item, and returns the diff-output of the return structure:

```
spawn-diff (item-to-diff: class item) = (text)
all-diffs: value;
user-name: value;
ndiffs: integer;
result: structure;
v0, v1, whole-diff-text: text;
begin
    all-diffs = the change-log of item-to-diff;
    ndiffs = the number of elements in all-diffs;

    user-name = call g2-name-for-item(item-to-diff);

    if ndiffs = 0 then
        post "[user-name] has no change-log"
    else if ndiffs = 1 then
        post "[user-name] has only one revision; cannot diff"
    else
        post "[user-name] has [ndiffs] revisions total";
    v0 = the text-value of all-diffs[0];
    v1 = the text-value of all-diffs[1];
    result = call g2-diff-texts(v1, v0);
    whole-diff-text = the diff-output of result;
    return whole-diff-text
end
```

Here is the result of doing the "diff" test on put-up-text-box-dialog:

Version Control Inspect Commands

You can use the following Inspect command to show the change log for items, tag change log entries, revert change log entries, delete change log entries, and enable/disable change logging for items.

show on a workspace the change log entry of the *attribute* of *item*
    {as of *timestamp*} | {with revision *num*} | {with tag *tag*}

Shows the list of change log entries for the given *attribute* of *item*, as of *timestamp*, with revision *num*, or tagged with *tag*.

show on a workspace the differences between
    the change log entry of the *attribute* of *item*
    {as of *timestamp*} | {with revision *num*} | {with tag *tag*}
    and the change log entry of the *attribute* of *item*
    {as of *timestamp*} | {with revision *num*} | {with tag *tag*}

Shows the differences between two change log entries for the *attribute* of *item*, as of *timestamp*, with revision *num*, or tagged with *tag*. This command uses the external "diff" program specified by g2-set-external-diff-specification.

use version control to tag the change log entry of every logged attribute of every
    item in module *module-name* [as of *timestamp*] using tag *tag*

Tags the change log entry of every item for which change logging is enabled in *module-name*, using *tag*, optionally as of *timestamp*.

use version control to tag the change log entry of the *attribute* of *item*
    [as of *timestamp* | with revision *num*] using tag *tag*

Tags the change log entry of the *attribute* of *item*, using *tag*, optionally as of *timestamp* or with revision *num*.

use version control to revert the text of every logged attribute of every item in
    module *module-name* to the change log entry
    {as of *timestamp*} | {using tag *tag*}

Reverts the text of every item for which change logging is enabled in *module-name* to the change log entry as of *timestamp* or tagged with *tag*.

use version control to revert the text of the *attribute* of *item* to the change log
    entry {as of *timestamp*} | {with revision *num*} | {with tag *tag*}

Reverts the text of the *attribute* of *item* to the change log entry as of *timestamp*, with revision *num*, or tagged with *tag*.

use version control to delete the change log entry of the *attribute* of *item*
    {as of *timestamp*} | {with revision *num*} | {with tag *tag*}

Deletes the change log entry for the *attribute* of *item*, as of *timestamp*, with revision *num*, or tagged with *tag*.

use version control to enable change logging on *item*

> Enables change logging on *item*. If change logging is already disabled on the module, this command has no effect.

use version control to disable change logging on *item*

> Disables change logging on *item*. If change logging is already disabled on the module, this command has no effect.

Locking Mechanism for Objects

G2 provides two techniques for invoking a method:

- Using the call procedure statement, which executes the method synchronously. It tells G2 to invoke the method and wait until it returns before continuing to execute the calling procedure or method.

- Using the start action, which executes the method and runs it asynchronously. It tells G2 to schedule the method for execution, then continue processing the calling procedure or method. The method being started does not run until the entity that issued the start action either completes or enters a wait state.

To enter a wait state and allow other processing to occur, a procedure must execute one of the following statements: allow other processing, call ... across, collect data, for each ... do in parallel, or wait.

Historically, to avoid problems due to concurrency when a procedure or method enters a wait state, it has been up to the developer to ensure that only one procedure is accessing the same object at the same time.

G2 Version 8.3 Rev. 0 provides a locking mechanism, which allows other processing to occur when executing a method with a wait state, while ensuring that no more than one procedure that locks the same object can operate at the same time. This feature is similar to the use of the synchronized keyword in Java.

To support this feature, methods define the synchronized attribute, whose value is one of the following:

- no — When an unsynchronized method executes, the item that is the first argument to the method call is not locked and is, therefore, vulnerable to concurrency problems if the method contains a wait state. This is the default.

- yes — When a synchronized method executes, the item that is the first argument to the method call is locked, which prevents any other synchronized method from obtaining a lock on the item. When the synchronized method completes, either normally or due to an error, the lock is released.

When a synchronized method attempts to execute on a locked item, there are two possible outcomes, depending on how the method is executed, as follows:

When a synchronized method is executed in...	The synchronized method...
A different call chain	Waits to execute until the lock is released.
	If the synchronized method is invoked via a call statement, then the calling method enters a wait state until the lock is released, allowing other processing to occur while waiting for the lock to be released.
The same call chain	Executes normally.

A synchronized method is executed in a different call chain when:

- Using a call statement in a different procedure or method.

- Using the start action.

- Within a do in parallel, do in parallel until one completes, or for each...do in parallel statement.

A synchronized method is executed in the same call chain when it is executed via a call statement within the same method or within any method in the calling hierarchy of the synchronized method. Note that the call statement can be inside an on error block and is still considered within the same call chain. However, if the call statement is within a do in parallel, do in parallel until one completes, or for each ... do in parallel statement, this is considered a different call chain.

It is possible, due to careless use of synchronization, to have a program in which several methods are permanently in wait states, each waiting for the other to release a lock, in what is usually referred to as a *deadlock*. You can avoid this situation by ensuring that a call chain that obtains locks on multiple items always attempts to lock the items in the same order.

G2 provides a way of detecting and releasing deadlocks when they occur, interactively, programmatically, and automatically.

As a result of this change, the call action now causes procedures to enter a wait state when calling a synchronized method and when the first argument to the call is an item that is currently locked by another synchronized method.

The locking mechanism for G2 methods works in the same way as synchronized methods in Java. For more information, see `http://java.sun.com/docs/books/jls/third_edition/html/classes.html#8.4.3.6` and `http://java.sun.com/docs/books/jls/third_edition/html/memory`.

Example: Calling a Synchronized Method from a Procedure

This example demonstrates executing a synchronized method via a call statement in two separate procedure invocations when the method contains a wait state. When the method executes in the first invocation of the procedure, the item that is the first argument to the method is locked. Any other synchronized method call on the same item is not allowed to execute until the first method completes, even though the method contains a wait state, because the method is being executed in a different procedure invocation.

In this example, my-method-caller calls my-method on items of class lock. The action button starts my-method-caller for lock-one in two separate procedure invocations.

MY-METHOD-CALLER

MY-METHOD

Synchronized no

LOCK::MY-METHOD

LOCK   LOCK-ONE

start my-method-caller(lock-one, "invocation one") and start my-method-caller(lock-one, "invocation two")

My-method-caller procedure calls my-method. When synchronized = yes, the first call to my-method locks lock-one. The second invocation of my-method is not allowed to execute until lock-one is unlocked, because my-method is executed in a different procedure invocation.

When synchronized = no, my-method is allowed to execute in two separate procedure invocations because lock-one is not locked. This can lead to concurrency issues.

Here is the my-method synchronized method, which has synchronized set to yes. This method simply posts to the Message Board that my-method is starting, waits, then posts to the Message Board that my-method is completed. Because the method contains a wait state, it allows other processing to occur; however, because the method is synchronized, the item that is the first argument to the method is locked. Thus, any other synchronized method that attempts to execute on the same item must wait until the first method completes and releases the lock before it can execute.

```
my-method(thing-to-lock: class lock, message: text)
begin
    post "Starting my-method on [the name of thing-to-lock]: [message]";
    wait for 5 seconds;
    post "Completed my-method on [the name of thing-to-lock]: [message]";
end
```

Here is the procedure that executes my-method, using a call statement. It posts to the Message Board that it is about to call my-method, calls my-method, then posts to the Message Board that my-method is complete.

```
my-method-caller(thing-to-lock: class lock, message: text)
begin
    post "About to call my-method on [the name of thing-to-lock]: [message]";
    call my-method(thing-to-lock, message);
    post "Returned from my-method on [the name of thing-to-lock]: [message]";
end
```

The action button starts the my-method-caller procedure twice, which creates two separate procedure invocations, each using the same target object, lock-one:

```
start my-method-caller(lock-one, "invocation one") and
start my-method-caller(lock-one, "invocation two")
```

Using method synchronization (synchronized is yes) and a wait state, the synchronized method allows other processing to occur due to the wait state; however, it cannot execute the method on the same item because the item is locked. In this scenario, there are no concurrency issues.

Synchronized, wait state: No concurrency issues

Message Board 29 Aug 2006

#60 10:20:31 a.m. About to call
my-method on LOCK-ONE: invocation
one

> Procedure statements that appear
> before the wait state execute in
> invocation one of my-method.

#61 10:20:31 a.m. Starting
my-method on LOCK-ONE: invocation
one

> The first argument to the method,
> lock-one, is locked.

#62 10:20:31 a.m. About to call
my-method on LOCK-ONE: invocation
two

> The procedure my-method-caller is
> allowed to execute due to the wait state.

> However, my-method cannot execute on
> lock-one until the item is unlocked.

#63 10:20:36 a.m. Completed
my-method on LOCK-ONE: invocation
one

> Procedure statements that appear after
> the wait state execute in invocation one
> of my-method.

#64 10:20:36 a.m. Returned from
my-method on LOCK-ONE: invocation
one

> Lock-one is now unlocked, making it
> available for processing by other
> methods on the same item.

#65 10:20:36 a.m. Starting
my-method on LOCK-ONE: invocation
two

#66 10:20:41 a.m. Completed
my-method on LOCK-ONE: invocation
two

> Procedure statements in my-method
> can now execute on lock-one in
> invocation two of my-method.

#67 10:20:41 a.m. Returned from
my-method on LOCK-ONE: invocation
two

With no method synchronization (synchronized is no) and a wait state, the synchronized method allows other processing to occur due to the wait state, including executing the method on the same item in a different invocation of the method. This is the scenario that can lead to concurrency issues because the item is not locked.

Unsynchronized, wait state: Concurrency issues

Message Board 29 Aug 2006

#68 10:31:05 a.m.   About to call
my-method on LOCK-ONE: invocation
one

#69 10:31:05 a.m.   Starting
my-method on LOCK-ONE: invocation
one

Procedure statements that appear
before the wait state execute in
invocation one of my-method.

#70 10:31:05 a.m.   About to call
my-method on LOCK-ONE: invocation
two

#71 10:31:05 a.m.   Starting
my-method on LOCK-ONE: invocation
two

Procedure statements that appear
before the wait state execute in
invocation two of my-method.

#72 10:31:10 a.m.   Completed
my-method on LOCK-ONE: invocation
one

#73 10:31:10 a.m.   Returned from
my-method on LOCK-ONE: invocation
one

Procedure statements that appear
after the wait state execute in
invocation one of my-method.

#74 10:31:10 a.m.   Completed
my-method on LOCK-ONE: invocation
two

#75 10:31:10 a.m.   Returned from
my-method on LOCK-ONE: invocation
two

Procedure statements that appear
after the wait state execute in
invocation two of my-method.

Example: Calling a Synchronized Method from the Same Method

This example demonstrates executing two synchronized methods on the same
item when one method uses the call statement to execute the other method and
both methods contain a wait state. When the outer method executes, the item that
is the first argument to the method is locked. However, because the inner method
is being called from the same call chain, it is allowed to execute, even though the
item is locked.

In this example, outer calls inner on items of class lock. The action button starts outer for lock-one, which starts inner for lock-one:

Synchronized yes

ITEM::OUTER

Synchronized yes

ITEM::INNER

OUTER

INNER

start outer(lock-one, lock-one)

The outer method calls the inner method. When synchronized = yes, the outer method locks lock-one. The inner method is allowed to execute because it is within the same call chain.
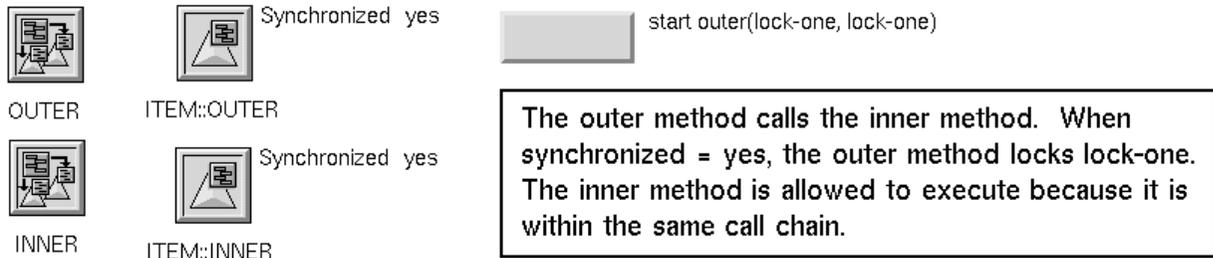
Here is the outer synchronized method, which has synchronized set to yes. This method posts to the Message Board that outer is starting, waits, calls inner, waits, then posts to the Message Board that outer is completed. The method contains a wait state, which allows other processing to occur. Because the method is synchronized, the item that is the first argument to the method is locked when outer executes. However, because outer executes inner by using a call statement in the same synchronized method, inner is allowed to execute because it is within the same call chain.

```
outer(lock-for-outer: class item, lock-for-inner: class item)
begin
    post "Starting OUTER method on [the name of lock-for-outer]";
    wait for 5 seconds;
    call inner(lock-for-inner);
    wait for 5 seconds;
    post "Returning from OUTER method on [the name of lock-for-outer]";
end
```

Here is the inner synchronized method, which also has synchronized set to yes. This method simply posts to the Message Board that inner is starting, waits, then posts to the Message Board that inner is complete.

```
inner(lock: class item)
begin
    post "Starting INNER method on [the name of lock]";
    wait for 5 seconds;
    post "Returning from INNER method on [the name of lock]";
end
```

The action button starts outer using lock-one as the argument to both outer and inner:

```
start outer(lock-one, lock-one)
```

Using method synchronization (synchronized is yes) and a wait state, the outer synchronized method allows other processing to occur due to the wait state. Because the outer method calls the inner method, the inner method is allowed to execute, even though the item is locked, because it is in the same call chain. In this

scenario, there are no concurrency issues because the methods execute within the same call chain.

Calling a synchronized method in the same call chain:
No concurrency issues
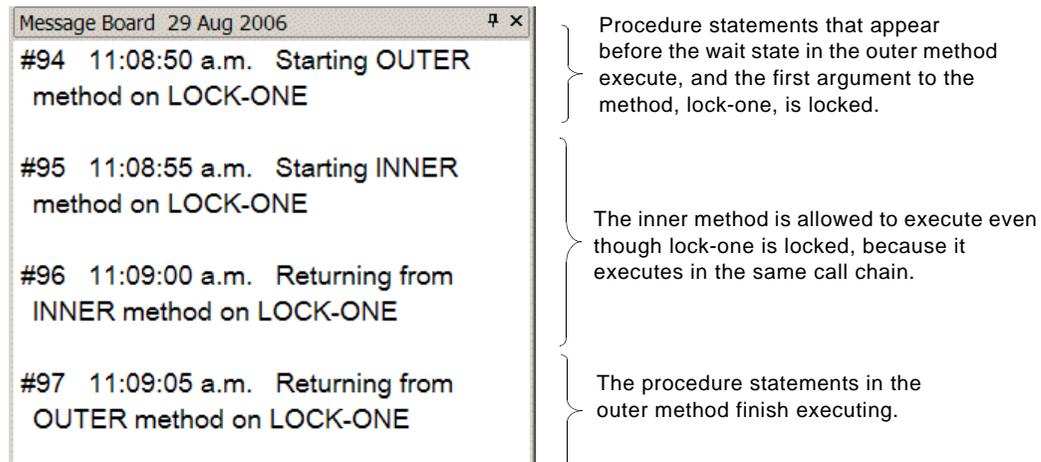
Message Board 29 Aug 2006

#94 11:08:50 a.m. Starting OUTER method on LOCK-ONE

Procedure statements that appear before the wait state in the outer method execute, and the first argument to the method, lock-one, is locked.

#95 11:08:55 a.m. Starting INNER method on LOCK-ONE

#96 11:09:00 a.m. Returning from INNER method on LOCK-ONE

The inner method is allowed to execute even though lock-one is locked, because it executes in the same call chain.

#97 11:09:05 a.m. Returning from OUTER method on LOCK-ONE

The procedure statements in the outer method finish executing.

Detecting and Releasing Deadlocks

You have two options for detecting deadlocks:

- Choose Miscellany > Detect Deadlocks.

- Use this system procedure:

      g2-detect-deadlocks
          ( )
          −> _return-value_: truth-value

These options simply indicate whether a deadlock exist.

You can also detect and break deadlocks by using one of these options:

- Choose Miscellany > Detect and Break Deadlocks.

- Use this system procedure:

      g2-detect-and-break-deadlocks
          ( )
          −> _return-value_: truth-value

   Returns false if no deadlock is detected; otherwise, returns true and breaks the deadlock.

- Setting the Automatic Deadlock Detection Frequency parameter in the Miscellaneous parameters system table to the frequency, in seconds, with which to check for deadlocks and break them when found.

These options detect deadlocks and abort one of the involved methods by generating an instance of g2-deadlock-error, a subclass of g2-error. G2 chooses which method to abort, as follows:

- It chooses a method that contains an on error clause that catches errors of type g2-deadlock-error or one of its superior classes, like g2-error, or a method that was called by a procedure or method that contains such an on error clause, and so forth.

- Otherwise, it arbitrarily chooses one of the methods that is participating in a deadlock. Note that in the absence of an on error clause, all methods in the call chain will be aborted as a result when a deadlock exists.

Note that it always chooses a method that is waiting for a lock to be released, rather than a method or procedure that has called a method or procedure and is waiting for that method to return.

Example: Detecting and Releasing Deadlocks Using an Error Handler

This example shows a simple deadlock in which two synchronized methods are waiting for locks to be released on the same locked objects. The button below starts outer-with-error-handler locking lock-one and passing lock-two as the argument to inner, then it starts outer locking lock-two and passing lock-one as the argument to inner. The result is a deadlock, because each procedure is waiting for the other to complete before it can release the lock on the respective locked objects.

start outer-with-error-handler(lock-one, lock-two) and start outer(lock-two, lock-one)

ITEM::OUTER-WITH-ERROR-HANDLER

ITEM::OUTER

ITEM::INNER

Here is the outer-with-error-handler method:

```
outer-with-error-handler (lock: class item, lock-for-inner: class item)
errobj: class error;
begin
    post "Starting OUTER-WITH-ERROR-HANDLER method on [the name of lock]";
    wait for 5 seconds;
    begin
        call inner(lock-for-inner);
    end
    on error (errobj)
        post "An error of class [the class of errobj] occurred: [the text of the
            error-description of errobj]";
        delete errobj;
    end;
    post "Returning from OUTER-WITH-ERROR-HANDLER method on [the name of
        lock]";
end
```

Here is the outer method:

```
outer (lock: class item, lock-for-inner: class item)
begin
    post "Starting OUTER method on [the name of lock]";
    wait for 5 seconds;
    call inner(lock-for-inner);
    post "Returning from OUTER method on [the name of lock]:";
end
```

Here is the inner method:

```
inner(lock: class item)
begin
    post "Starting INNER method on [the name of lock]";
    wait for 5 seconds;
    post "Returning from INNER method on [the name of lock]";
end
```

When you click the button that starts outer-with-error-handler, the outer-with-error-handler method starts on lock-one, then the outer method starts on lock-two. Neither method returns due to the deadlock whereby the first call to inner is waiting for lock-two, which is locked by outer, and the second call to inner is

waiting for **lock-one**, which is locked by **outer-with-error-handler** and cannot complete.
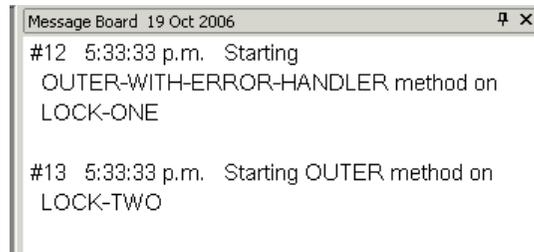
Message Board 19 Oct 2006

#12 5:33:33 p.m. Starting
 OUTER-WITH-ERROR-HANDLER method on
 LOCK-ONE

#13 5:33:33 p.m. Starting OUTER method on
 LOCK-TWO

The following button and procedure detect deadlocks:

start detect-deadlocks()

DETECT-DEADLOCKS

Here is the **detect-deadlocks** procedure:

```
detect-deadlocks()
result: truth-value;
begin
    result = call g2-detect-deadlocks();
    if (result) then
        post "Deadlock detected"
    else
        post "No Deadlock detected"
end
```

Clicking the button that detects deadlocks displays this new message in the Message Board:
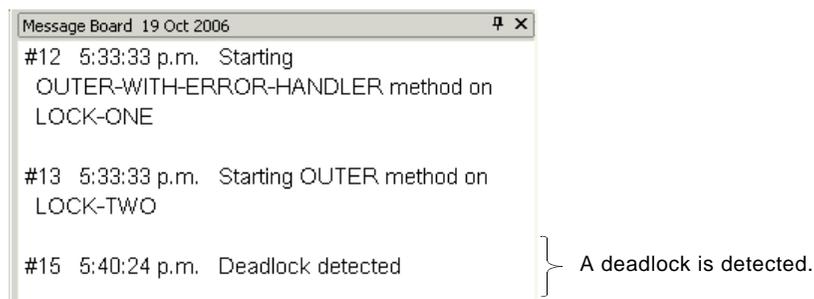
Message Board 19 Oct 2006

#12 5:33:33 p.m. Starting
 OUTER-WITH-ERROR-HANDLER method on
 LOCK-ONE

#13 5:33:33 p.m. Starting OUTER method on
 LOCK-TWO

#15 5:40:24 p.m. Deadlock detected   A deadlock is detected.

Alternatively, choosing Miscellany > Detect Deadlocks displays the following message in the G2 Operator Logbook:

```
#14   5:38:23 p.m.   Deadlock detected: Use
 detect-and-break-deadlocks to abort deadlocked
 procedures
```

The following button and procedure detect and break deadlocks:

start detect-and-break-deadlocks()

DETECT-AND-BREAK-DEADLOCKS

Here is the detect-and-break-deadlocks procedure:

```
detect-and-break-deadlocks()
result: truth-value;
begin
    result = call g2-detect-and-break-deadlocks();
    if not (result) then
        post "No deadlock detected"
end
```

Clicking the button that detects and breaks deadlocks displays these new messages in the Message Board. First, the outer-with-error-handler method is aborted and a g2-deadlock-error occurs and is posted to the Message Board. Next, the outer-with-error-handler method returns, which releases the lock on lock-one.

The call to outer is allowed to proceed by executing inner on lock-one and returning.

A g2-deadlock-error occurs, and the outer-with-error-handler method is aborted to break the deadlock.

The outer-with-error-handler method returns after it is aborted, which releases the lock on lock-one.

The inner method is allowed to execute on lock-one, which allows the outer method to complete.

Example: Detecting and Releasing Deadlocks with No Error Handler

This example shows what happens when a deadlock occurs on a method that has no error handler. In this case, the button below starts outer locking lock-one and passing lock-two as the argument to inner, then it starts outer locking lock-two and passing lock-one as the argument to inner. Again, the result is a deadlock, because each procedure is waiting for the other to complete before it can release the lock on the respective locked objects.

ITEM::OUTER
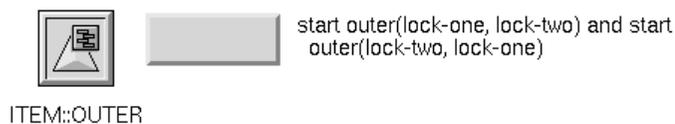
In this case, clicking the button that starts outer, then clicking the button that detects deadlocks, then clicking the button that detects and breaks deadlocks results in these messages in the Message Board:
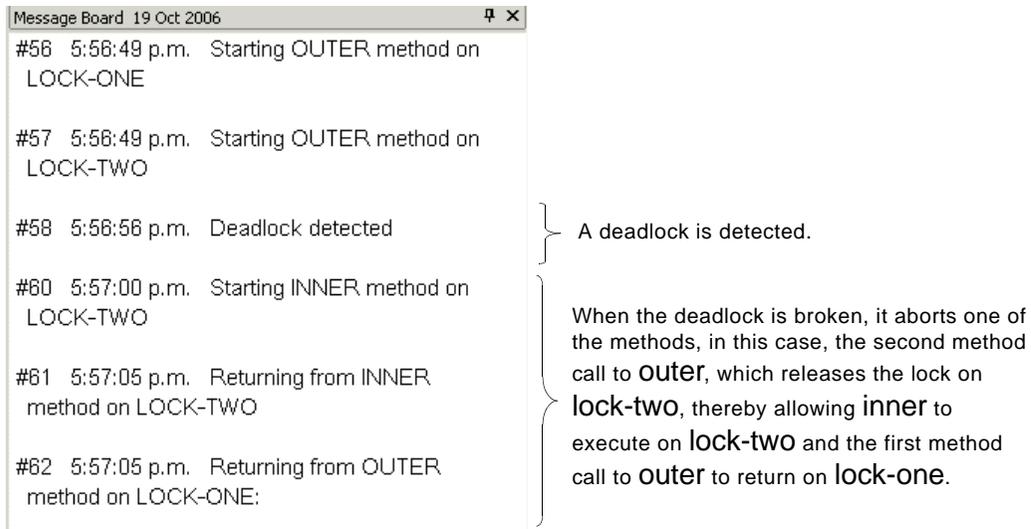
```
Message Board  19 Oct 2006                    ⏻ ✕

#56  5:56:49 p.m.   Starting OUTER method on
 LOCK-ONE

#57  5:56:49 p.m.   Starting OUTER method on
 LOCK-TWO

#58  5:56:56 p.m.   Deadlock detected

#60  5:57:00 p.m.   Starting INNER method on
 LOCK-TWO

#61  5:57:05 p.m.   Returning from INNER
 method on LOCK-TWO

#62  5:57:05 p.m.   Returning from OUTER
 method on LOCK-ONE:
```

A deadlock is detected.

When the deadlock is broken, it aborts one of the methods, in this case, the second method call to outer, which releases the lock on lock-two, thereby allowing inner to execute on lock-two and the first method call to outer to return on lock-one.

Because neither method defined an error handler, the g2-deadlock-error appears in the G2 Operator Logbook, as follows:

```
#59  5:57:00 p.m.  Error:
 (G2-DEADLOCK-ERROR)

Procedure aborted to break a deadlock


Activity: system call statement
Within: ITEM::OUTER(LOCK-TWO, LOCK-ONE)
Local Names:
 LOCK: class item = LOCK-TWO;
 LOCK-FOR-INNER: class item = LOCK-ONE
Aborting procedure stack from
 ITEM::OUTER(LOCK-TWO, LOCK-ONE).
```

Summing Values in Histories

G2 supports the following new grammar to sum values in histories:

the sum of the values of *quantitative-var-or-param*
    during the last *time-expression*

the sum of the values of *quantitative-var-or-param* between
    time-expression ago and *time-expression* ago

For example, this procedure sums the values of the parameter named param during the last 1 hour:

```
sum-of-values()
q1: float;
begin
    q1 = the sum of the values of param during the last 1 hour;
post "[q1]";
end
```

This procedure sums the values of param between 30 and 10 seconds ago:

```
sum-of-values-over-interval()
q1: float;
begin
    q1 = the sum of the values of param between 30 seconds ago and 10 seconds
ago;
post "[q1]";
end
```

Referencing a Time Interval Ending with the Collection Time

G2 supports the following grammar for referencing a time interval ending with the collection time:

history-expression of *variable-or-parameter* during the *time-expression* ending with the collection time

where:

- *history-expression* is one of the following:

  ```
  the value of
  the number of history datapoints in
  the average value of
  the integral in {seconds | minutes | hours | days | weeks} of
  the interpolated value of
  the {maximum | minimum} value of
  the rate of change per {seconds | minutes | hours | days | weeks} of
  the standard deviation of
  the sum of the values of
  ```

- *variable-or-parameter* is a quantitative variable or parameter, except when using the number of history datapoints in, in which case it can be any type of variable or parameter.

For example, the following procedure computes the average value of a parameter during the 30 minutes ending with the collection time:

```
compute-average()
q1: float;
begin
    q1 = the average value of param during the 30 minutes ending with
        the collection time;
    post "[q1]";
end
```

Referring to the Current System Time

G2 supports the following new time expressions:

- the current system time — Returns the current UNIX time, as a float, which is the number of seconds since January 1, 1970, using the G2 clock.

- the current system real time — Returns the current UNIX time, as a float, which is the number of seconds since January 1, 1970, using the real-time clock.

Confirming Run State Changes

The Miscellaneous Parameters system table defines a new attribute named confirm-run-state-changes with values yes or no. When set to yes, G2 posts a confirmation dialog for any attempt to start, restart, reset, resume, or pause G2. The dialog is posted on the window where the request was made. The default is no.

Error Handling

Error objects have a read-only hidden attribute called error-source-trace, which provides a history of error descriptions and source information, and is updated each time the error is signaled. The error-source-trace attribute has this syntax:

```
sequence
(structure
    (error-trace-description: text,
    error-trace-source-item: symbol,
    error-trace-source-line: integer,
    error-trace-source-column: integer)
. . .)
```

Each structure uses the current values from the error at the time it was signaled, where the first entry corresponds with the most recent error that was signalled.

System Procedures

sys-mod.kb

The User Interface Operations workspace of sys-mod.kb has various changes:

- Native Menu API has been moved from the top-level workspace to the User Interface Operations workspace.

- The Dialog API workspace has been renamed Dialog Views and now contains only system procedures related to dialogs.

- There are now separate workspaces for Chart Views, HTML Views, Property Grid Views, Shortcut Bar Views, Tree Views, and Window Handles and Views, all of which used to be under Dialog API.

Clearing Histories

You can clear the history and set the initial value and collection time of a parameter by using this system procedure:

g2-initialize-parameter
   (*parameter*: class parameter, *value*: value, *collect-time*: quantity)

   Initializes *parameter* to have only one historical value. The value, initial-value, and last-recorded-value are all set to the specified *value* and *collection-time*.

Getting Highly Precise Timestamps

Use the following system procedures to get performance statistics for executing tasks in G2. On Windows, these system procedures return highly precise timestamps that indicate the number of "ticks" that the Microsoft high-resolution timer takes to execute a task.

The g2-get-performance-counter and g2-get-and-log-performance-counter system procedures perform logging, as follows. All output goes to the standard G2 output. On Windows, standard output is usually a log file, but it could be a console window if G2 is run with the -no-log command-line option. On UNIX, standard output is usually the shell where G2 was started, unless G2's output is redirected into a file. If tracing-and-breakpoints is enabled, the output is also logged to that file.

g2-get-performance-frequency
   ( )
   −> *ticks*: integer

   Returns the number of "ticks" per second on the machine as an integer. This number is useful only for interpreting the results of the following two system procedures.

g2-get-performance-counter
( )
−> *ticks*: float

Returns the current number of "ticks" as a float and logs the output.

g2-get-and-log-performance-counter
(*msg*: text)
−> *ticks*: float

Returns the current number of "ticks" as a float and logs the output. The message is formatted on two lines, as follows:

High Precision Counter: *counter-as-64-bit-int*
msg

Pinging a Network Host

g2-ping
(*hostname-or-ip*: text, *options*: structure)
−> *info*: structure

Pings a remote host or IP address and returns an *info* structure. The *options* structure has this syntax:

structure
(packet-size: *integer*,
timeout: *quantity*,
pause-between-ping: *quantity*,
number-of-pings: *integer*,
ttl: *integer*,
progress-procedure: *symbol*,
progress-procedure-user-data: *item-or-value*)

where:

- packet-size is the size of the packet for the ping request. The default is 32 bytes.

- timeout is the minimum timeout period rounded based on the minimum scheduling interval. The default is 10 seconds.

- pause-between-ping is the minimum time, in seconds, for pausing between ping attempts, rounded based on the minimum scheduling interval. The default is 0.

- number-of-pings is the number of ping attempts. The default is 1.

- ttl is the maximum number of routers through which this packet should travel, which is an integer between 0 and 255. Each time an IP packet goes through a router, its ttl value is decremented by 1.

- **progress-procedure** is a symbol naming a callback procedure, which is called after each ping is received, where the event is the symbol ping-reply, or after a timeout occurs.

- **progress-procedure-user-data** is passed as the *data* argument to the progress procedure.

The signature of the progress procedure is:

```
my-ping-progress-callback
    (event: symbol, data: value, user-data: item-or-value)
```

When the *event* is ping-reply, the *data* argument contains a structure with the same syntax as the <u>*info*</u> structure return value to g2-ping.

The <u>*info*</u> structure has this syntax:

```
structure
(hostname-or-ip: text,
packets-transmitted: integer,
packets-received: integer,
packets-lost: integer,
packet-size: integer,
minimum-round-trip-in-s: quantity,
maximum-round-trip-in-s: quantity,
average-round-trip-in-s: quantity,
reply-type: symbol, [none | icmp_echoreply | icmp_timxceed]
last-reply-type: symbol, [none | icmp_echoreply | icmp_timxceed]
last-hostname-or-ip: text,
packets: sequence)
```

where packets is a sequence of structures, one for each ping request, where each structure has this syntax:

```
structure
(hostname-or-ip: text,
reply-type: symbol,
round-trip-in-s: quantity,
ttl: integer)
```

Tracing a Network Host

g2-trace-route
   (*hostname-or-ip*: text, *options*: structure)
   −> *info*: structure

Detects the reachability of a remote host or IP address and number of intermediate hops, and returns an *info* structure. The *options* structure has this syntax:

    structure
    (maximum-hops: *integer*,
    packet-size: *integer*,
    timeout: *integer*,
    number-of-pings: *integer*,
    progress-procedure: *symbol*,
    progress-procedure-user-data: *item-or-value*)

where:

- maximum-hops is the maximum number of hops to detect. The default is 30.

- packet-size is the size of the packet for the request. The default is 32 bytes.

- timeout is the time before the procedure times out. The default is 10 seconds.

- number-of-pings is the number of ping attempts. The default is 1.

- progress-procedure is a symbol naming a callback procedure, which is called at the end of each hop, where the event is the symbol trace-route-hop.

- progress-procedure-user-data is passed as the *data* argument to the progress callback procedure.

The signature of the progress procedure is:

    my-trace-route-progress-callback
        (*event*: symbol, *data*: value, *user-data*: item-or-value)

When the *event* is trace-route-hop, the *data* argument contains a structure with the same syntax as the *info* structure return value to g2-trace-route.

The *info* structure has this syntax:

    structure
    (hostname-or-ip *text*,
    maximum-hops: *integer*,
    hops: *sequence*)

where hops is a sequence of structures, one for each hop, where each structure has this syntax:

```
structure
(hop: integer,
hostname-or-ip: text,
reply-type: symbol, [none | icmp_echoreply | icmp_timxceed]
packets-transmitted: integer,
packets-received: integer,
packets-lost: integer,
packet-size: integer,
minimum-round-trip-in-s: quantity,
maximum-round-trip-in-s: quantity,
average-round-trip-in-s: quantity)
```

Getting the IP Address of the G2 Server

You can use the following system procedure, which is located on the Network Information Procedures workspace of sys-mod.kb, to get the IP address of the G2 server machine:

g2-get-network-address-list
    ( )
    −> *ip-addresses*: sequence

Returns a sequence of strings representing the dotted octet notation of all relevant internet addresses of the G2 server machine, both external and internal, for example:

    sequence("192.168.0.2","66.203.92.21")

Getting the G2 Process ID

Use the following system procedure to get the process ID of the current G2 process, as a float:

g2-get-g2-process-identifier
    ( )
    −> *pid*: float

Using the Explanation Facilities Programmatically

The explanation facility allows you to trace rule invocations, as well as variable and parameter updates. In addition to getting a visual representation of an explanation tree on the subworkspace of an explanation, you can now get a representation as a structure as well. The system procedure is located on the Get Hierarchy workspace of sys-mod.kb.

For details about the Explanation Facilities, see Chapter 52, "Explanation Facilities" in the *G2 Reference Manual*.

g2-get-explanation-hierarchy
    (*var-or-param*: variable-or-parameter)
    −> *explanation-tree*: structure

Returns a structure describing the hierarchy of rule invocations for a variable or parameter, as follows:

    structure
    (node-type: *symbol*,
    item-or-value: *item-or-value*,
    node-specific-data: *sequence*,
    children: *sequence*)

where:

- node-type is one of the symbols: item, rule, variable-or-parameter, specific-formula, or data-server-or-initial-value.

- item-or-value depends on the node-type, as follows:

    – When node-type is item, rule, or variable-or-parameter, the item.

    – When node-type is specific-formula, the text of the specific formula.

    – When node-type is data-server-or-initial-value, the text "External Data Server" or "Initial Value", as appropriate.

- node-specific-data depends on the node-type, as follows:

    – When node-type is variable-or-parameter, a value-structure as described in variable-or-parameter in Chapter 2, "Class Dictionary" in the *G2 Class Reference Manual*.

    – When node-type is rule, a sequence of structures that describe the bindings of the local variables in the rule, where each structure has this syntax:

    structure
    (local-name: *text*,
    item-or-value: *item-or-value*)

    – For all other values of node-type, node-specific-data is an empty sequence.

- children is a sequence of structures that describe the source of the data for the explanation node, where each structure has the same syntax as the *explanation-tree* return value.

    If a circularity is detected, for example, a rule both gets triggered by a variable and also concludes a value to that variable, the variable-or-

parameter structure appears as one of the rule's children, with no children of its own, to prevent an infinitely deep structure.

Generating a List of System-Defined Attributes

If *g2-attribute* is the name of a reserved attribute of a system-defined class, then you cannot use it as the name of an attribute of a user-defined class.

Some system-defined classes such as object or connection are user-extensible; other system-defined classes such as logbook-parameters are not user-extensible. To avoid possible inheritance problems, you cannot use system-defined attributes of user-extensible system-defined classes as user-defined attributes; thus, these attributes are considered reserved words in G2. However, you can use system-defined attributes of non user-extensible system-defined classes as user-defined attributes; these attributes are considered unreserved.

If you attempt to use a reserved word as a user-defined attribute, G2 takes the following actions:

- When entering a reserved word in the G2 Text Editor, an error such as the following appears in the text editor:

  > This is uncompilable. HEIGHT-OF-IMAGE is the name of a G2 system attribute and cannot be a user-defined attribute."

- When loading a KB from an older version of G2 in which the reserved word was not a system-defined attribute, an error such as the following appears in the Operator Logbook:

  > HEIGHT-OF-IMAGE is the name of a G2 system attribute and cannot be a user-defined attribute.

  Also, the notes of the user-defined class-definition contains an error such as the following:

  > OK, and note that the class-specific-attribute height-of-image is now a reserved G2 attribute. You must rename it before starting G2.

To obtain the name of the user-defined class that uses the reserved word, use the following Inspect command:

> highlight the symbol height-of-image in every class-definition

You can use the following system procedure, which is located on the Attribute Information workspace of sys-mod.kb, to get a list of all reserved words:

g2-get-all-reserved-system-attribute-names
    (*type*: symbol)
    –> *reserved-words*: sequence

Returns a sequence of all reserved system-defined class attribute names, in alphabetical order, where *type* is one of these symbols:

- ordinary — Returns all non-hidden attributes.

- hidden — Returns all hidden attributes of user-extensible classes.

- all — Returns both ordinary and hidden attributes.

If *g2-hidden-attribute* is the name of a hidden attribute of a system-defined class, you may use it as the name of an attribute of a user-defined class. However, we recommend that you avoid this practice. For example, using history as the name of an attribute of a user-defined class would shadow its use as a hidden attribute of a float-parameter. Similarly, using containing-module as the name of an attribute of a user-defined class would shadow its use in GFR and GMS.

Note that you cannot use reserved symbols as the name of a user-defined attribute. For a list of reserved symbols, see Appendix B, "Reserved Symbols" in the *G2 Reference Manual*.

Here is the list of reserved ordinary system-defined attributes in G2 Version 8.3 Rev. 0:

action	background-color	categories
action-priority	background-images	change
allow-duplicate-elements	blank-for-type-in?	change-log
annotations		characters-procedure
applicable-class		chart-style
array-is-permanent		class
array-length		class-inheritance-path
attribute-displays		class-name
attribute-initializations		class-of-object-computed-by-this
authors		class-of-procedure-invocation
		class-specific-attributes
		comment-procedure
		condition
		connection-arrows
		connector-formats
		cross-section-pattern

data-series	element-type	fatal-error-procedure
data-server	end-document-procedure	file-name-of-image
data-server-for-messages	end-element-procedure	file-status
data-type	error-description	file-system
default-cell-format	error-procedure	first-class
default-evaluation-setting	error-source-column	focal-classes
default-message-properties	error-source-item	focal-objects
default-procedure-priority	error-source-line	foreground-color
default-simulation-time-increment	expression-to-display	format-for-type-in-box
default-update-interval	external-simulator-configuration	format-of-image
depth-first-backward-chaining-precedence	external-system-has-a-scheduler	formula
depth-of-image		frame-style
description-of-frame		
dialog-height		
dialog-title		
dialog-type		
dialog-width		
dialog-x-position		
dialog-y-position		
direct-superior-classes		
disable-interleaving-of-large-messages		
dismissed-callback		
display-format		
display-update-interval		
display-update-priority		
display-wait-interval		

g2-connection-status	have-edit-option-buttons-for-type-in?	icon-description
g2-meter-name	height-of-image	icp-connection-specification
g2-routing-information	history-keeping-spec	identifying-attributes
g2-to-g2-interface-name		ignore-gfi-input-base-time
g2-user-mode		include-in-menus
g2-user-name		increment-per-dial-ruling
g2-window-height		increment-per-meter-ruling
g2-window-initial-window-configuration-string		inherited-attributes
		initial-value
g2-window-management-type		initial-value-for-simulation
g2-window-mode-is-valid		initial-values
g2-window-operating-system-type		initializable-system-attributes
g2-window-remote-host-name		initialization-vector-for-external-simulator
g2-window-reroute-problem-report		
g2-window-specific-language		input-vector-to-external-simulator
g2-window-style		instance-configuration
g2-window-time-of-last-connection		instantiate
g2-window-user-is-valid		integration-algorithm
g2-window-user-name-in-operating-system		interface-initialization-timeout-period
g2-window-width		interface-status
g2-window-x		interface-timeout-period
g2-window-x-resolution		interface-warning-message-level
g2-window-y		interpolate
g2-window-y-resolution		interval-to-poll-external-system
gfi-input-file-format		inverse-of-relation
gfi-input-file-pathname		item-configuration
gfi-input-interface-object		items-belonging-to-this-model
gfi-input-time-stamp-format		
gfi-input-variables-update-mode		
gfi-input-when-active		
gfi-output-file-format		
gfi-output-file-pathname		
gfi-output-file-update-frequency		
gfi-output-time-stamp-format		
gfi-output-values		
gfi-output-when-active		
grouping-specification		
gsi-application-name		
gsi-connection-configuration		
gsi-interface-name		
gsi-interface-status		
gsi-variable-status		

junction-block	keep-sorted	label label-to-display last-recorded-value line-pattern list-is-permanent low-value-for-dial-ruling low-value-for-meter-ruling
maximum-number-of-output-lines maximum-value message-keywords minimum-value model-simulation-status module-assignment	name-in-foreign-image name-in-remote-system names native-window-height native-window-left native-window-state native-window-top native-window-width notes number-of-pending-callbacks	off-value on-value options output-vector-from-external-simulator
patterns-definition plots point-formats poll-external-system-for-data prefer-buffered-drawing proprietary-package	qualified-name	readout-table-display-value relation-is-permanent relation-is-symmetric relation-name relation-summary remote-g2-expression remote-process-initialization-string renamed-gfi-output-file-pathname requires-call-next-method? rule-priority

save-image-data-with-kb scan-interval second-class send-all-values-at-beginning-of-cycle set-value-while-sliding? show-operator-logbook-in-this-window? show-prompts-for-type-in show-simulated-values simulation-control-specifications simulation-details simulation-formula simulation-procedure start-document-procedure start-element-procedure stub-length stubs superior-connection synchronized	table-size text text-conversion-style time-axis time-increment-for-update timeout-for-rule-completion timeout-interval title title-bar-text tokens-definition tracing-and-breakpoints trend-chart-format type-of-relation	uninterrupted-procedure-execution-limit update-callback uuid
validity-interval value-axes value-on-activation variable-or-parameter view-preferences	warning-procedure when-to-show-value width-of-image workspace-margin	

Here is the list of hidden system-defined attributes in G2 Version 8.3 Rev. 0:

active-stubs attribute-display-items	background-color border-color button-status	cached-media-bin chart-axis-computed-details chart-data-series class connection-input connection-is-directed connection-output connection-position-sequence connection-style connection-vertices containing-module current-attribute-displays
default-window-position-and-scale do-not-strip-text-mark dynamic-breakpoints	edges-of-workspace effective-data-type evaluation-attributes	following-item-in-workspace-layering format-type foundation-class

g2-array-sequence g2-hash-table-number-of-entries g2-hash-table-sequence g2-list-sequence g2-priority-queue-number-of-entries g2-priority-queue-sequence g2-window-client-version g2-window-is-embedded g2-window-of-view g2-window-ui-style	history history-using-unix-time	icon-color icon-heading icon-reflection icon-variables image-data inlined-calls internal-media-bin item-active item-color-pattern item-height item-notes item-status item-width item-x-position item-y-position items-in-this-relation
last-recorded-value-text latent-listeners layer-position	manually-disabled? minimum-size-in-workspace mouse-cursor	name-box name-box-item
parent-of-subworkspace permanent position-in-workspace	relationships representation-type	selected-items selected-window-handle selected-workspace size-in-workspace slider-value strip-text-mark stripe-color

table-cells table-header table-rows text-alignment text-color text-font text-x-magnification text-y-magnification transient type-in-box-value type-in-box-variable-or-parameter	ui-client-connection- status ui-client-mode-is-valid ui-client-operating- system-type ui-client-remote-host- name ui-client-specific- language ui-client-time-of-last- connection ui-client-user-is-valid ui-client-user-mode ui-client-user-name ui-client-user-name-in- operating-system uses-floating-license	value-structure value-structure-using-unix-time value-to-display values-for-table-of-values
window-handles		

Saving and Loading KBs

The g2-save-kb system procedure has the *file-progress-display* argument, which you can set to false to stop displaying file progress in the Telewindows client. Previously, this option only affected the display of the progress bar in the server.

The g2-merge-kb-ex system procedure has the merge-kb option, which you can set to false to load a KB instead of merging it. The various display options, such as the display of the progress bar, now work in the Telewindows client when merge-kb is false. Previously, these options only affected the display in the server.

Reading and Writing Bytes from and to a File

g2-read-bytes-as-text
    (*stream*: class g2-stream, *n*: integer)
    –> *bytes*: text

Reads bytes from a connected *stream* and returns it as a text, where *n* is the number of characters to read.

g2-read-bytes-as-sequence
    (*stream*: class g2-stream, *n*: integer)
    –> *bytes*: sequence

Reads bytes from a connected *stream* and returns it as a sequence, where *n* is the number of characters to read. This version is more useful for binary data.

The **g2-stream** status is one of the following symbols after calling these procedures:

> successfully-read-bytes
> end-of-file-reached
> text-truncated-during-read
> successfully-read-bytes
> tried-read-bytes-when-closed
> error-during-read-bytes
> tried-read-bytes-on-stream-not-opened-for-read

Both procedures are limited to the 1,000,000 byte length enforced by Gensym strings.

g2-write-bytes
(*stream*: class g2-stream, *data*: sequence)

Writes data to an existing non-listener connection represented by *stream*, where *data* is a sequence of 8-bit bytes to be written as binary data and a byte is a value from 0 - 255.

Upper Limit on UNIX Time Increased

The upper limit on textual timestamps that can be generated from unix-time floats, such as **g2-unix-time-to-text-4-digit-year**, is now Jan. 19, 2100 03:14:07 GMT. Previous, it was Jan. 19, 2038 03:14:07 GMT.

Networking and Web Integration

G2 Version 8.3 Rev. 0 provides system procedures for networking and integration with Web services, SOAP, HTTP, and TCP/IP sockets.

The system procedures described in "Web Services," "SOAP," and "HTTP" are located in `g2web.kb`, which is in `g2\kbs\utils` or `g2/kbs/utils`, depending on your platform.

For examples of interfacing with Web Services, SOAP, and HTTP, see `g2web-demo.kb`, located in the `g2\kbs\demos` or `g2/kbs/demos` directory of your G2 Bundle installation directory.

Web Services

According to the World Wide Web Consortium (W3C) Web Services Architecture Working Group Note (`http://www.w3.org/TR/ws-arch/`):

A Web service is a software system designed to support interoperable machine-to-machine interaction over a network. It has an interface described in a machine-processable format (specifically WSDL). Other systems interact with the Web service in a manner prescribed by its description using SOAP

messages, typically conveyed using HTTP with an XML serialization in conjunction with other Web-related standards.

G2 can act as a Web service requester agent (client), using the following system procedures in g2web.kb.

g2-import-web-service-description
    (*url*: text)
    –> *description*: class web-service-description

Imports a Web service description from a URL on the Web. This system procedure is equivalent to using g2-send-web-request and importing it by using g2-import-web-service-description-from-xml-text.

g2-import-web-service-description-from-xml-text
    (*document*: text)
    –> *description*: class web-service-description

Returns a web-service-description item from a WSDL *document*, expressed as XML text. Items representing the interfaces, bindings, and services defined by the document are placed on the subworkspace of the returned *description* item. An error is signaled if the text argument is not a well-formed WSDL document.

Currently, only WSDL 1.1 documents are supported. However, the names for G2 Web service system classes and attributes use the terminology of WSDL 2.0. In WSDL 1.1 terminology, an endpoint is a port and an interface is a portType. WSDL 1.1 messages do not exist in WSDL 2.0; an interface refers to schema elements directly through its interface message references.

g2-invoke-web-service-operation
    (*endpoint-reference*: structure, *operation-name*: symbol,
    input-message: structure)
    –> *output-message*: structure

Invokes an operation on a remote Web service given an *endpoint-reference* structure and *operation-name*, and returns the *output-message*. The *endpoint-reference* is a structure with this syntax:

```
structure
(service-namespace: text,
service-name: text,
endpoint-name: text)
```

The service-name and endpoint-name attributes identify an endpoint according to the web-service-description item whose XML namespace matches the service-namespace attribute.

An error is signaled if the *endpoint-reference* does not identify a valid endpoint, if the input message is not well-formed, or if an error occurs while communicating with the Web service.

The *input-message* and *output-message* are Web service message structures, which is a structure whose attributes correspond to message parts. The value of a message part attribute is a text, an XML element value, or a sequence of texts and/or XML element values.

An XML element value is a structure representing an XML element with this syntax:

```
structure
(tag-name: text,
attributes: structure,
children: sequence)
```

where:

- tag-name is the element tag name. This attribute is required.

- attributes is a structure containing named attribute values, which are texts. This attribute is optional.

- children is a sequence of XML elements and/or texts. This attribute is optional.

Attribute names use the same correspondence between XML names and G2 symbols used by G2GL, for example, myAttribute becomes my-attribute.

For example, this XML text:

```
<elt attrName="attrValue">
    <child>text1</child>
    text2
</elt>
```

corresponds to this XML element value:

```
structure
    (tag-name: "elt",
    attributes: structure
        (attr-name: "attrValue"),
        children: sequence (structure
            (tag-name: "child",
            children: sequence ("text1")), "text2"))
```

SOAP

G2 can send and receive SOAP 1.1 requests, using the following system procedures in g2web.kb.

g2-send-soap-request
  (*url*: text, *request*: structure)
  —> *response*: structure

  Sends a SOAP 1.1 request to a SOAP receiver at the given URL, returning the SOAP response when it arrives. Currently, only HTTP URLs are supported.

  The *request* structure has the following syntax:

  structure
  (header-entries: *sequence*,
   body-entries: *sequence*,
   action: *text*)

  where:

  • header-entries — A sequence of XML element values. For a description of XML element values, see g2-invoke-web-service-operation.

  • body-entries — A sequence of XML element values. This is the only required attribute of the structure.

  • action — The SOAPAction URI, indicating the intent of the request; this attribute is only used for HTTP.

  The *response* structure has the following syntax:

  structure
  (header-entries: *sequence*,
   body-entries: *sequence*)

g2-handle-http-request-as-soap
  (*server*: class http-server, *http-request*: structure,
  soap-dispatch: class procedure, *user-data*: item-or-value)
  —> *response*: structure

  Converts an *http-request* message into a SOAP request structure, passes it to the *soap-dispatch* procedure, and converts the resulting SOAP *response* structure into an HTTP response. The g2-handle-http-request-as-soap system procedure is intended to be called by the http-server-dispatch procedure of an http-server, which is described under g2-start-http-server. You can configure *user-data* to be any value, for example, you might want to pre-compute session information in the HTTP handler before determining if the request is a SOAP request, then pass the pre-computed information to the SOAP handler.

HTTP

Web Client

G2 can act as a Web client, using the following system procedure in `g2web.kb`.

g2-send-web-request
    (*url*: text, *request*: structure)
    −> *response*: structure

Sends a request to a Web server at the given *url*, returning the response when it arrives. The *request* structure and the returned *response* structure depend on the URL scheme. Currently, only HTTP URLs are supported. The *request* structure has the following syntax, where all attributes are optional:

structure
(method: *symbol*,
headers: *structure*,
entity: *structure | text*)

where:

- method — The HTTP method of the request, such as get or post. The default is get.

- headers — A structure of HTTP header attributes included in the request.

- entity — If provided as a structure, the attributes of the structure are encoded using the application/x-www-form-urlencoded MIME type. If provided as a text, the body of the request, which is assumed to already be encoded into 8-bit characters. Note that providing the entity as a text is only valid if the method is post.

The *response* structure has the following syntax:

structure
(http-version: *text*,
status-code: *integer*
reason-phrase: *text*,
headers: *structure*,
transfer-length: *integer*,
connection: *g2-socket*)

where:

- http-version — The HTTP version, for example, "HTTP/1.1".

- status-code — The RFC 2616 status code of the response message.

- reason-phrase — The reason phrase in the HTTP response message.

- headers — The message headers from the HTTP response message.

- transfer-length — The number of bytes to be transferred as part of the entity. If the transfer-length is -1, the entity length is undetermined and all bytes should be read from the connection.

- connection — If transfer-length is nonzero, a g2-socket instance from which the response entity can be read.

If the transfer-length is nonzero, the caller of the Web request is responsible for closing the g2-socket connection by using the g2-tcp-close system procedure.

g2-read-http-entity-body
   (*http-message*: structure)
   —> *entity-body*: text

Reads the *http-message*, decoding it according to the Transfer-Encoding header, as required, and returns the entity-body of an HTTP message as a text. The *http-message* structure is the same as the *response* structure returned by g2-send-web-request.

g2-check-http-response-status-code
   (*response*: structure)

Examines an HTTP *response* structure and signals an error if the status code indicates an HTTP error.

HTTP Server

G2 can act as an HTTP server, using the following system procedures in g2web.kb.

g2-start-http-server
   (*server*: class http-server, *port*: integer)

Starts a task that listens on the given TCP *port* number for HTTP 1.1 requests and passes them to the dispatch procedure of the specified *server*. The dispatch procedure is the value of the http-server-dispatch attribute of the http-server instance. The http-server-port attribute of the *server* is set to the specified *port*. If the server is already currently listening, it is shut down first.

The http-server-dispatch attribute of an http-server item names a G2 procedure with the following signature:

my-http-server-dispatch-procedure
   (*server*: class http-server, *request*: structure)
   —> *response*: structure

For each HTTP request that is received by the server, this procedure is called with the *server* and a *request* structure, which has the following syntax:

```
structure
(method: symbol,
request-uri: text,
http-version: text,
path: sequence,
query: text,
headers: structure,
entity: text)
```

where:

– method — The HTTP method of the request, such as get or post.

– request-uri — The URI of the request, which is everything after the host name in a URL. For example, if the URL is "http://www.gensym.com/index.asp?p=gensym_in_the_news", the URI is "/index.asp?p=gensym_in_the_news".

– http-version — The HTTP version, for example, "HTTP/1.1".

– path — A sequence of path segments in the request-uri. The path sequence begins with the first segment after the leading slash. In the example above, the path would be sequence("index.asp"). If there is no path, that is, the request-uri is an absolute URI with no slashes after the host, then the path sequence is empty. If the path is just "/", then the path sequence contains one empty text.

– query — (Optional) The portion of the URI that is the HTTP query, which follows the ?. In the example above, the query would be "p=gensym_in_the_news".

– headers — A structure of HTTP header attributes included in the request.

– entity — (Optional) The entity of the request, if any.

The *response* structure has the following syntax, where all the attributes are optional and no other attributes can be included:

```
structure
(status-code: integer,
reason-phrase: text,
headers: structure,
entity: text,
entity-producer: procedure,
user-data: item-or-value)
```

where:

– status-code — The RFC 2616 status code of the response. The default is 200 (OK).

– reason-phrase — A custom reason phrase. The default is the recommended reason phrase from RFC 2616 for the given status-code.

– headers — A structure of HTTP header attributes to be included in the request.

– entity — The entire entity text to be included in the response.

– entity-producer — A procedure that returns the entity incrementally. This attribute should be present when the response contains an entity that is too large to fit into a single entity text. The signature of the procedure is:

my-entity-producer
    (*response*: structure)
    −> *entity-bytes*: sequence

After the response headers are written to the HTTP connection, if the response structure includes an entity-producer attribute, the procedure that is its value is called repeatedly, and each return value sequence is written to the HTTP connection, using g2-tcp-write-bytes. An empty sequence signals the end of the entity.

– user-data — Any user-defined item or value. This can be used to pass other data to the entity-producer procedure.

The user may change the http-server-dispatch attribute while a server is running, in which case each incoming request goes to the current dispatch procedure. If this attribute is none or does not name an existing procedure, all requests receive a generic "404 not found" response.

g2-shutdown-http-server
    (*server*: class http-server)

Stops the listener task of the specified *server*. The http-server-port attribute is set to NONE. This procedure does nothing if the server is not currently listening.

Network Socket Communication

G2 provides classes and system procedures for managing network connections, using TCP/IP sockets, such as HTTP, and performing input/output operations to read and write data. These system procedures are located in sys-mod.kb.

Using the g2-socket Class

The system procedures for interfacing with sockets take as argument and return instances of the g2-socket class. The g2-socket class defines these read-only attributes:

- g2-socket-status — A symbol that indicates that status of the socket. The value is one of these symbols: newly-created, connected, connected-secure, listening, listening-secure, connection-closed, connection-closed-with-unread-data, connection-write-error.

- g2-socket-remote-host — A text that indicates the host name of the socket when available for a connected socket, or no value if the host is a listener socket.

Managing Connections to TCP/IP Sockets

Use the following system procedures to manage connections to TCP/IP sockets, all of which allow other processing.

g2-tcp-connect
> (*host*: text, *port*: integer, *options*: structure)
> –> *socket*: class g2-socket

Returns a g2-socket representing a TCP/IP connection established on a given host and port. *Host* is a text string representing a network host, for example, "www.gensym.com". *Port* is an integer, for example, port 80 for HTTP servers.

The *options* structure has this syntax:

structure
(secure: *boolean*)

where secure indicates whether to invoke SSL security on the connection. The default value is false. You can also specify an empty structure.

This system procedure returns a g2-socket instance that is appropriate as the argument to the system procedures used for reading and writing data over a network, as described in "Network Reading and Writing" on page 149.

This system procedure generates an error when various problems occur, for example, when a connection is refused.

g2-tcp-listen
(*port*: integer, *options*: structure)
−> *socket*: class g2-socket

Returns a g2-socket representing a TCP/IP listener on a given port, where *port* is an integer, for example, 80 for HTTP.

This system procedure returns a g2-socket instance that represents a listener, which is appropriate as the argument to g2-tcp-accept.

Using port numbers below 1024 on UNIX requires G2 to be running as root, as these are privileged ports.

The *options* structure has this syntax:

structure
(exact: *boolean*,
secure: *boolean*,
certificate: *text*)

where:

- exact — Whether to make an exact connection. If true, then if the listener could not be established on the exact port, it generates an error. If false, then if the listener cannot be established on the exact port, it increments the port number and tries again until it finds a port that is available. The default is false.

- secure — Whether to accept SSL security for clients that connect to this port. This option does not require the client to use SSL; it also accepts insecure connections. The new connection is reported as connected if it is insecure, and connected-secure if it is secure. The default is false.

- certificate — A string that identifies the SSL certificate to be used if the secure option is set to true. If the -cert G2 command line option has been given, it overrides the certificate option in the structure. Also, if another certificate was used to establish security, either for general G2/Telewindows communication or in another g2-tcp-listen call, that certificate is used instead. Thus, only one certificate may be active in a G2 session at one time, and once established, it is used for the remainder of the session.

g2-tcp-accept
(*socket*: class g2-socket)
−> *socket*: class g2-socket

Returns a g2-socket instance that represents an actual connection from an established listener. *Socket* is the g2-socket that represents the listener. Generally, a program should establish a listener by calling g2-tcp-listen, then enter a loop accepting connections by calling g2-tcp-accept, starting a new procedure invocation for each connection it received by g2-tcp-accept to handle the I/O.

g2-tcp-close
> (*socket*: class g2-socket)

> Closes an established TCP/IP connection represented by *socket* and deletes the g2-socket. You call g2-tcp-close on a g2-socket instance returned by g2-tcp-connect, g2-tcp-accept, or g2-tcp-listen.

> The connection may already be disconnected, for example, if the remote end has disconnected. However, because unread data may be buffered from this connection, the system procedure does not delete the g2-socket upon remote end closure, thereby allowing the buffered data to be processed. In this case, the g2-socket-status status of the g2-socket is connection-closed-with-unread-data. You must use this system procedure to remove the socket object from G2 memory.

Network Reading and Writing

In general, the system procedures that perform I/O through sockets use the same procedure names as the system procedures that perform I/O using streams. However, note that the I/O system procedures for both sockets and streams are implemented as methods rather than as procedures. All system procedures allow other processing.

The following system procedures write data to a socket:

g2-write-string
> (*socket*: class g2-socket, *data*: text)
> −> *success*: truth-value

> Writes data to an existing non-listener connection represented by *socket*, where *data* is a text string. The procedure returns true if the string is written successfully; otherwise, it returns false.

g2-write-bytes
> (*socket*: class g2-socket, *data*: sequence)
> −> *success*: truth-value

> Writes data to an existing non-listener connection represented by *socket*, where *data* is a sequence of 8-bit bytes to be written as binary data and a byte is a value from 0 - 255. The procedure returns true if the bytes are written successfully; otherwise, it returns false.

The following system procedures read data from a socket and update the g2-socket-status of the g2-socket accordingly:

g2-read-block
> (*socket*: class g2-socket)
> −> *block*: text

> Reads a block of text from a non-listener connection represented by *socket*.

This system procedure does not have an analogue in file I/O; it simply reads whatever data is presently available, or waits for new data and returns the text that arrives. It returns an empty text when the connection becomes closed and there is no more data.

g2-read-byte
(*socket*: class g2-socket)
−> *byte*: integer

Reads a byte from a non-listener connection represented by *socket* and returns it as either an 8-bit integer or -1 if G2 is unable to read it.

g2-read-bytes-as-text
(*socket*: class g2-socket, *n*: integer)
−> *bytes*: text

Reads bytes from a non-listener connection represented by *socket* and returns it as a text, where *n* is the number of characters to read.

g2-read-bytes-as-sequence
(*socket*: class g2-socket, *n*: integer)
−> *bytes*: sequence

Reads bytes from a non-listener connection represented by *socket* and returns it as a sequence, where *n* is the number of characters to read.

g2-read-line
(*socket*: class g2-socket)
−> *line*: text

Reads a line of text from a non-listener connection represented by *socket* and returns it as a text.

Note   Currently, not all character sets are supported, so lines end with a CRLF.

G2 Bundle

Platform Support

The G2 Bundle supports and will support the following platforms and operating systems in the following versions. Note that several platforms and operating systems will no longer be supported in G2 Version 8.3 Rev. 0 or in a future release (shown in bold). The OS versions listed below are the minimum OS versions. Because OS vendors maintain upward compatibility between versions, G2 is compatible with more recent versions.

Platform/OS	G2 Version 8.0 - 8.2	G2 Version 8.3 Rev. 0	Post-G2 Version 8.3 Rev. 0
Windows	Windows NT 4.0 Windows 2000 Windows 2003 Windows XP Pro	Windows 2000 Windows 2003 Windows XP Pro **Windows NT 4.0 not supported**	Windows 2000 Windows 2003 Windows XP Pro **Windows NT 4.0 not supported**
Solaris SPARC	Solaris 2.6	Solaris 2.6	Solaris 8
HP-UX PA-RISC	HP-UX 10.20 HP-UX 11.0	HP-UX 11.0 HP-UX 11i V2	HP-UX 11.0 HP-UX 11i V2
HP-UX Itanium	Not supported	HP-UX 11.23	HP-UX 11.23
HP Tru64 UNIX	Tru64 V5.1A	Tru64 V5.1A	**HP Tru64 not supported**
AIX	AIX 4.3.3 AIX 5.1 w/ Javalink	AIX 5L V5.2	AIX 5L V5.2
Red Hat Linux	2.4.26 Kernel	2.4.26 Kernel	2.4.26 Kernel

G2 Version 8.3 Requires New OK File

G2 Version 8.3 requires a new `g2.ok` file, which ships with the bundle. G2 Version 8.3 uses the `G2V_83` and `TWV_83` environment variable and `-v83` command-line option for specifying the location of the OK file.

G2 OK File User Names and Passwords

Both the user name and the password that identify an authorized user of a secure G2 can now include hyphen, underscore, period, and question mark characters. The first character must still be alphabetic.

Support for Secure Communication on UNIX

G2 supports encrypting communication between G2 and Telewindows on UNIX platforms, using the following command-line options:

`-secure`

Use OpenSSL on all TCP/ICP connections.

`-cert` *file*

Specifies the name of the OpenSSL server certificate to use, where *file* is a file containing a private key and a certificate in PEM format, which consists of the DER format base64 encoded with additional header and footer lines.

You can use the `G2_CERT` environment variable to provide the default certificate name.

G2 also supports secure communication between G2 and G2 Gateway, G2 and G2 ActiveXLink, and G2-to-G2 connections. For details, see "Support for Secure Communication" on page 200.

For OpenSSL copyright information, see the `readme-g2.html` file.

Log Command-Line Option on UNIX

G2, G2 Gateway, and Telewindows supports the `-log` command-line option on UNIX platforms for redirecting standard output and error output to a log file.

For compatibility with Windows, the `-no-log` is also supported on UNIX platforms, although no logging is the default.

Exit-on-Abort Command-Line Option

G2 supports a new command-line option, `-exit-on-abort`, which causes G2 to exit if a G2 abort occurs.

Windows Help

The Windows help now includes `.chm` files for all books in the G2 bundle, including all the bridges, *G2 Gateway User's Guide*, and all the G2 Developer's Utilities user's guides.

Part III

Telewindows

Telewindows
Version 8.3 Rev. 1

Describes the new features and changes in Telewindows Version 8.3 Rev. 1, which ships with the G2 Bundle.

Introduction

This chapter describes Telewindows Version 8.3 Rev. 1, which ships with the G2 Bundle Version 8.3 Rev. 1.

Information on bug fixes and known bugs is available separately as Technical Bulletins through HelpLink at `www.gensym-support.com`. For information on bugs in Telewindows Version 8.3 Rev. 1, choose Technical Bulletins for Telewindows, specify Starting From Version as 8.3 Rev. 0 and Ending At Version as 8.3 Rev. 1, choose the bugs and suggestions that either Remain Open or Were Fixed, then click Execute.

Telewindows
Version 8.3 Rev. 0

Describes the new features and changes in Telewindows Version 8.3 Rev. 0, which ships with the G2 Bundle.

Introduction

This chapter describes Telewindows Version 8.3 Rev. 0, which ships with the G2 Bundle Version 8.3 Rev. 0.

Information on bug fixes and known bugs is available separately as Technical Bulletins through HelpLink at `www.gensym-support.com`. For information on bugs in Telewindows Version 8.3 Rev. 0, choose Technical Bulletins for Telewindows, specify Starting From Version as 8.2 Rev. 4 and Ending At Version as 8.3 Rev. 0, choose the bugs and suggestions that either Remain Open or Were Fixed, then click Execute.

Telewindows Next Generation Only Features

These features are only supported in Telewindows Next Generation (`twng.exe`)

- Windows views:
  - Chart views
  - HTML views
  - Tree view
  - Shortcut bar
  - Status bar
  - Property grid
  - Tabbed MDI mode
  - Native logbook
  - Native message board
- Custom Windows dialogs
- Toolbars

Removed Support for Multiwindow Mode

Telewindows Version 8.3 Rev. 0 has several changes designed to simplify the use and deployment of Telewindows client user interface. Users are advised to take into account these changes while developing plans for taking advantage of G2 Version 8.3.

Changes in deployment options include:

- G2 Version 8.3 will refuse connections from Telewindows Next Generation clients (`twng.exe`) prior to Version 8.3. Similarly, Telewindows Next Generation Version 8.3 will not connect to G2 versions prior to G2 Version 8.3.

- G2 Version 8.3's ActiveX control for workspace views will be based on Telewindows Next Generation Version 8.3.

- The non-native Windows version of Telewindows (`tw.exe`) will no longer support the `-ui multiwindow` command-line option, custom dialogs, and the native debugger dialog. Removing it will simplify installations and Gensym's engineering efforts. Users should work with Telewindows Next Generation to leverage custom native dialogs and a multiple document interface with scrolling workspaces if this functionality is required.

- G2 Version 8.3 will continue to accept connections from all earlier versions of cross-platform Telewindows running on any of G2's supported operating systems (`tw.exe` on Windows and `tw` on UNIX).

Please also note that G2 Version 8.3 will continue Gensym's tradition of maintaining full application compatibility to enable loading and running G2 knowledge bases built in prior versions of G2.

Text Editor

The text editor has a number of significant enhancements:

- Automatic insertion of G2 code by clicking grammar prompts that are always visible in a separate window in the editor.

- Automatic insertion of a space after grammar prompts.

- Toolbar buttons for cut, copy, paste, undo, redo, delete, find, go to item, save, and save and exit.

- Expanding and collapsing nested blocks of code, such as begin-end, if-then-else, and repeat statements.

  The native editor displays a margin that contains boxes and lines that indicate the nesting structure of a procedure being editing. Click on a box to toggle the expansion of that block of code. Press Control+F8 to toggle the visibility of the margin itself. The margin is visible by default for all items; however, it is only useful when editing procedures.

- Automatic indentation when pressing Enter to match the indentation of the previous line.

  Pressing Enter in the native editor now indents relative to the previous line, unless cursor is within a string, comment, symbol, or free text. To insert a new line without indenting, you can use Control+J.

- When editing attributes in tables, pressing Enter to accept the value, rather than CTRL+Enter.

  Editing some symbol or number type attributes (the name of an item is one example) puts the native editor into single-line mode. In this mode, the Enter key does "save and exit". Also, the editor is initially smaller, with a text buffer just tall enough to show the initial text.

- New icons for text editor buttons.

The following figures show examples of some of the new features.

This figure shows the text editor with the new toolbar buttons and the automatic grammar prompts that appear when creating a rule
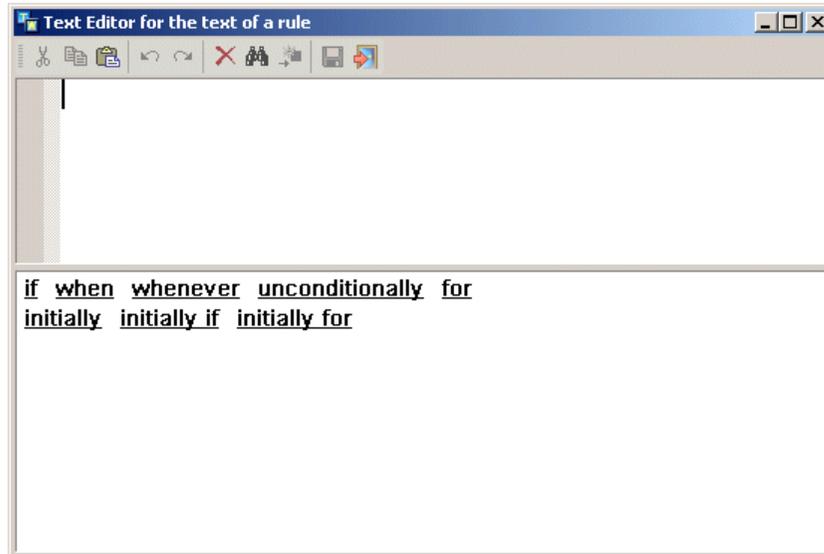
Click a keyword to enter it in the editor and update the grammar prompts below. The keyword automatically has a space after the word. G2 reserved words and other syntax such as parentheses appear in bold underline. Arguments to keywords appear in italics underline. For example, here are the grammar prompts that appear when creating a whenever rule:
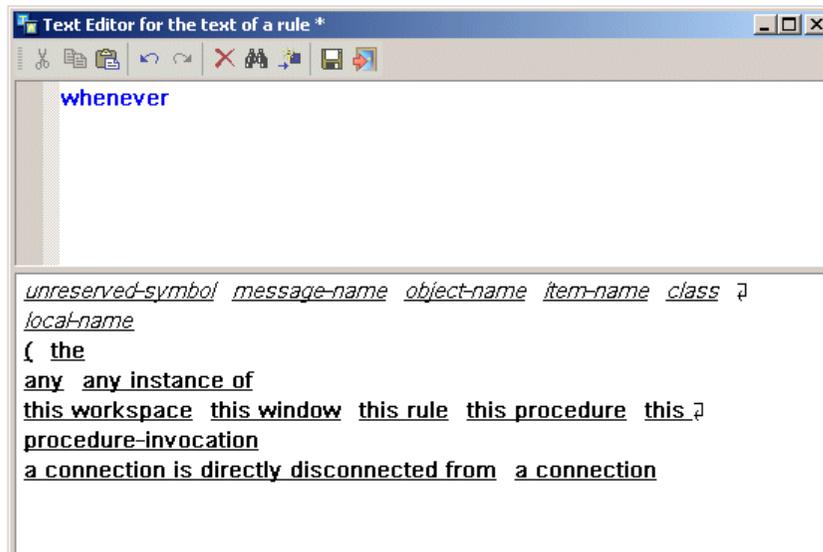
Here are the grammar prompts that appear after clicking any instance of:
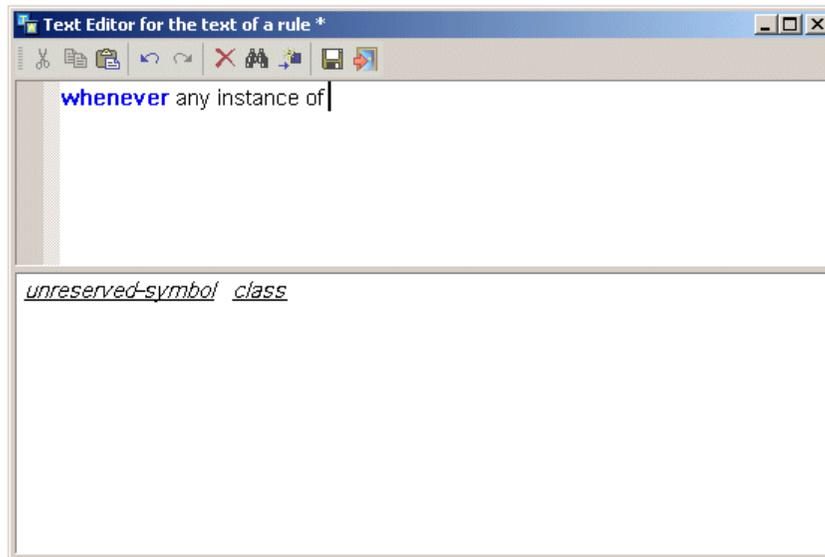
Clicking _**class**_ updates the grammar prompts to show all available classes:
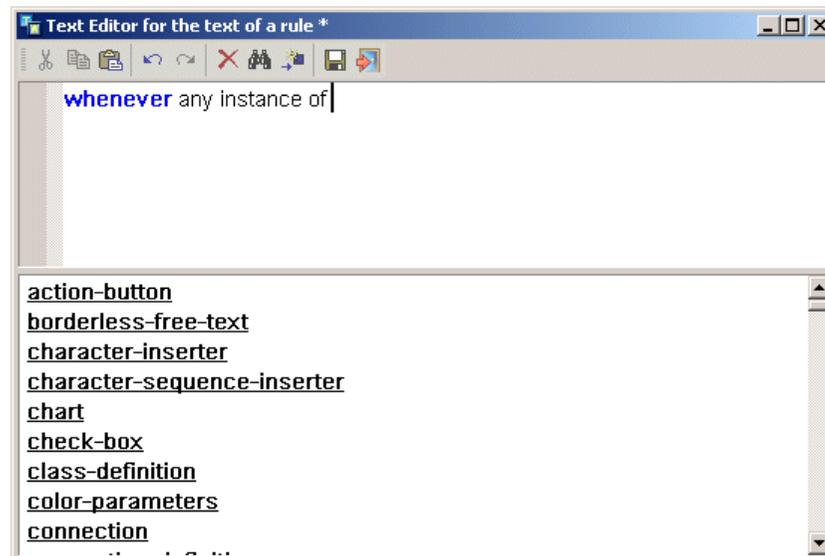
This figure shows the text editor when editing a procedure, where the begin-end block is expanded and the if-end block is collapsed:
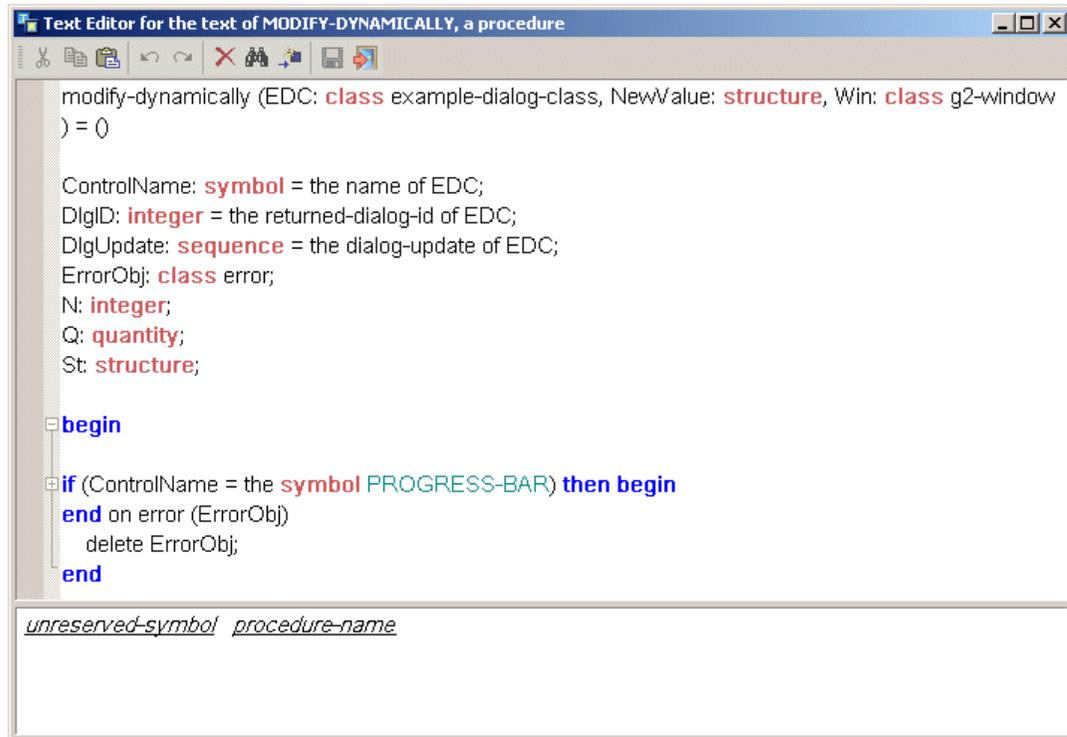
This figure shows the same procedure with the **if-end** block expanded, which contains a **repeat** block and a second **if-end** block, both of which are collapsed:

To go to a named item, simply place the cursor anywhere within the text of the named item and click the Go To Item button or press the F12 key. For example, placing the cursor on the example-dialog-class class in the argument list places the cursor o the class definition:
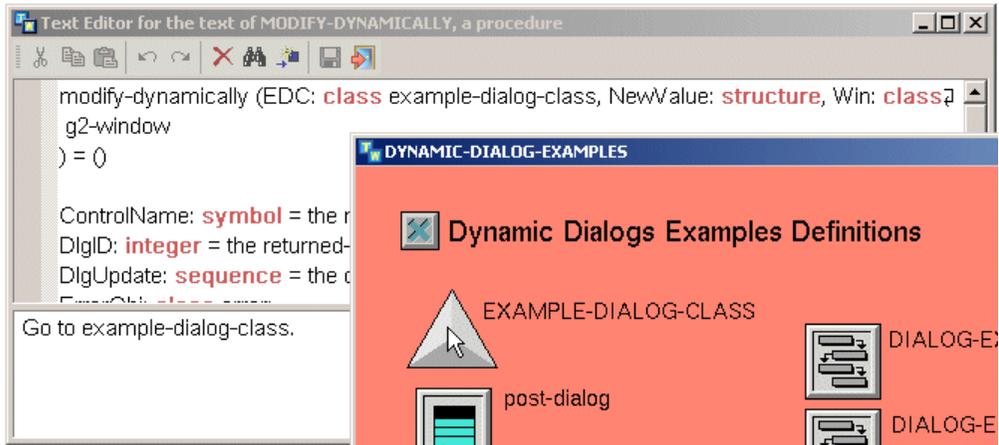
G2 Utilities

G2 Developer's Utilities

Describes the G2 Developer's Utilities, which provide numerous high-level tools for application development in these areas: business rules, integration with Web services, reporting and charting, event and data processing, event and alarm management, message queuing and browsing, Windows dialog utilities, error handling, data source management, datapoint management, and relation browsing.

Introduction

This chapter describes the G2 Developer's Utilities (G2i) Version 2.3 Rev. 1, which ships with the G2 Bundle Version 8.3 Rev. 1.

Information on bug fixes and known bugs is available separately as Technical Bulletins through HelpLink at `www.gensym-support.com`. For information on bugs in the G2 Developer Utilities Version 2.3 Rev. 1, choose Technical Bulletins for G2 Developer Utilities, specify Starting From Version as 2.3 Rev. 0 and Ending At Version as 2.3 Rev. 1, choose the bugs and suggestions that either Remain Open or Were Fixed, then click Execute.

G2 Developer's Utilities Version 2.3 Rev. 1

G2 Developer's Utilities Version 2.3 Rev. 1 is a compatibility release only. It has no new features or changes. For information on bug fixes, see the Introduction.

G2 Developer's Utilities Version 2.3 Rev. 0

Introduction to the G2 Developer's Utilities

While some G2 Developer's Utilities were introduced in prior versions of G2, G2 Version 8.3 introduces several new modules and includes significant extensions to existing modules. The purpose of these modules is to provide a consistent way of developing G2-based applications, improve developer productivity, and provide consistency in applications.

These modules support a consistent development framework in these three areas:

- Application development with an emphasis on decision management applications.

- Architecture that focuses on three layers: data access layer, business layer, and presentation layer.

- Implementation within an extensible framework.

This section briefly summarizes decision management, while the following section summarizes each module within the architectural layer. Gensym's application products are examples of extensible solution frameworks that leverage these modules.

Decision Management

G2 application architecture follows a paradigm for operational decision management. The decision management paradigm comprises three elements: the managed system, the managed system model, and the management logic. The managed system is the real system being controlled or optimized. The management logic is the reasoning used to achieve the objectives of control or optimization. A managed system model is a model of the managed system, typically used for system -wide analysis and predictions.
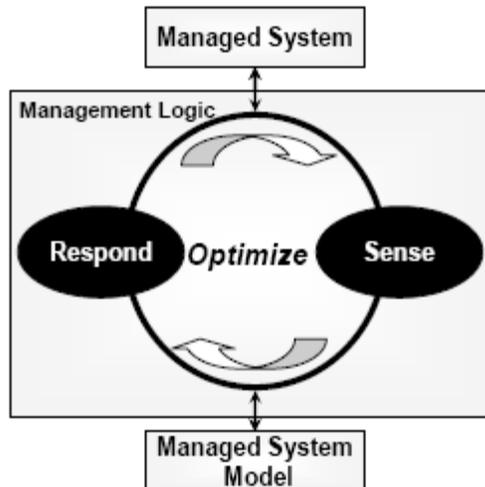
Managed System

The managed system is the real system being controlled or optimized. The managed system may be hardware, software, or some combination of hardware and software. Gensym applications interface with the managed system through bridges or adapters. In a process control application, the managed system is a reactor, a group of production processes, or a whole plant. In a supply chain application, the managed system is the supply chain or some portion of the supply chain. In a manufacturing execution system (MES) application, the managed system is the scope of production processes and associated equipment to be synchronized. In a managed service provider application, the managed system is the network of machines or equipment or software applications. In a rules-driven business process application, the managed system is data or information in a database.

Managed System Model

You use Gensym software to develop models of managed systems. The managed system model extends management logic by providing projections or forecasts. Examples of managed system models are supply chain models (e.g., G2 e-SCOR supply-chain models), continuous production process models (e.g., G2 NeurOn-Line software sensor models), discrete or batch production line models (e.g., G2 ReThink workflow models), communication network models (e.g. G2 Integrity network models) and models of battle space (e.g., G2-JMACE).

Management Logic

You use Gensym software to develop management logic to control or optimize managed systems. The management logic in a Gensym application follows a sense-respond paradigm. Sense tasks have as a goal determining the state or condition of the managed system. Respond tasks have as a goal the development

of actions to control or optimize the managed system. Gensym applications optimize a managed system by iterating through the sense-respond cycle. Management logic is also a model and, when linked into a managed system model, can be simulated to test or validate that the management logic works as expected.

The following diagram summarizes typical capabilities used in Decision Management Solutions and the entities describing the Managed System:

disturbances

Capabilities - verbs
Optimize
Correlate
Diagnose
Model
Simulate
Alarm
(to) Message
Visualize
Control
Adapt
Plan
Schedule
Advise
Detect
Isolate (a root cause)
Trend
Infer
Reason

inputs

Managed System

outputs   KPIs Metrics

actions

raw data

Control
Plan
Schedule
Alarm
•
•
•

Correlate
Diagnose
Detect
Trend
•
•
•

conditions, states, events

Model of the Managed System

Model, Simulate, Forecast, (make) Projection • • •

"Things" - nouns
Data
Knowledge
(e.g., rules, etc.)
Information
Message
Events

Networks
Supply Chains
Processes
Sensors
Production Lines
Buildings
Pumps, compressors, etc.
Pipelines
(oil & gas) platforms
Financial instruments
Warehouses

Legend
☐ Capabilities (G2, Partners, etc)
☐ Capabilities we don't provide

Application Framework: G2 Developer's Utilities

The suite of G2 Developer's Utilities implement patterns typically used in decision management solutions and cover different aspects of the architecture of a solution. To build complete solutions, you might use other modules, from Gensym or a third party, or you might develop custom modules.

This diagram shows how these modules map into the three layers in a decision management application:

Rich Client - Telewindows **Thin Client - Web**

Presentation Layer

GRTL | GEVM | GWEB | GRPE | GDU | GRLB

Business Layer

BPMS | BRMS | GEDP | SymCure* | NeurOn-line*

Data Access Layer

GDSM | GDPM | GWEB | GEUC

Data Sources **Services**

Alarms and Event States (GEVM)

Runtime (GRTL & GERR)
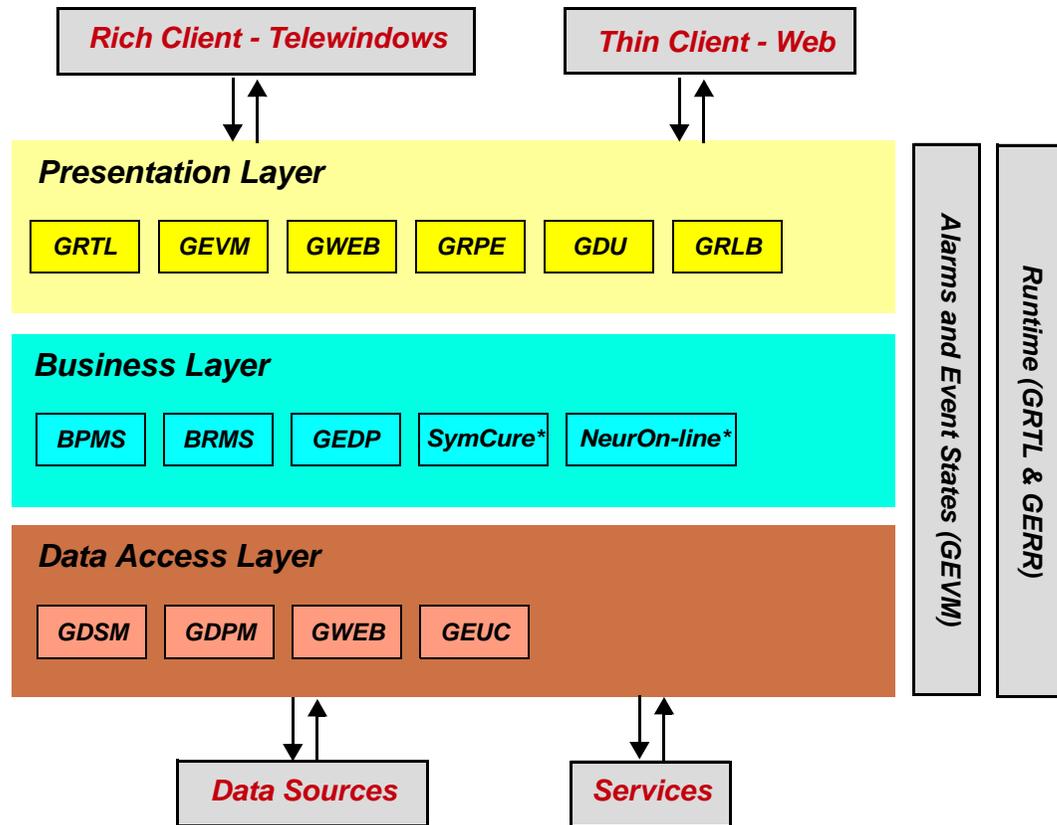
The following sections briefly summarize the G2 Developer's Utilities modules and the changes in G2 Version 8.3.

G2 Run-Time Library (GRTL)

G2 Run-Time Library (GRTL) is a core module that defines a common set of utility functions, an object model, and user interface patterns to speed up the development process and provide a consistent object model. GRTL provides an infrastructure for addressing key issues within a common framework, including:

- Utility functions and procedures GRTL provides numerous utility functions to improve productivity and build consistency in applications.

- Application configuration. GRTL includes APIs to extract application configuration from external text configuration files, in addition to providing KB-level configuration of modules.

- Application localization. GRTL includes APIs to localize text into multiple languages with pattern replacement.

- Object Model. GRTL includes a core object model including APIs for common operations.

- Object Repository. GRTL includes a mechanism to dynamically store objects created by users as they configure their application.

- User Interface. GRTL includes a user interface for rich clients, using Telewindows Next Generation. The user interface is dynamically enhanced as modules are merged in and organizes elements used in decision management applications.

For more information and a complete description of the enhancements in this version, see the *G2 Run-Time Library User's Guide*.

Here is a summary of what's new in G2 Version 8.3:

- Additional system configurations. The grtl-module-settings object has these additional attributes: date-time-format, utc-offset, user-preferences-filename, available-user-modes, installation-directory, add-missing-message-localization-to-gfr-resource, available-user-modes, enable-menus-and-toolbars-upon-startup, kb-item-configuration-for-classes-enabled-for-user-modes, kb-item-configuration-for-classes-enabled, and show-selection-handles, update-top-level-system-tables.

- KB initialization enhancements: The ability to configure and update the configuration of system tables of the top-level KB.

- User preferences enhancements:
  – Configurations may now be saved and restored from a file.
  – Selection between MDI and TDI display modes.
  – Better support for custom user interface management and layout.
  – Ability to set the default user mode upon connection.

- User interface. The user interface has been significantly enhanced, including loading specifications from the configuration file:
  – New application navigator/project menu that organizes elements used in decision management applications.
  – New configurable status bar.
  – Support for keyboard accelerators, including built-in keystrokes.
  – Support for large and small icons in palette panes.
  – Support for a status bar.
  – Ability to configure menus, toolbars, and status bars upon startup.
  – Configurable menubar, toolbars, status bar, and window panes from the configuration file.

- Object model extensions:
  - Domain objects include information about their availability status, such as active, inactive, in-repair, or failed.
  - Ability to differentiate between detection, test, response, or general grtl-scheme.
  - XML serialization APIs.
- Additional utility APIs:
  - Localization of messages based given a specific GFR text resource.
  - Ability to define menu items using a specific GFR resource for text localization.
  - Support for additional timestamp formats.
  - Ability to extract different type of information from the configuration file.
- GRTL module split into separate modules: grtl.kb, grtl-utils.kb, and grtl-icons.kb.

G2 Event Manager (GEVM)

G2 Event Manager (GEVM) provides two basic capabilities for event management:

- Operator messages, including all kind of messages that need to be generated and presented to operators. Operator messages include messages about the loss of connection to a bridge, advisory messages, fault messages, root cause messages, action request messages, as well as errors generated in the code that need to be displayed to developers, administrators, and operators.

- Blackboard of internal event states that are detected by event-detection logic but are not presented to the operator. These so-called "raw events" encapsulate the existence of a state such as a temperature exceeding a limit or a network device failing. Filtering and diagnostic logic can use these event states to generate operator messages or to trigger corrective actions. The benefit of using a blackboard of events is to provide explicit state information that can be used by multiple correlation engines, including replicating the blackboard of events across servers to build distributed and scalable applications.

For more information and a complete description of the enhancements in this version, see the *G2 Event Manager User's Guide*.

Here is a summary of what's new in G2 Version 8.3:

- New and enhanced operator messages:
  - Additional attributes to assign and filter messages by groups instead of by individual users.
  - New severity alarm to filter and group messages at a higher level of granularity than the priority.
  - New alarm messages classes.
- Message browser usability enhancements:
  - Additional action buttons to forward messages via email, ping or telnet to the target associate with messages, display best practices associated with the target associated with messages (such as procedural, shutdown, maintenance), forward operator messages via email, view the history of changes for a specific message, and run detection, test, or response logic.
  - Message actions available as toolbar buttons and in popup menus. Actions may be displayed in the toolbar, the popup menu displayed in the tabular view, in both, or not at all.
  - Filtering of messages based on availability of target associated with a message, which enables you to filter out messages for equipment that is in maintenance mode or has been put offline, for example.
- New alarm reports:
  - Reports such as acknowledged messages for individual equipment and top alarms.
  - Statistical reports such as occurrence of alarms on a weekly basis, or frequency of alarms for individual equipment.
- G2 errors:
  - Errors signaled by G2 or application code may now be rerouted to the GEVM message queues and browsers.
- APIs for sending email for a gevm-message.

G2 Web (GWEB)

G2 Web (GWEB) defines out-of-the-box Web pages, SOAP services, WSDL support, as well as classes and APIs enabling G2 to implement an HTTP server and serve HTML pages, XML structures, SOAP services, G2GL/BPMS processes, and files.

GWEB supports using both G2 WebLink and the integrated G2 HTTP server. The APIs for both are similar and support defining Web pages that provide a level of abstraction to use either HTTP server. The only difference is that the G2 built-in

HTTP server supports SSL, SOAP, and WSDL, whereas the G2 WebLink HTTP server does not.

For more information, see the *G2 Web User's Guide*.

G2 Reporting Engine (GRPE)

G2 Reporting Engine (GRPE) defines out-of-the-box reports, classes and APIs to define reports and charts, and dialogs to configure and visualize them. GRPE provides a consistent approach for defining reports and charts, collecting values, displaying tabular values in reports, and charting those values.

GRPE supports collecting values from CSV files, databases, or G2 items, displaying values in Telewindows Next Generation, and exporting values to CSV files, Excel, databases, and G2 items. The contents of a report can be updated on demand, using menu choices or APIs, or on a regular interval.

Charts support the definition of multiple chart views.

GRPE also defines APIs for easy integration with Excel. These APIs enable you to easily configure Excel workbooks dynamically from G2. To enable this functionality, GRPE provides an Excel add-in. Once installed and included in Excel, you can connect Excel to a G2 server and receive dynamic updates from G2.

For more information, see the *G2 Reporting Engine User's Guide*.

G2 Dialog Utility (GDU)

G2 Dialog Utility (GDU) is a core module to build static and dynamic dialogs.

For more information and a complete description of the enhancements in this version, see the *G2 Dialog Utility User's Guide*.

Here is a summary of what's new in G2 Version 8.3:

- Keystroke and mouse click notification in tabular view and grid view dialog controls.

- Selection notification in grid view control.

- New APIs for adding a control to:

  – Show the dialog details.

  – Select instances.

  – Edit expressions using the G2 text editor.

  – Toggle the button state.

  – Select a directory.

- New APIs to manage dialogs.

- New APIs to convert between grid coordinates and dialog units.

Note	The signature of the activation procedure for adding a grid-view has changed.

G2 Relation Browser (GRLB)

The G2 Relation Browser (GRLB) provides the ability to display G2 relations and user-defined relations in a graphical layout. For example, SymCure, a component of Optegrity and Integrity, is a graphical modeling language used to build cause-and-effects models and makes use of GRLB for displaying relations in generic and specific event models. For more information, see the *SymCure User's Guide*.

Business Process Management System (BPMS)

Business Process Management System (BPMS) is based on the G2 Graphical Language (G2GL). The process models, debugging capabilities, and run-time environment are based on G2GL. The BPMS module provides the following extensions:

- Windows dialogs and palettes for all G2GL blocks.

- A class hierarchy of G2GL process subclasses to organizes the processes as detection flows, tests, responses, or orchestration processes. These processes are automatically organized in the navigator tree view and the Project menu.

- Standard menus, message browser integration, and APIs to invoke the detection, test, and response processes for a domain object.

- Predefined G2 services that can be called from G2GL processes including services for interacting with OS processes, performing ping and trace route operations on a computer, sending email, interacting with databases and files, creating or querying operator messages and event states, generating SymCure events, and invoking BRMS rules.

For more information, see the *Business Process Management System User's Guide*.

Business Rules Management System (BRMS)

Business Rules Management System (BRMS) provides a mechanism for easily editing, organizing, analyzing, and executing business rules. You define business rules for a class of G2 objects in a given category. A business rule consists of one or more conditions and actions, which you define interactively based on the class. You invoke rules programmatically by invoking all rules in one or more categories for a set of G2 objects.

You can use BRMS to create individual business rules or to create more complex sets of related business rules in a decision table, which tests conditions and take actions on attributes defined on G2 objects. You can edit these decision tables in G2 or Excel.

You can analyze business rules for conflicts, collisions, and redundancies, create a report for a given time period that indicates various statistics about all the rules in

your application, and show explanation data for attributes modified by the rules you have defined.

For more information, see the *Business Rules Management System User's Guide*.

G2 Event and Data Processing (GEDP)

G2 Event and Data Processing (GEDP) is a multi-purpose graphical language. It is composed of graphical blocks that can be connected together to express a flow of data, perform calculations, execute functions, generate messages, and events. GEDP flow diagrams are typically used to analyze numeric values, detect patterns and generate event states. You use it to monitor sensor information and build event-detection diagrams. The result of the event detection flow typically records event stated, for example, using GEVM events, and optionally generates operator messages or triggers diagnostics logic, for example, using SymCure. Generic templates behave like methods, whereby each generic template is uniquely identified by its name and its target class, just as a method is defined by the method name and class.

For more information, see the *G2 Event and Data Processing User's Guide*.

G2 Data Source Manager (GDSM)

G2 Data Source Manager (GDSM) defines classes and APIs related to managing network connections. GDSM provides a consistent approach for configuring, connecting, disconnecting, and monitoring network connections to and from remote G2 processes and bridges. GDSM monitors network connections, detects and reports problems to operators, and attempts automatic reconnection to the remote G2 process or G2 bridge when a connection is lost. In addition, GDSM also supports an option for automatically launching bridge processes.

For applications performing many exchanges with remote processes or bridges, GDSM provides a network pooling mechanism. Network pools are typically used to provide scalable throughput and be transparent to the application logic. Networking pooling enables you to have a pool of connections to a database, for example, and to perform multiple queries in parallel, picking the next available database connection.

For more information and a complete description of the enhancements in this version, see the *G2 Data Source Manager User's Guide*.

Here is a summary of what's new in G2 Version 8.3:

- Agent to manage remote resources such as files, processes, and logging.

- Management of bridge process, including automatic startup and shutdown of the bridge process by G2, as required.

- Monitoring of network interfaces on a periodic basis and automatic reconnection to bridge process upon failure.

- Configuration of default interfaces from configuration file. This allows easy administration and automatic configuration of bridges upon G2 startup.

- GDSM module split into separate modules for each type of interface.

- gdsm-demo.kb added with examples.

G2 Data Point Manager (GDPM)

G2 Data Point Manager (GDPM) provides functionality to configure, log, replay, and simulate datapoints, typically related to external sensors such as temperature, pressure, and flow. These external values are represented in GDPM as external datapoints and obtain their values typically via an OPC or PI interface and bridge. For example, to provide connectivity with DCS systems, you could use the G2-OPC Client Bridge and the G2-PI Bridge.

For more information, see the *G2 Data Point Manager User's Guide*.

G2 Engineering Unit Conversion (GEUC)

G2 Engineering Unit Conversion (GEUC) provides a way of specifying the engineering units for entering and displaying values, as well as a large number of synonyms for those conversions in both the English and metric systems. GEUC defines a large set of built-in engineering unit conversions and synonyms for dimensions such as pressure, length, volume, volumetric flow, mass, density, temperature, power, heat transfer, and time. It also provides a mechanism for defining custom dimensions, engineering units, and synonyms.

For more information, See the *G2 Engineering Unit Conversion User's Guide*.

G2 Error Handling Foundation (GERR)

G2 Error Handling Foundation (GERR) is a core module that provides a common approach for dealing with errors, including logging them and reporting them.

For more information and a complete description of the enhancements in this version, see the *G2 Error Handling Foundation User's Guide*.

Here is a summary of what's new in G2 Version 8.3:

- Easier configuration of error handling.

- Best practices to throw errors and handle errors.

- Error source trace attribute on error objects.

Documentation

The following manuals are new for this release:

- BPMS — *Business Process Management System User's Guide*

- BRMS — *Business Rules Management System User's Guide*

- GRPE — *G2 Reporting Engine User's Guide*

- GWEB — *G2 Web User's Guide*

- GEDP — *G2 Event and Data Processing User's Guide*

- GEUC — *G2 Engineering Unit Conversion User's Guide*

- GDPM — *G2 Data Point Manager User's Guide*

In addition, the *G2 Dialog Utility User's Guide* and *G2 Developers' Utilities Runtime Library User's Guide* have been split into separate books for each module. These guides no longer exist. Instead, the G2 Bundle provides the following books, which document these modules:

- GRTL — *G2 Run-Time Library User's Guide*

- GEVM — *G2 Event Manager User's Guide*

- GDU — *G2 Dialog Utility User's Guide*

- GDSM — *G2 Data Source Manager User's Guide*

- GERR — *G2 Error Handling Foundation User's Guide*

- GRLB — *G2 Relation Browser User's Guide*

In addition, the *G2 Event Manager User's Guide* has been significantly restructured, and numerous features of the API that were previously not exposed are now documented.

These books are provided in PDF and HTML formats. On Windows platforms, they are also available in Winhelp format and are accessible through the Start menu.

G2 Utilities

Describes new features of the G2 utilities, which ship with the G2 Bundle.

gensym

Introduction

This chapter describes the G2 Utilities Version 8.3 Rev. 1, which ships with the G2 Bundle Version 8.3 Rev. 1.

Information on bug fixes and known bugs is available separately as Technical Bulletins through HelpLink at `www.gensym-support.com`. For information on bugs in the G2 Utilities Version 8.3 Rev. 1, choose Technical Bulletins for the particular utility, specify Starting From Version as 8.3 Rev. 0 and Ending At Version as 8.3 Rev. 1, choose the bugs and suggestions that either Remain Open or Were Fixed, then click Execute.

G2 Utilities Version 8.3 Rev. 1

The G2 Utilities Version 8.3 Rev. 1 is a compatibility release only. It has no new features or changes. For information on bug fixes, see the Introduction.

G2 Utilities Version 8.3 Rev. 0

GFR uses g2-get-g2-version-information instead of g2-get-software-version to provide version information.

Otherwise, the G2 Utilities Version 8.3 Rev. 0 is a compatibility release only. It has no new features or changes. For information on bug fixes, see the Introduction.

Part V

G2 Bridges and Integration

Chapter 16   **G2 Java Socket Manager**   213

Describes new features of G2 Java Socket Manager, which ships with the G2 Bundle.

Chapter 17   **G2 JMSLink**   217

Describes new features of G2 JMSLink, which ships with the G2 Bundle.

Chapter 18   **G2-OPC Client Bridge**   219

Describes new features of the G2-OPC Client Bridge, which ships with the G2 Bundle.

Chapter 19   **G2-PI Bridge**   221

Describes new features and changes in the G2-PI Bridge, which ships with the G2 Bundle.

Chapter 20   **G2-SNMP Bridge**   223

Describes the G2-SNMP Bridge, which ships with the G2 Bundle.

Chapter 21   **G2 WebLink**   225

Describes new features of G2 WebLink, which ships with the G2 Bundle.

G2 ActiveXLink

Describes changes in G2 ActiveXLink, which ship with the G2 Bundle.

gensym

Introduction

This chapter describes G2 ActiveXLink Version 8.3 Rev. 1, which ships with the G2 Bundle Version 8.3 Rev. 1.

Note   Beginning with the G2 Bundle Version 8.1 Rev. 1, G2 ActiveXLink (`g2com.kb`) is located in the `g2\kbs\utils` directory on Windows and `g2/kbs/utils` on UNIX. Also, the G2 ActiveXLink demo KB (`axldemo.kb`) is located in the `g2\kbs\demos` directory on Windows and `g2/kbs/demos` on UNIX.

Information on bug fixes and known bugs is available separately as Technical Bulletins through HelpLink at `www.gensym-support.com`. For information on bugs in G2 ActiveXLink Version 8.3 Rev. 1, choose Technical Bulletins for G2 ActiveXLink, specify Starting From Version as 8.3 Rev. 0 and Finishing At Version as 8.3 Rev. 1, choose the bugs and suggestions that either Remain Open or Were Fixed, then click Execute.

G2 ActiveXLink Version 8.3 Rev. 1

G2 ActiveXLink Version 8.3 Rev. 1 is a compatibility release only. It has no new features or changes. For information on bug fixes, see the Introduction.

G2 ActiveXLink Version 8.3 Rev. 0

G2 ActiveXLink Version 8.3 Rev. 0 is a compatibility release only. It has no new features or changes. For information on bug fixes, see the Introduction.

Upgrading Existing Applications

Registry Problems

When upgrading existing applications, remnants of old installed and/or uninstalled versions of G2 ActiveXLink in the registry may interfere with registering the AXL control, running some demos, and adding the control to a container application. You might receive errors performing these tasks:

- VB demo: (Line 102: Property OleObjectBlob in G2Gateway1 could not be set. Line 102: Cannot load control G2Gateway1)

- Excel VB application: Can't find project library

- Inserting a new G2Gateway: Cannot insert object

- IE demo

- WorkspaceView demo (which relies on AXL)

There is a cleaning utility on the Gensym FTP server that will clean out all versions of G2 ActiveXLink in `ftp.gensym.com/pub/gensym/products/axl/utility/AxlRmv.exe`. The Analyze button shows you what the program found in the registry; it is normal not to find everything. The Clean button deletes G2 ActiveXLink from the registry. Cleaning the registry and then re-registering the control should solve the problem.

The name of the ftp machine is: `ftp.gensym.com`. Contact Gensym Customer Support for the IP address, user name, and password.

You should not have a problem if you are starting with G2 ActiveXLink Version 1.1 Rev. 1 or later and you unregister one control before registering the next version by either choosing Start > Programs > Gensym G2 8.3r1 > Unregister Active X Controls or by entering the following command in a DOS window: `regsvr32 /u g2com.dll`

It is most likely to be a problem if you have used G2 ActiveXLink Version 1.0 Rev. 9, Version 1.0 Rev. 10, and Version 1.1 Rev. 0.

G2Gateway

When you upgrade to G2 ActiveXLink Version 8.3 Rev. 1 and then load an application you developed with a prior version into a development environment such as Visual Basic, your development program may report that it could not set an `OleObjectBlob`. The reason for the error is that the `G2Gateway` object has two new properties. When your development environment tries to find the prior settings of `DisconnectOnReset` and `G2Symbols`, it fails because they did not previously exist.

Caution When your development environment reports the error, it deletes the `G2Gateway` objects. Thus, before upgrading to G2 ActiveXLink Version 8.3 Rev. 1 and loading your existing application, note the configuration of the `G2Gateway` objects in your existing installation, then backup your application.

Once you have backed up your application, upgrade to G2 ActiveXLink Version 8.3 Rev. 1 and load your application. Add the `G2Gateway` controls back and configure them as before, configure the two new properties as desired, then save the application. The application will now run correctly when you load it.

G2 CORBALink

Describes new features of G2 CORBALink, which ships with the G2 Bundle.

Introduction

This chapter describes G2 CORBALink Version 8.3 Rev. 1, which ships with the G2 Bundle Version 8.3 Rev. 1.

Information on bug fixes and known bugs is available separately as Technical Bulletins through HelpLink at `www.gensym-support.com`. For information on bugs in G2 CORBALink Version 8.3 Rev. 1, choose Technical Bulletins for G2 CORBALink, specify Starting From Version as 8.3 Rev. 0 and Ending At Version as 8.3 Rev. 1, choose the bugs and suggestions that either Remain Open or Were Fixed, then click Execute.

G2 CORBALink Version 8.3 Rev. 1

G2 CORBALink Version 8.3 Rev. 1 is a compatibility release only. It has no new features or changes. For information on bug fixes, see the Introduction.

G2 CORBALink Version 8.3 Rev. 0

G2 CORBALink Version 8.3 Rev. 0 is a compatibility release only. It has no new features or changes. For information on bug fixes, see the Introduction.

G2 Database Bridges

Describes new features and changes in the G2 Database Bridges, which ship with the G2 Bundle.

gensym

Introduction

This chapter describes the G2 Database Bridges Version 8.3 Rev. 1, which ship with the G2 Bundle Version 8.3 Rev. 1.

Information on bug fixes and known bugs is available separately as Technical Bulletins through HelpLink at `www.gensym-support.com`. For information on bugs in the G2-Database Bridges Version 8.3 Rev. 1, choose Technical Bulletins for G2-ODBC Bridge, G2-Sybase Bridge, or G2-Oracle Bridge and specify Starting From Version as 8.3 Rev. 0 and Ending At Version as 8.3 Rev. 1, choose the bugs and suggestions that either Remain Open or Were Fixed, then click Execute.

G2 Database Bridges Version 8.3 Rev. 1

The changes made to the G2-Database Bridges Version 8.3 Rev. 1 involved modifications to `g2-database.kb`. As a result, you must upgrade `g2-database.kb` to use the G2-ODBC Bridge Version 8.3 Rev. 1.

G2 Database Bridges Version 8.3 Rev. 0

Upgrading g2-database.kb

The changes made to the G2-Database Bridges Version 8.3 Rev. 0 involved modifications to both the bridge executables and `g2-database.kb`. As a result, you must upgrade `g2-database.kb` to use the G2-ODBC Bridge Version 8.3 Rev. 0.

Required Modules for G2-Database Bridges

The G2-Database Bridges now require `sys-mod.kb` and `uilroot.kb`.

"Smart Fetch"

Previously, to avoid G2 integer overflow problems when the retrieved value was too large, the database bridge "fetch" procedures (see below) returned floats under these conditions:

- When fetching DECIMAL types when:
    - The precision is 9 or greater.
    - The precision is 0.
    - The scale is non-zero.
- When any of the returned INTEGER values would cause integer overflow in G2.

Note that in Oracle, the INTEGER type is defined as DECIMAL(38,0); thus, the Oracle INTEGER type always returned as a float.

You can now use the "smart fetch" feature to better handle the case when fetching INTEGER or DECIMAL types might cause integer overflow in G2. In general, when smart fetch is not used, the procedures behave as they did previously, with a couple exceptions noted below.

To use smart fetch, add –F to the initialization string of any interface object that should use the smart fetch behavior. When smart fetch is enabled, the fetch procedures behave as follows:

- db-fetch-structure

  When retrieving INTEGER or DECIMAL types from the database and "smart fetch" is enabled, this procedure returns values as integers, or as floats if returning them as integers would cause G2 integer overflow.

- db-fetch-query-item

  When retrieving INTEGER or DECIMAL types and "smart fetch" is enabled, this procedure returns the corresponding values as an array or list of quantities, where individual values are returned as integers, or as floats if returning them as integers would have caused an overflow.

- db-fetch-object (*return-format* = single)

  When retrieving INTEGER or DECIMAL types and "smart fetch" is enabled:

  – When an attribute of *user-object* is an integer, this procedure returns the corresponding value as an integer, or it returns the maximum or minimum G2 integer (which have values 536,870,911 and -536,870,912, respectively) and generates an error if the actual value would have caused an overflow.

  – When an attribute of *user-object* is a float, this procedure returns the corresponding value as a float.

  – When an attribute of *user-object* is a quantity or value, this procedure returns the corresponding value as an integer, or as a float if returning it as an integer would have caused an overflow.

  When retrieving INTEGERs or DECIMALs, this procedure detects incompatible types and reports an error.

- db-fetch-object (*return-format* = list)

  When retrieving INTEGER or DECIMAL types:

  - When an attribute of *user-object* is an integer, returns the corresponding value as an integer, or returns maximum or minimum integer and generates an error if returning it as an integer would have caused G2 integer overflow.

  - When an attribute of *user-object* is a float, returns the corresponding value as a float.

  - When an attribute of *user-object* is a quantity, returns the corresponding value as an integer, or as a float if returning it as an integer would have caused G2 integer overflow.

  When retrieving INTEGERs or DECIMALs, this procedure now detects incompatible types and reports an error.

- db-fetch-records

    When retrieving INTEGER or DECIMAL types and "smart fetch" is enabled:

    – When an attribute of *user-object* is an integer, the corresponding value is returned as an integer, or, if returning it as an integer would cause G2 integer overflow, the maximum or minimum valid G2 integer (which have values 536,870,911 and -536,870,912, respectively) is stored in the object and an error is returned by the procedure call.

    – When an attribute of *user-object* is a float, the corresponding value is stored as a float.

    – When an attribute of *user-object* is a quantity, the value is stored as an integer or, if it will not fit in a G2 integer, as a float.

    When "smart fetch" is not enabled:

    – When the attribute of *user-object* is an integer, an error is generated if the value would cause G2 integer overflow.

    – When the attribute type of *user-object* is incompatible with the value being returned, an error is generated.

    When retrieving INTEGERs or DECIMALs, this procedure detects incompatible types and reports an error.

- db-update-object

    When retrieving INTEGER or DECIMAL types and "smart fetch" is enabled:

    – When an attribute of *user-object* is an integer, the corresponding values are returned as integers. If any value is too large or small to fit in a G2 integer (the upper and lower limits are 536,870,911 and -536,870,912, respectively), the procedure call returns an error.

    – When an attribute of *user-object* is a float, the value is returned as a float.

    – When an attribute of *user-object* is a quantity, the corresponding values that will fit in a G2 integer are returned as integers; those that will not are returned as floats.

    When "smart fetch" is not enabled, when attribute of *user-object* is an integer, an error is generated if any value would cause G2 integer overflow.

    When no records are added to the *user-object* due to an error, the *rows-processed* return value is 0 and the *cursor-position* return value is not updated.

- db-update-query-item

When retrieving INTEGER or DECIMAL types:

– When *update-action* = replace, the values are returned as an array or list of quantities, where individual values are returned as integers, or as floats if they would cause G2 integer overflow.

– When *update-action* = append

- When an attribute of the *query-item* is an integer, the corresponding values are returned as integers. If any value is too large or small to fit in a G2 integer, that value is returned as the maximum or minimum valid G2 integer (which have values 536,870,911 and -536,870,912, respectively) and an error is returned by the procedure call.

- When an attribute of the *query-item* is a float, the values are returned as floats.

- When attribute of *query-item* is a quantity, values that will fit in a G2 integer are returned as integers; those that will not are returned as floats.

This procedure:

- Requires the *query-item* to have the same number of columns as the query in the cursor and the column names to match and be in the same order.

- Generates an error if there is an incompatibility between a value returned by the query and the type of the array or list that should hold it.

G2-Oracle Bridge

The G2-Oracle Bridge supports the Oracle 10g server. The new bridge executable is g2-ora10. You must use the Oracle 10g Client software with g2-ora10. The g2gora10 bridge is supported on all platforms.

The g2-ora92 bridge is now supported on the IBM AIX platform.

G2-ODBC Bridge

DNS Configuration via an RPC

Before the G2-ODBC Bridge can connect to a database, you must define a user DSN (data source name) on the computer on which g2-odbc is running. Typically, you do this by configuring Data Sources (ODBC), which you access from the Administrative Tools icon in the control panel:
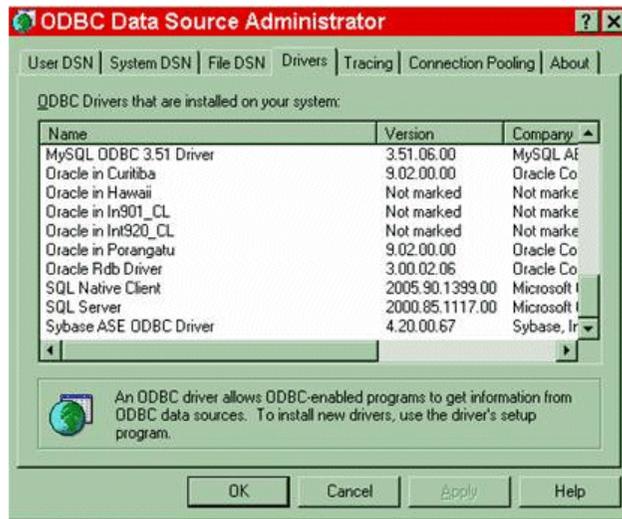
When you click the Add button, the ODBC Data Source Administrator (ODSA) presents a list of ODBC drivers that have been registered on your computer. After you select one of the drivers and click the Finish button, the ODSA passes control to the driver. It asks for the information it needs to configure the DSN that it is creating. The information depends upon which driver you select. The requirements of the database and the authors of the driver decide what information is required.

When you are finished providing the requested information and click the button to create the new DSN, the ODBC driver passes a specially formatted string to an ODBC function, causing the configuration information to be recorded for access by ODBC clients, including g2-odbc. The string consists of a sequence of key/value pairs. The required keys depend on which ODBC driver you select.

For users of G2 who need the ability to create DSNs programmatically and who have the information about which key/value pairs are required for the ODBC driver they will use, the G2-ODBC Bridge provides system procedures for performing the last step of DSN creation.

Note   To use this feature, you must contact the authors of the ODBC driver that you are using for the required key/value pairs. Gensym has no way of knowing what key/value pairs are required for any particular ODBC driver.

db-add-dsn
(*odbc-driver-name*: text, *system-dsn*: truth-value, *configuration*: structure, *interface*: g2-database-interface)

Adds a DSN to the computer on which g2-odbc is running.

db-modify-dsn
> (*odbc-driver-name*: text, *system-dsn*: truth-value, *configuration*: structure, *interface*: g2-database-interface)

> Modifies an existing DSN on the computer on which g2-odbc is running.

db-delete-dsn
> (*odbc-driver-name*: text, *system-dsn*: truth-value, *dsn-name*: text, *interface*: g2-database-interface)

> Deletes an existing DSN on the computer on which g2-odbc is running.

Here are the arguments to these procedures:

Argument	Description
odbc-driver-name	The name you would see if you looked for the driver in the ODBC Data Source Administrator (Control Panel > Administrative Tools > Data Sources (ODBC) > Drivers). For example, to create a DSN for MS SQL Server, you would use the driver name SQL Server, which is toward the end of the list in the table on the Drivers tab.
system-dsn	Set to true if you are manipulating a system DSN; otherwise, set it to false. A system DSN is shared by all users, whereas a user DSN is user-specific.
configuration	The key/value pairs required by the particular ODBC driver you are adding or modifying (db-add-dsn and db-mod-dsn only).
	To add or modify a DSN, specify dsn: *name* pair in the structure.
dsn-name	The DSN name to delete (db-del-dsn only).
interface	The g2-database-interface that is connected to the G2-ODBC Bridge

Stored Procedures Returning Record Sets

Stored procedures in MS SQL Server can return record sets. The G2-ODBC Bridge can now retrieve a record set by executing a stored procedure. Once you have a cursor to the record set, you can use all the different forms of the db-fetch- procedures on the cursor to retrieve data from the record set.

To support this feature, the second parameter to db-define-cursor now accepts EXEC statements, as well as SELECT statements. The EXEC statement is used to start a stored procedure. If a cursor contains an EXEC statement and the stored

procedure returns a record set, fetching on the cursor behaves just as it would if the cursor had contained an equivalent SELECT statement.

G2 Gateway

Describes the new features and changes to G2 Gateway.

Introduction

This chapter describes the G2 Gateway Version 8.3 Rev. 1, which ships with the G2 Bundle Version 8.3 Rev. 1.

Information on bug fixes and known bugs is available separately as Technical Bulletins through HelpLink at `www.gensym-support.com`. For information on bugs in G2 Gateway Version 8.3 Rev. 1, choose Technical Bulletins for G2 Gateway, specify Starting From Version as 8.3 Rev. 1 and Ending At Version as 8.3 Rev. 1, choose the bugs and suggestions that either Remain Open or Were Fixed, then click Execute.

G2 Gateway Version 8.3 Rev. 1

On Linux, the G2 Bundle ships with an additional G2 Gateway directory called `gsi-redhat4`, which contains the GSI libraries and examples compiled with Redhat Enterprise Linux 4 with kernel 2.6.9-55.EL. For more information on support for this version on Linux, see "Linux" on page 44 in Chapter 3, "G2 Version 8.3 Rev. 1."

As of G2 Version 8.3 Rev. 1, G2, Telewindows, G2 Gateway, and G2 JavaLink are all supported on the HP Itanium platform under HP-UX 11.23. Other bridges and layered products have been compiled and linked on a PA-RISC machine running HP-UX 11.00 and run in compatibility mode on the HP Itanium machine under HP-UX 11.23. Thus, the `gsi-itanium` directory no longer exists.

Otherwise, G2 Gateway Version 8.3 Rev. 1 is a compatibility release only. It has no new features or changes. For information on bug fixes, see the Introduction.

G2 Gateway Version 8.3 Rev. 0

Support for Secure Communication

G2 Gateway optionally supports the `-secure` and `-cert` command-line options, which are available for G2. See "Support for Secure Communication on UNIX" on page 152.

To access SSL, you need to include the following new libraries, depending on your platform:

UNIX	Windows
libgsec.a	libgsec.lib

If you do not want to use SSL, you need to include the following new libraries instead:

UNIX	Windows
libnogsec.a	libnogsec.lib

Failure to include one of these libraries or attempts to include both results in link errors.

In addition, you must also provide the following platform-specific libraries:

- Windows: `crypt32.lib`, available with your Microsoft compiler.

- Solaris, Linux, HP-UX, IBM AIX: `libssl.a` and `libcrypto.a`, which are supplied with G2 Gateway. Note that you must supply these two libraries in exactly this order; failure to do so will result in link errors.

- HP-UX: You must also include `libgcc.a`, also provided with G2 Gateway.

On the Windows platforms, the default `gsi.dll` is linked without SSL support; a separate library `gsi_ssl.dll` is provided to include SSL support as a DLL.

Currently, G2 Gateway does not support SSL on the alphaosf platform, but `libnogsec.a` must be linked in anyway. The example is not present.

The example `makefile` for G2 Gateway compiles most of the examples without SSL support. The `skeleton_ssl` example includes SSL support.

Attempting to give the `-secure` option to a G2 Gateway bridge that has not been linked with SSL support results in a warning message; however, the bridge will start up normally, but without SSL support.

Upon startup, a bridge gives the port number with /SSL appended when `-secure` is requested and available. For example:

```
GSI Version 8.3 Rev. 0 IBM POWERstation (JA28)
2007-01-30 15:00:05   Waiting to accept a connection on:
2007-01-30 15:00:05   TCP_IP:cs-aix4:22000/SSL
```

To establish a secure connection and test the secure status, use these procedures:

```
gsi_int gsi_establish_secure_listener
    (network, port, exact, certificate)
```

Attempts to establish a listener, using the SSL protocol. The first three arguments are exactly like `gsi_establish_listener`, and `certificate` is a `gsi_char*` containing the name of the certificate, which is a name in the certificate store on Windows or a filename on UNIX. The certificate can be NULL. Note that if a request is received from a clear text (insecure) connection, it is accepted as clear text. To check whether the connection is secure, use `gsi_current_context_is_secure` or `gsi_context_is_secure`.

```
gsi_int gsi_initiate_secure_connection
    (interface_name, class_name, keep_connection, network, host,port,
    rpis)
```

Behaves exactly like `gsi_initiate_connection`, but tries to make a secure connection with SSL to G2.

```
gsi_int gsi_initiate_secure_connection_with_user_data
    (interface_name, class_name, keep_connection, network, host,
    port, rpis, context_user_data)
```

Behaves exactly like `gsi_initiate_connection_with_user_data`, but tries to make a secure connection with SSL to G2.

Note that if G2 is not listening for secure connections, this connection fails and G2 Gateway becomes inoperative. We recommend that you determine whether G2 is listening securely before executing either of these procedures.

```
gsi_int gsi_current_context_is_secure()
```

Returns the security status of the current context: 0 (insecure) or 1 (secure).

```
gsi_int gsi_context_is_secure(context)
```

> Returns the security status of the specified context (a `gsi_int`): 0 (insecure) or 1 (secure).

To establish a GSI connection with security, use the **secure yes** option in the **gsi-connection-configuration** attribute, after the host and port number. For example:

> tcp-ip host "localhost" port-number 22044 secure yes

For G2-G2 connections, use the **icp-connection-specification** attribute.

Specifying the **secure yes** option attempts to make a secure connection to the port number on the specified host. Note that if the host is not listening for secure connections on the specified port, this connection fails and G2 becomes inoperative. If no host is listening at the port, then the connection simply fails.

In addition, the **gsi-interface** class defines the **gsi-interface-is-secure** attribute, and the **g2-to-g2-interface** class defines the **interface-is-secure** attribute, whose value is **yes** or **no**, which determines whether or not security was established on the connection from the remote system.

Note that you cannot make a secure G2-to-G2 connection to the same G2. This condition is detected, and an insecure connection is created instead, with a warning on the logbook.

G2 Gateway and Visual Studio 2005

On Windows platforms, the G2 Bundle ships with two G2 Gateway directories, one called `gsi-intc`, which contains the GSI libraries and examples compiled with the Intel compiler, and the other called `gsi-msvc`, which contains the GSI libraries and examples compiled with Microsoft Visual Studio. Previously, the bundle shipped with one directory called `gsi`, which was compiled with the Intel compiler.

Otherwise, G2 Gateway Version 8.3 Rev. 0 is a compatibility release only. It has no new features or changes. For information on bug fixes, see the Introduction.

G2 Gateway on HP-UX Itanium

Note   This change has been superseded in G2 Gateway Version 8.3 Rev. 1.

G2 Gateway Version 8.3 provides a separate component called G2 Gateway (GSI) for HP-UX Itanium, which is available on HP platforms only. This component installs into the `gsi-itanium` directory in the G2 bundle installation directory, a parallel directory to the `gsi` directory. The GSI libraries and examples in the `gsi-itanium` directory have been compiled on an HP Itanium machine running HP-UX 11.23, and are in the native ELF format. All other files in the G2 bundle on

the HP platform are compiled and linked on a PA-RISC machine running HP-UX 11.00 and will run in compatibility mode on the HP Itanium machine under HP-UX 11.23.

G2-HLA Bridge

Describes the G2-HLA Bridge, which ships with the G2 Bundle.

gensym

Introduction

This chapter describes the G2-HLA Bridge Version 2.0 Rev. 1, which ships with the G2 Bundle Version 8.3 Rev. 1.

This bridge was new in the G2 Bundle Version 8.3 Rev. 0.

Information on bug fixes and known bugs is available separately as Technical Bulletins through HelpLink at `www.gensym-support.com`. For information on bugs in the G2-HLA Bridge Version 2.0 Rev. 1, choose Technical Bulletins for G2-HLA Bridge, specify Starting From Version as 2.0 Rev. 0 and Ending At Version as 2.0 Rev. 1, choose the bugs and suggestions that either Remain Open or Were Fixed, then click Execute.

G2-HLA Bridge Version 2.0 Rev. 1

The G2-HLA Bridge Version 2.3 Rev. 1 is a compatibility release only. It has no new features or changes. For information on bug fixes, see the Introduction.

G2-HLA Bridge Version 2.0 Rev. 0

The formal definition of the Modeling and Simulation (M & S) High Level Architecture (HLA) comprises three main components: the HLA rules, the HLA interface specification, and the HLA object model template (OMT). It provides a language independent specification (LIS) and multiple language bindings to support inter-simulation communication in a distributed simulation domain. To learn more about HLA, visit the HLA web site.

The High Level Architecture (HLA) is an integrated architecture that has been developed to provide a common architecture for M & S. The HLA requires that inter-federate interactions use a standard API. The specification defines the standard services and interfaces to be used by the federates in order to support efficient information exchange when participating in a distributed federation execution and reuse of the individual federates.

The RTI provides services to federates in a way that is analogous to how a distributed operating system provides services to applications. These interfaces are arranged into six basic RTI service groups:

- Federation management.

- Declaration management.

- Object management.

- Ownership management.

- Time management.

- Data distribution management.

The six service groups describe the interface between the federates and the RTI, and the software services provided by the RTI for use by HLA federates. The initial set of these services was carefully chosen to provide those functions most likely to be required across multiple federations. As a result, federate applications require most of the services described. The RTI requires a set of services from the federate that are referred to as "RTI callbacks methods."

The G2-HLA Bridge provides the communicate interface between the RTI and G2. It provides methods and callback methods to any G2 application. The definition and basic implementation of these methods is provided in the `ghla.kb` module. This module requires the `sys-mod.kb` and `uilroot.kb` modules. You should merge the `ghla.kb` module into your application and make it a required module.

Before using this package, you should familiarize yourself with HLA and RTI. You should review the documentation available from the HLA web site. In additional and since this implementation is based on the C++ libraries of the RTI, the documentation of these libraries available from the HLA web site is a good resource for additional information.

For details, see the *G2-HLA Bridge User's Guide*.

G2 JavaLink

Describes new features and changes in G2 JavaLink, which ship with the G2 Bundle.

Introduction

This chapter describes G2 JavaLink Version 8.3 Rev. 1, which ships with the G2 Bundle Version 8.3 Rev. 1.

Information on bug fixes and known bugs is available separately as Technical Bulletins through HelpLink at `www.gensym-support.com`. For information on bugs in G2 JavaLink Version 8.3 Rev. 1, choose Technical Bulletins for G2 JavaLink, specify Starting From Version as 8.3 Rev. 0 and Ending At Version as 8.3 Rev. 1, choose the bugs and suggestions that either Remain Open or Were Fixed, then click Execute.

For additional information, see `readme-javalink.html`. For information on the directory location of this file, see "Online Documentation" on page 26.

G2 JavaLink Version 8.3 Rev. 1

G2 JavaLink Version 8.3 Rev. 1 is a compatibility release only. It has no new features or changes. For information on bug fixes, see the Introduction.

G2 JavaLink Version 8.3 Rev. 0

G2 JavaLink supports secure communication on Windows and Linux platforms, using SSPI and OpenSSL, respectively. Support for OpenSSL on other UNIX platforms will be available in a future release.

To support secure communication, G2 JavaLink provides the following new constructors:

```
G2ConnectionInfo
 (String hostName, String portName, boolean secureRequested)
```

```
G2ConnectionInfo
 (String brokerURL, String logicalName, String hostName,
 String portName, String connectionClassName,
 String gsiInterfaceClassName, String interfaceName,
 String protocol, Boolean isPerm, String rpis, boolean forceNew,
 boolean sharable, boolean secure)
```

To establish a secure listener within G2 JavaLink, use the `-secure` and `-cert` G2 Gateway command-line arguments in `G2Gateway.initialize`. For example:

```
G2Gateway.initialize(new String[] {"-listenerport", "22044",
    "-secure", "-cert", "mycert.pem"});
```

The following method returns whether the current context is secure.

`G2Gateway.getSecureContext()`

G2 Java Mail Bridge

Describes new features of the G2 JMail Bridge, which ships with the G2 Bundle.

gensym

Introduction

This chapter describes the G2 Java Mail Bridge Version 2.3 Rev. 1, which ships with the G2 Bundle Version 8.3 Rev. 1.

Information on bug fixes and known bugs is available separately as Technical Bulletins through HelpLink at `www.gensym-support.com`. For information on bugs in the G2 Java Mail Bridge Version 2.3 Rev. 1, choose Technical Bulletins for G2 Java Mail Bridge, specify Starting From Version as 2.3 Rev. 0 and Ending At Version as 2.3 Rev. 1, choose the bugs and suggestions that either Remain Open or Were Fixed, then click Execute.

G2 Java Mail Bridge Version 2.3 Rev. 1

The G2 Java Mail Bridge Version 8.3 Rev. 1 is a compatibility release only. It has no new features or changes. For information on bug fixes, see the Introduction.

G2 Java Mail Bridge Version 2.3 Rev. 0

Pinging the Bridge

You can ping the G2 Java Mail Bridge to ensure that it is connected, using this API procedure:

> jmail-ping
> > (*io*: class jmail-interface)
> > —> *status*: truth-value

Changing the SNMP Port Number

You can change the outgoing and incoming port numbers for sending email, using one of these techniques:

- Edit the following attributes on jmail-interface:

  - outgoing-email-host-port — The port number of the outgoing mail host. Default is 25.

  - incoming-email-host-port — The port number of the incoming mail host. Default is 25.

- When getting email, use this API procedure:

  jmail-get-mail
  > (*io*: class jmail-interface, *protocol*: text, *host*: text, *host-port*: integer, *user-name*: text, *password*: text, *folder*: text, *delete-messages*: truth-value, *max-nb-of-messages*: integer, *max-message-length*: integer)
  > —> *email-structures*: sequence

- When sending email, use one of these API procedures:

  jmail-send-mail
  > (*io*: class jmail-interface, *host*: text, *host-port*: integer, *user-name*: text, *password*: text, *send-to*: sequence, *cc*: sequence, *from*: text, *subject*: text, *contents*: text)
  > —> *status*: text

  jmail-send-mime-mail
  > (*io*: class jmail-interface, *host*: text, *host-port*: integer, *user-name:* text, *password:* text, *send-to*: sequence, *cc*: sequence, *subject*: text, *from:* text, *contents*: sequence)

For a description of the arguments to these procedures, see the *G2 JMail Bridge User's Guide*.

Sending MIME Email

The G2 Java Mail Bridge Version 2.2 Rev. 2 provided several new features for sending MIME email, which were previously not documented. See the *G2 JMail Bridge User's Guide*.

Disconnecting without Killing the Bridge Process

The G2 Java Mail Bridge provides the following new method that disconnects the jmail-interface without terminating the G2 Java Mail Bridge process. The existing jmail-disconnect method terminates the connection between G2 and the G2 Java Mail Bridge by terminating the bridge process.

jmail-disconnect-without-kill
    (*io*: class jmail-interface)

Additional Documentation

The G2 Java Mail Bridge requires J2SE Version 1.4 or later.

G2 Java
Socket Manager

Describes new features of G2 Java Socket Manager, which ships with the G2 Bundle.

gensym

Introduction

This chapter describes the G2 Java Socket Manager Version 4.2 Rev. 1, which ships with the G2 Bundle Version 8.3 Rev. 1.

Note   Beginning with G2 Java Socket Manager Version 4.1 Rev. 0, the name of the KB changed from `jsockman.kb` to `gsockman.kb`. For information about upgrading KBs, see "Upgrading the G2 Java Socket Manager Module" on page 214.

Information on bug fixes and known bugs is available separately as Technical Bulletins through HelpLink at `www.gensym-support.com`. For information on bugs in the G2 Java Socket Manager Version 4.2 Rev. 1, choose Technical Bulletins for G2 Java Socket Manager, specify Starting From Version as 4.2 Rev. 0 and Ending At Version as 4.2 Rev. 1, choose the bugs and suggestions that either Remain Open or Were Fixed, then click Execute.

G2 Java Socket Manager Version 4.2 Rev. 1

G2 Java Socket Manager Version 8.3 Rev. 1 is a compatibility release only. It has no new features or changes. For information on bug fixes, see the Introduction.

G2 Java Socket Manager Version 4.2 Rev. 0

The `gsockdemo.kb` has an example of sending a file over a socket, which only sends the file if the file has a line feed or carriage return at the end. The demo KB has been updated to add this information.

Upgrading the G2 Java Socket Manager Module

Beginning with G2 Java Socket Manager Version 4.1 Rev. 0, the name of the KB changed from `jsockman.kb` to `gsockman.kb`.

If you have instances of classes defined in the jsockman module, you may need to make changes in the 4.0 version so that it loads correctly in the 4.1 version. G2 Java Socket Manager Version 4.0 shipped with the G2 Bundle Version 7.1, and G2 Java Socket Manager Version 4.1 shipped with the G2 Bundle Version 8.0.

To replace jsockman with gsockman in your application:

1   Load the 4.0 version of your G2 Java Socket Manager application into G2 Version 7.1.

2   Execute this command, using Inspect:

> replace the symbol jsockman with gsockman in the module-assignment of every kb-workspace W such that the item superior to W does not exist and assigned to module jsockman

You should have three kb-workspaces now assigned to jsockman.

3   Execute this command, using Inspect:

> replace the symbol jsockman with gsockman in the directly-required-modules of every module-information

4   Execute this command, using Inspect:

> show on a workspace the module hierarchy

5   In the jsockman module, edit the top-level-module from jsockman to gsockman.

6   Save the application, including all directly required modules.

7   Copy the application-specific modules into your G2 Java Socket Manager Version 4.2 directory within the G2 Bundle Version 8.3.

8   Edit `g2\StartServer.bat` to include `sockman\kbs` in the `G2_MODULE_SEARCH_PATH`.

9) Load the application in G2 Version 8.3.

G2 JMSLink

Describes new features of G2 JMSLink, which ships with the G2 Bundle.

gensym

Introduction

This chapter describes G2 JMSLink Version 2.3 Rev. 1, which ships with the G2 Bundle Version 8.3 Rev. 1.

Information on bug fixes and known bugs is available separately as Technical Bulletins through HelpLink at `www.gensym-support.com`. For information on bugs in G2 JMSLink Version 2.3 Rev. 1, choose Technical Bulletins for G2 JMSLink, specify Starting From Version as 2.3 Rev. 0 and Ending At Version as 2.3 Rev. 1, choose the bugs and suggestions that either Remain Open or Were Fixed, then click Execute.

G2 JMSLink Version 2.3 Rev. 1

G2 JMSLink Version 2.3 Rev. 1 is a compatibility release only. It has no new features or changes. For information on bug fixes, see the Introduction.

G2 JMSLink Version 2.3 Rev. 0

G2 JMSLink Version 2.3 Rev. 0 is a compatibility release only. It has no new features or changes. For information on bug fixes, see the Introduction.

The following information should be documented in the *G2 JMSLink User's Guide*:

G2 JMSLink requires J2SE Version 1.4 or later.

G2-OPC Client Bridge

Describes new features of the G2-OPC Client Bridge, which ships with the G2 Bundle.

Introduction

This chapter describes the G2-OPC Client Bridge Version 8.3 Rev. 1, which ships with the G2 Bundle Version 8.3 Rev. 1.

Note   The G2-OPC Client Bridge Version 8.3 Rev. 0 was a complete re-write of the G2 OPCLink Version 3.2. Therefore, the name of the bridge has changed.

Information on bug fixes and known bugs is available separately as Technical Bulletins through HelpLink at `www.gensym-support.com`. For information on bugs in the G2-OPC Client Bridge Version 8.3 Rev. 1, choose Technical Bulletins for G2-OPC Client Bridge, specify Starting From Version as 8.3 Rev. 0 and Ending At Version as 8.3 Rev. 1, choose the bugs and suggestions that either Remain Open or Were Fixed, then click Execute.

G2-OPC Client Bridge Version 8.3 Rev. 1

The G2-OPC Client Bridge Version 8.3 Rev. 1 is a compatibility release only. It has no new features or changes. For information on bug fixes, see the Introduction.

G2-OPC Client Bridge Version 8.3 Rev. 0

The G2-OPC Client Bridge Version 8.3 Rev. 0 is a complete re-write of G2 OPCLink Version 3.2. As a result, the name of the directory has been changed from `opclink` to `opcclient`.

In general, all functionality in the previous bridge is supported in the new bridge, with the following exceptions:

*   The menu that leads to the control panel and the server browser is opened by right-clicking on the icon in the notification tray instead of double-clicking it:

*   The new bridge does not provide the ability to browse other machines for OPC DA servers. This capability will be added to the next release of the G2-OPC DA Client bridge.

In addition, the G2-OPC Client Bridge defines a new error code, 308, OPC_BAD_ NOT_INITIALIZED, which represents the state in which the bridge attempts to read a value before the OPC Server has had a chance to initialize it.

G2-PI Bridge

Describes new features and changes in the G2-PI Bridge, which ships with the G2 Bundle.

Introduction

This chapter describes the G2-PI Bridge Version 8.3 Rev. 1, which ships with the G2 Bundle Version 8.3 Rev. 1.

Information on bug fixes and known bugs is available separately as Technical Bulletins through HelpLink at `www.gensym-support.com`. For information on bugs in Version 8.3 Rev. 1 of the G2-PI Bridge, choose Technical Bulletins for the bridge, specify Starting From Version as Version 8.3 Rev. 0 and Ending At Version as 8.3 Rev. 1, choose the bugs and suggestions that either Remain Open or Were Fixed, then click Execute.

G2-PI Version 8.3 Rev. 1

The G2-PI Bridge Version 8.3 Rev. 1 is a compatibility release only. It has no new features or changes. For information on bug fixes, see the Introduction.

G2-PI Version 8.3 Rev. 0

The `g2-pi.kb` no longer requires `gensym-raised-buttons.kb`. It has been reimplemented to use UIL buttons and, therefore, now requires `uilroot.kb` and `sys-mod.kb`. The use of `gensym-raised-buttons.kb` has been deprecated.

G2-SNMP Bridge

Describes the G2-SNMP Bridge, which ships with the G2 Bundle.

Introduction

This chapter describes the G2-SNMP Bridge Version 4.0 Rev. 1, which ships with the G2 Bundle Version 8.3 Rev. 1.

This bridge was new in the G2 Bundle Version 8.3 Rev. 0.

Information on bug fixes and known bugs is available separately as Technical Bulletins through HelpLink at `www.gensym-support.com`. For information on bugs in the G2-SNMP Bridge Version 4.0 Rev. 1, choose Technical Bulletins for G2- SNMP Bridge, specify Starting From Version as 4.0 Rev. 0 and Ending At Version as 4.0 Rev. 0, choose the bugs and suggestions that either Remain Open or Were Fixed, then click Execute.

G2-SNMP Bridge Version 4.0 Rev. 1

The Java-based G2-SNMP (Simple Network Management Protocol) Bridge enables a user application to communicate with devices that support SNMP. This guide describes the functionality of the bridge and explains how to install and run it.

The Java-based G2-SNMP bridge provides a set of functions to perform SNMP v2c transactions (SET, GET, GETNEXT, GETBULK, and INFORM) and to send and receive SNMP traps.

The bridge uses the G2 JavaLink, a toolkit for creating Java-based bridges between G2 and external systems.

The Java-based G2-SNMP Bridge is designed to interact with the Operations Expert SNMP (OXS) application. The user's application will in turn interface with the bridge through the OXS application.

For details, see the *G2-SNMP Bridge User's Guide*.

G2 WebLink

Describes new features of G2 WebLink, which ships with the G2 Bundle.

gensym

Introduction

This chapter describes G2 WebLink Version 8.3 Rev. 1, which ships with the G2 Bundle Version 8.3 Rev. 1.

Information on bug fixes and known bugs is available separately as Technical Bulletins through HelpLink at `www.gensym-support.com`. For information on bugs in G2 WebLink Version 8.3 Rev. 1, choose Technical Bulletins for G2 WebLink, specify Starting From Version as 8.3 Rev. 0 and Ending At Version as 8.3 Rev. 1, choose the bugs and suggestions that either Remain Open or Were Fixed, then click Execute.

G2 WebLink Version 8.3 Rev. 1

G2 WebLink Version 8.3 Rev. 1 is a compatibility release only. It has no new features or changes. For information on bug fixes, see the Introduction.

G2 WebLink Version 8.3 Rev. 0

G2 WebLink Version 8.3 Rev. 0 is a compatibility release only. It has no new features or changes. For information on bug fixes, see the Introduction.

Part VI

G2 Applications

G2 Diagnostic
Assistant (GDA)

Describes the new features and changes to G2 Diagnostic Assisant (GDA).

gensym

Introduction

This chapter describes the G2 Diagnostic Assistant (GDA) Version 5.1 Rev. 1, which ships with the G2 Bundle Version 8.3 Rev. 1.

Note   The GDA documentation has not been updated to include the changes in GDA Version 5.1 Rev. 1, Version 5.1 Rev. 0, Version 5.0 Rev. 1, or Version 5.0 Rev. 0, which shipped with the G2 Bundle Version 8.3 Rev. 1, 8.3 Rev. 0, 8.2 Rev. 1, Rev. 2, Rev. 3, and Rev. 4, respectively. Therefore, these release notes describe the changes in these releases.

Information on bug fixes and known bugs is available separately as Technical Bulletins through HelpLink at `www.gensym-support.com`. For information on bugs in GDA Version 5.1 Rev. 1, choose Technical Bulletins for GDA, specify Starting From Version as 5.1 Rev. 0 and Ending At Version as 5.1 Rev. 1, choose the bugs and suggestions that either Remain Open or Were Fixed, then click Execute.

G2 Diagnostic Assistant Version 5.1 Rev. 1

The G2 Diagnostic Assistant Version 5.1 Rev. 1 is a compatibility release only. It has no new features or changes. For information on bug fixes, see the Introduction.

G2 Diagnostic Assistant Version 5.1 Rev. 0

The three GDA demos, `gdacerel.kb`, `gdareact.kb`, and `gdatank.kb`, have been updated to add Windows charts. Also, the workspace contents for `gdacerel.kb` and `gdatank.kb` have changed.

G2 Diagnostic Assistant Version 5.0 Rev. 1

G2 Diagnostic Assistant Version 5.0 Rev. 1, which shipped with the G2 Bundle Version 8.2 Rev. 2, Rev. 3, and Rev. 4 has various changes to address several issues in GDA Version 5.0 Rev. 0. Some of these changes were also present in GDA Version 5.0 Rev. 0.

User Interface

GDA Version 5.0 Rev. 0 introduced Windows dialogs for all items when using Telewindows Next Generation (`twng.exe`) or Telewindows (`tw.exe`). However, many of the GDA Windows dialogs use features that are only supported by Telewindows Next Generation. Therefore, GDA Version 5.0 Rev. 1 makes these changes with respect to the user interface:

- Uses GUIDE/UIL dialogs for the following items in Telewindows Next Generation:

    - The default dialog for custom blocks.

    - The dialog for the multi-state observation block.

- Uses classic G2 dialogs in Telewindows (`tw.exe`).

System Table Changes

GDA Version 5.0 Rev. 1 and Rev. 0 have the following system table attribute changes:

System Table	Attribute	GDA Version 4.5	GDA Version 5.0
Timing	meter-lag-time	10 seconds	0 seconds
Message Board	prefer-native-message-board	no	yes
Logbook	prefer-native-logbook	no	yes
Miscellaneous	start-kb-after-load	no	yes
	show-uuids-in-attribute-tables	no	yes
	backward-compatibility-features	Does not include ignore duplicate list element errors	Includes ignore duplicate list element errors

For GDA users who also use GRTL, note these differences in the Timing Parameters system table:

Attribute	GDA	GRTL
minimum-scheduling-interval	0.05 seconds	0.002 seconds
attribute-display-update-interval	continuous	0.5 seconds
interface-mode	interruptible interface service	always service interface first

Additional Changes

GDA Version 5.0 Rev. 0 and Rev. 1 have the following additional changes:

- Sets the G2 user mode of the window to developer.

- Suppresses mouse tracking using grtl-track-mouse-over-message over uil-edit-boxes.

- Suppresses the GRTL popup menu user menu choice for the following classes:
    - connection
    - default-junction
    - free-text
    - borderless-free-text
    - kb-workspace
    - network-interface
    - gqs-queue-access-table
    - gqsv-tabular-view-template

Note To avoid extraneous user menu choices on GDA objects, we recommend that GDA users set the enable-menus-and-toolbars-upon-startup of the active grtl-module-settings to false.

Adding Comments to Queue Entries

In the detail dialog of an entry for any queue, you must click the Add Comment button each time you add a comment; otherwise, the comments will not be added correctly.

Adding Attributes to Custom Blocks

When adding attributes to a custom block through the wizard, you must click the Apply button each time you create a new attribute; otherwise, the attributes will not be added correctly.

Finding Blocks

When searching for blocks by choosing Help > Find Block, you must click the Apply button after entering the keyword to execute the search.

View Templates

When creating custom view templates or modifying the existing built-in queues, you should now use the gevm-native-view-manager-template, not the gqmv-tabular-view templates. Each of the four Gensym-supplied queues has on its workspace a gevm-native-view-manager-template, which you can clone and modify for this purpose.

G2 Diagnostic Assistant Version 5.0 Rev. 0

G2 Diagnostic Assistant Version 5.0 Rev. 0, which shipped with the G2 Bundle Version 8.2 Rev. 1, provides a Windows user interface for these aspects of GDA:

- Windows dialogs for all GDA blocks.

- Windows dialogs for configuring all preferences.

- Windows popups on all items.

- Windows browsers for Alarms, Error, Explanation, and Message queues.

- New Custom Class Wizard.

Here are two examples of the new dialogs for GDA blocks:
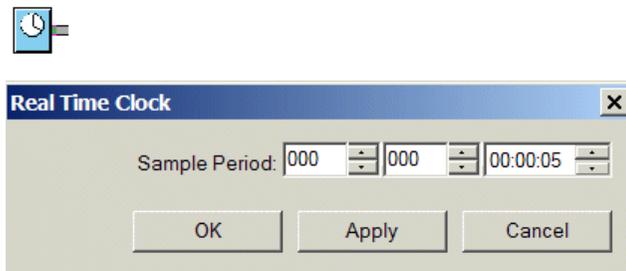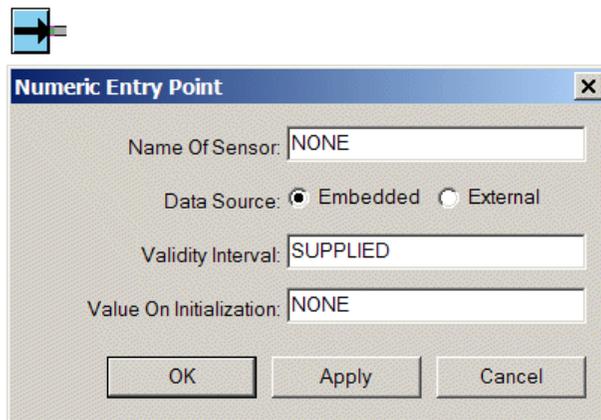
Here is an example of a dialog for configuring color properties:
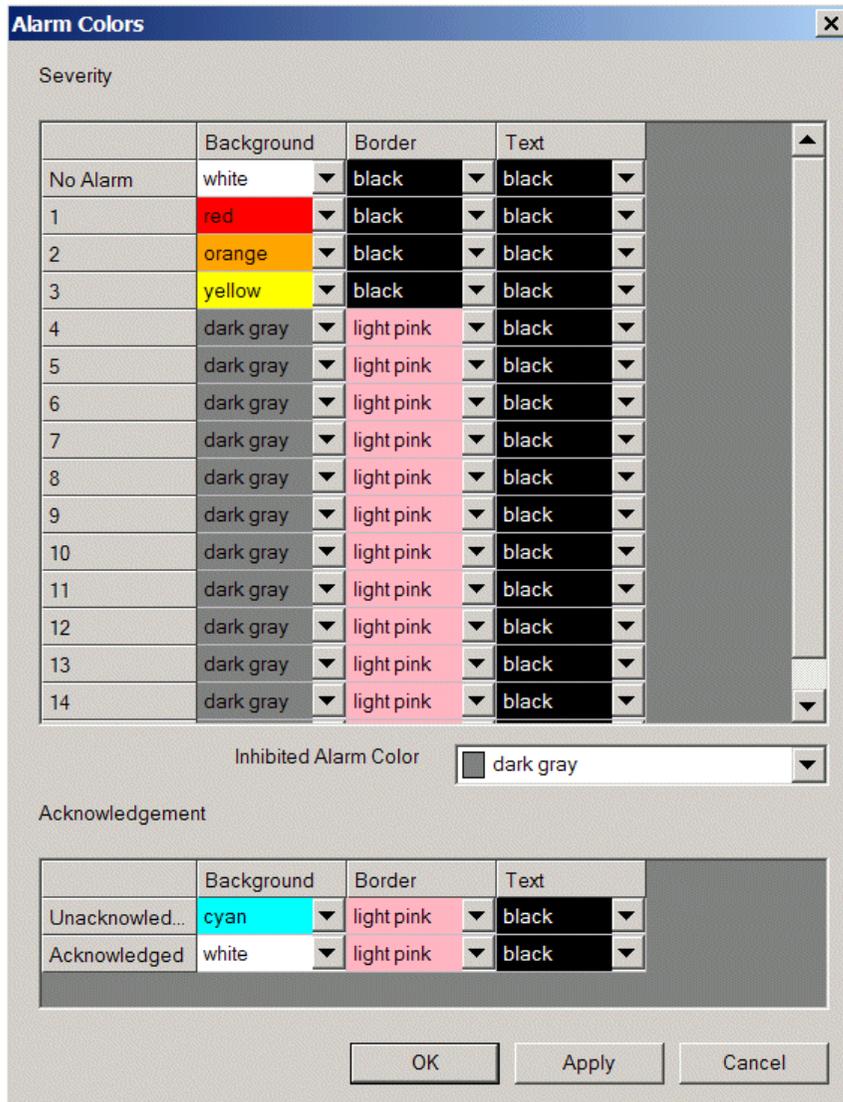
Here is an example of the custom class wizard:
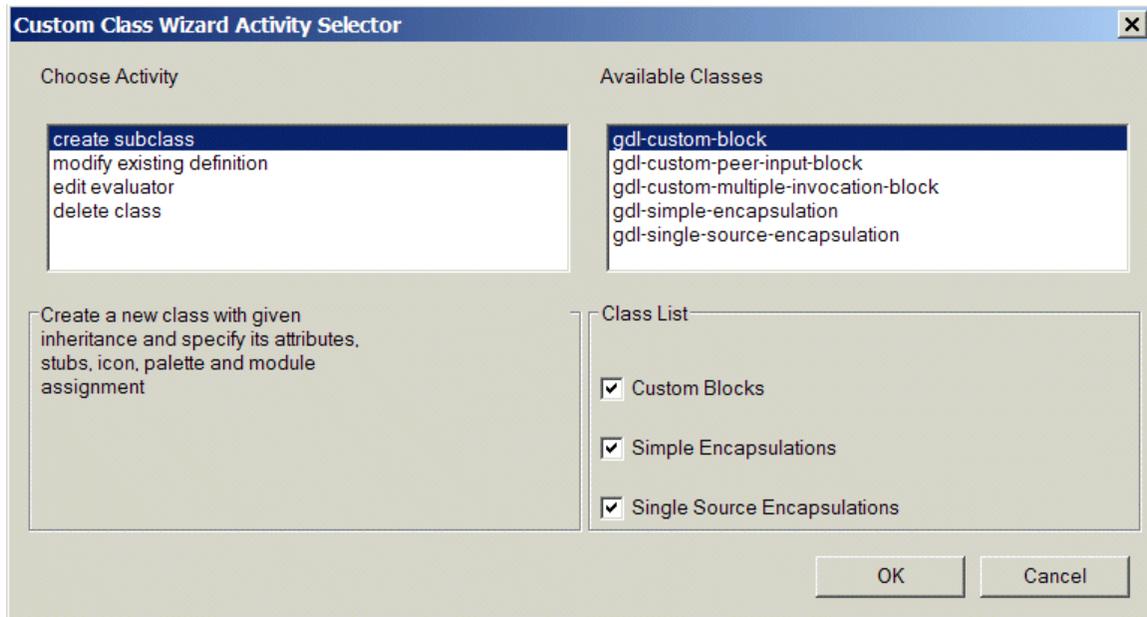
See "G2 Diagnostic Assistant Version 5.0 Rev. 1" on page 230 for information on system table changes.

G2 e-SCOR

Describes the new features and changes to G2 e-SCOR.

gensym

Introduction

This chapter describes G2 e-SCOR Version 5.1 Rev. 1, which ships with the G2 Bundle Version 8.3 Rev. 1.

G2 e-SCOR is available to licensed users. For further information on G2 e-SCOR, visit `http://www.gensym.com/` or contact Gensym at (781) 265-7100.

Information on bug fixes and known bugs is available separately as Technical Bulletins through HelpLink at `www.gensym-support.com`. For information on bugs in G2 e-SCOR Version 5.1 Rev. 1, choose Technical Bulletins for e-SCOR, specify Starting From Version as 5.1 Rev. 0 and Ending At Version as 5.1 Rev. 1, choose the bugs and suggestions that either Remain Open or Were Fixed, then click Execute.

G2 e-SCOR Version 5.1 Rev. 1

G2 e-SCOR Version 5.1 Rev. 1 is a compatibility release only. It has no new features or changes. For information on bug fixes, see the Introduction.

G2 e-SCOR Version 5.1 Rev. 0

G2 e-SCOR Version 5.1 Rev. 0 has the following new features and changes:

- G2 e-SCOR ships as part of the G2 bundle; therefore, the organization of the Start menu and the directory structure of the installation have changed to make e-SCOR a component of the G2 bundle, rather than vice versa.

  For example, to start G2 e-SCOR, you choose Start > Programs > Gensym G2 8.3r1 > G2 e-SCOR > Start G2 e-SCOR Server.

  Also, the escor directory is a subdirectory of the g2-8.3r1 directory.

- Windows user interface provided by Telewindows Next Generation in place of the e-SCOR Java client. This user interface is based on the G2 Run-Time Library and includes menu choices and features that are similar across all Gensym application products.

- Additional supply chain functionality, including:

  – Q and R-Q planning strategies.

  – Generic product specifications.

  – By-products and alternate products.

  – Support for multiple suppliers based on enhanced contract functionality.

  – Definition and Best Practice attributes for roles and categories.

- Prototype for returns, which includes source and deliver defective products, excess inventory, and maintenance, repair, and overhaul (MRO) products. Note that although e-SCOR Version 5.1 Rev. 0 provides blocks for returns, these blocks are not fully implemented. They will be implemented in a future release.

- Ability to create supply chains without connecting roles.

- Scenario Manager keywords no longer have "-parameter" on the names of all the keywords.

- New charting and reporting capabilities, based on the G2 Reporting Engine (GRPE) module.

- The logistics lookup report has been removed. This functionality will be provided in a future release.

- Fairly large performance improvements.

- Additional customization support.

- More detailed metrics, including time series and various other statistics.

G2 ReThink

Describes the new features and changes to G2 ReThink.

Introduction

This chapter describes G2 ReThink Version 5.1 Rev. 1, which ships with the G2 Bundle Version 8.3 Rev. 1.

G2 ReThink is available to licensed users. For further information on G2 ReThink, visit `http://www.gensym.com/` or contact Gensym at (781) 265-7100.

Information on bug fixes and known bugs is available separately as Technical Bulletins through HelpLink at `www.gensym-support.com`. For information on bugs in G2 ReThink Version 5.1 Rev. 1, choose Technical Bulletins for ReThink, specify Starting From Version as 5.1 Rev. 0 and Ending At Version as 5.1 Rev. 1, choose the bugs and suggestions that either Remain Open or Were Fixed, then click Execute.

G2 ReThink Version 5.1 Rev. 1

G2 ReThink Version 5.1 Rev. 1 is a compatibility release only. It has no new features or changes. For information on bug fixes, see the Introduction.

G2 ReThink Version 5.1 Rev. 0

ReThink Version 5.1 Rev. 0 has the following new features and changes:

*   G2 ReThink ships as part of the G2 bundle; therefore, the organization of the Start menu and the directory structure of the installation have changed to make ReThink a component of the G2 bundle, rather than vice versa.

    For example, to start G2 ReThink, you choose Start > Programs > Gensym G2 8.3r1 > G2 ReThink > Start G2 ReThink Server.

    Also, the `rethink` directory is a subdirectory of the `g2-8.3r1` directory.

*   ReThink Online blocks are now available with all installations of ReThink. Therefore, the top-level module is now `rethink-online.kb`.

*   Resource Utilization chart shows the percentage utilization of the specified resources.

*   BRMS Task Block allows ReThink users to invoke BRMS rules from a ReThink Model. See the *Business Rules Management System User's Guide* for details.

*   ReThink allows you to reference attributes in subtables, using dot notation in all places where you can specify an attribute, except instruments. For example, `my-subobject.my-subattribute`.

*   ReThink provides charting and reporting capabilities, based on the G2 Reporting Engine (GRPE) module.

*   User interface:

    –   Menus and dialogs for configuring item colors.

    –   Ability to configure X and Y magnification for free text and borderless free text.

    –   Search text box has been renamed Go To.

G2 Optegrity

Describes the new features and changes to G2 Optegrity.

Introduction

This chapter describes G2 Optegrity Version 5.1 Rev. 1, which ships with the G2 Bundle Version 8.3 Rev. 1.

G2 Optegrity is available to licensed users. For further information on G2 Optegrity, visit `http://www.gensym.com/` or contact Gensym at (781) 265-7100.

Information on bug fixes and known bugs is available separately as Technical Bulletins through HelpLink at `www.gensym-support.com`. For information on bugs in G2 Optegrity Version 5.1 Rev. 1, choose Technical Bulletins for Optegrity, specify Starting From Version as 5.1 Rev. 0 and Ending At Version as 5.1 Rev. 1, choose the bugs and suggestions that either Remain Open or Were Fixed, then click Execute.

G2 Optegrity Version 5.1 Rev. 1

G2 Optegrity Version 5.1 Rev. 1 is a compatibility release only. It has no new features or changes. For information on bug fixes, see the Introduction.

G2 Optegrity Version 5.1 Rev. 0

Optegrity Version 5.1 Rev. 0 has these new features and changes:

- G2 Optegrity ships as part of the G2 bundle; therefore, the organization of the Start menu and the directory structure of the installation have changed to make Optegrity a component of the G2 bundle, rather than vice versa.

  For example, to start G2 Optegrity, you choose Start > Programs > Gensym G2 8.3r1 > G2 Optegrity > Start G2 Optegrity Server.

  And the optegrity directory is a subdirectory of the g2-8.3r1 directory.

- The tree view of application objects is now referred to as the Navigator. The Navigator is divided into six main categories:

  - System Models – Provides access to process maps and the components on the process map.

  - Logic – Provides access to GEDP diagrams and SymCure models.

  - Reports – Provides access to reports.

  - Charts – Provides access to charting.

  - Object Models – Displays the object model for the application.

  - System Settings – Contains interface objects, external datapoints, datapoint series, logs, message browsers, unit conversions, user information, system performance, and event and alarm metrics.

- The Project menu reflects these same categories.

- Datapoint trends, which appear in the properties dialog of a datapoint, now use Windows charts instead of the G2 trend-charts. You can also display multiple chart views.

- The Message Browser has these changes:

  - Locked button is now a toggle button to show locked/unlocked state.

  - Filters button is now a toggle button to show when the filter is applied.

- The popup menu for domain objects includes menu choices to show and run the detection, response, or test logic.

- Optegrity provides new charting and reporting capabilities, based on the G2 Reporting Engine (GRPE) module.

- The user interface has these new features and changes:

  - Menus and dialogs for configuring item colors.

  - Ability to configure X and Y magnification for free text and borderless free text.

- – Search text box has been renamed Go To.
- New projects are stored it the `projects` directory of your Optegrity installation directory.

G2 Integrity

Describes the new features and changes to G2 Integrity.

Introduction

This chapter describes G2 Integrity Version 5.0 Rev. 1, which ships with the G2 Bundle Version 8.3 Rev. 1.

G2 Integrity is available to licensed users. For further information on G2 Integrity, visit `http://www.gensym.com/` or contact Gensym at (781) 265-7100.

Information on bug fixes and known bugs is available separately as Technical Bulletins through HelpLink at `www.gensym-support.com`. For information on bugs in G2 Integrity Version 5.0 Rev. 1, choose Technical Bulletins for Integrity, specify Starting From Version as 5.0 Rev. 0 and Ending At Version as 5.0 Rev. 1, choose the bugs and suggestions that either Remain Open or Were Fixed, then click Execute.

G2 Integrity Version 5.0 Rev. 1

G2 Integrity Version 5.0 Rev. 1 is a compatibility release only. It has no new features or changes. For information on bug fixes, see the Introduction.

G2 Integrity Version 5.0 Rev. 0

New Features and Changes

Integrity Version 5.0 Rev. 0 has these new features and changes:

- G2 Integrity ships as part of the G2 bundle; therefore, the organization of the Start menu and the directory structure of the installation have changed to make Integrity a component of the G2 bundle, rather than vice versa.

  For example, to start G2 Integrity, you choose Start > Programs > Gensym G2 8.3r1 > G2 Integrity > Start G2 Integrity Server.

  And the `integrity` directory is a subdirectory of the `g2-8.3r1` directory.

- Integrity uses a Windows user interface provided by Telewindows Next Generation in place of the Integrity 4.0 Java client. This user interface is based on the G2 Run-Time Library and includes menu choices and features that are similar across all Gensym application products.

- Integrity has these new features:

  - Ping – Allows you to ping a single device.

  - Trace route – Displays trace route information for a device.

  - Layer 2 mapping – Provides a display of connected devices and interfaces by using exported information from HP OpenView.

  - Manage devices based on pinging – Allows you to define ping intervals for devices and manage those devices.

  - IP scanner – Allows you to enter an IP range to be pinged and create devices to ping.

  - Web configuration – Allows you to connect to a devices Web server for configuration.

- Several modules no longer exist in the delivered Integrity product; however, the functionality still exists and has been combined into one or more modules. Below is a list of the modules removed and where the functionality now resides:

  - `opacui.kb` –> `opac.kb`

  - `oxs_sim.kb` –> `gsnmp.kb`

  - `asn1.kb` –> `gmib.kb` and `gtrap.kb`

  - `dxi.kb` –> `gdxi.kb`

  - `sockman.kb` –> `gsockman.kb`

  - `opxb.kb` –> `gndo.kb`

- – `dev_util.kb` –> `gndo.kb`

- – `perf_lib.kb` –> no longer required

- Integrity provides new charting and reporting capabilities, based on the G2 Reporting Engine (GRPE) module.

Upgrading to Integrity Version 5.0

To upgrade from Integrity Version 4.0 to Integrity Version 5.0:

1 Load the default Integrity module (`integrity.kb`).

2 Merge in your application.

3 Change the required modules list of your application to include integrity-core.

4 Remove any old Integrity product modules from the required modules list of your application.

5 Delete the modules that are no longer required.

6 Save your application.

Note The generic G2-SNMP Bridge that used to ship with Integrity has been deprecated. The G2 Bundle now ships with the Java-based G2-SNMP Bridge for use in place of the generic bridge. The generic bridge is available, if needed, upon request.

G2 SymCure

Describes the new features and changes to G2 SymCure.

Introduction

This chapter describes G2 SymCure Version 5.1 Rev. 1, which ships with the G2 Bundle Version 8.3 Rev. 1.

G2 SymCure is available as part of the G2 Optegrity and G2 Integrity bundles.

Information on bug fixes and known bugs is available separately as Technical Bulletins through HelpLink at `www.gensym-support.com`. For information on bugs in G2 SymCure Version 5.1 Rev. 1, choose Technical Bulletins for SymCure, specify Starting From Version as 5.1 Rev. 0 and Ending At Version as 5.1 Rev. 1, choose the bugs and suggestions that either Remain Open or Were Fixed, then click Execute.

G2 SymCure Version 5.1 Rev. 1

G2 SymCure Version 5.1 Rev. 1 is a compatibility release only. It has no new features or changes. For information on bug fixes, see the Introduction.

G2 SymCure Version 5.1 Rev. 0

G2 SymCure Version 5.1 Rev. 0 has these new features and changes:

* G2 SymCure is listed as a separate component when installing the G2 bundle.

* Generic events:

    – Icons for generic events indicate whether they have related actions and events by showing a different shade of purple in the center of the icon.

    – User-defined procedure named Occurs At, which is invoked to compute the inferred occurrence time of an event.

    – The dialogs for tuning specific N/M N/M events and updating their associated generic events contain notes to explain the use of Independent of Effects.

* Ability to provide intelligent, time-based text substitutions and priorities when configuring messages by using the $OCCURS and $BECOMES text substitutions in messages. You can also define you own custom time-based substitutions by updating the `cofig.txt` and GFR resources files.

* Fault Modeling Toolbar:

    – Ability to update SymCure configuration parameters without restarting the application by clicking the Load Fault Model Configuration Parameters button.

    – Ability to clear all diagnoses by clicking the Delete All Diagnoses button.

* Diagnostic Console:

    – Ability to show root causes, alarms, tests, and repair actions for individual diagnosis managers.

    – SymCure Alarms and Root Causes Browser have a button called Detailed Explanation.

* Root Cause Episode Manager:

    – Ability to chart root cause duration and frequency distributions, both interactively through the Root Cause Manager, as well as programmatically, using APIs.

    – Ability to save root cause episodes to XML files.

* API:

    – Ability to traverse a fault model in the order of causality to perform a topological sort by using an API. Use the Graph Traversal Procedure/Method on the User-Defined Procedures and Methods palette to configure the procedure to invoke.

– Ability to send notification to event listeners when SymCure detects a chattering event and goes offline.

– Six new APIs that include a user-defined data argument:

- cdg-send-event
- cdg-diagnose-event
- cdg-predict-event
- cdg-send-event-with-post-processing
- cdg-diagnose-event-with-post-processing
- cdg-predict-event-with-post-processing

For details, see the *SymCure User's Guide*.

G2 NeurOn-Line

Describes the new features and changes to G2 NeurOn-Line.

Introduction   **253**

G2 NeurOn-Line Version 5.1 Rev. 1   **253**

G2 NeurOn-Line Version 5.1 Rev. 0   **254**

Introduction

This chapter describes G2 NeurOn-Line Version 5.1 Rev. 1, which ships with the G2 Bundle Version 8.3 Rev. 1.

G2 NeurOn-Line is available to licensed users. For further information on G2 NeurOn-Line, visit `http` or contact Gensym at (781) 265-7100.

Information on bug fixes and known bugs is available separately as Technical Bulletins through HelpLink at `www.gensym-support.com`. For information on bugs in G2 NeurOn-Line Version 5.1 Rev. 1, choose Technical Bulletins for NeurOn-Line, specify Starting From Version as 5.1 Rev. 0 and Ending At Version as 5.1 Rev. 1, choose the bugs and suggestions that either Remain Open or Were Fixed, then click Execute.

G2 NeurOn-Line Version 5.1 Rev. 1

G2 NeurOn-Line Version 5.1 Rev. 1 is a compatibility release only. It has no new features or changes. For information on bug fixes, see the Introduction.

G2 NeurOn-Line Version 5.1 Rev. 0

G2 NeurOn-Line Version 5.1 Rev. 0 has these new features:

- G2 NeurOn-Line ships as part of the G2 bundle; therefore, the organization of the Start menu and the directory structure of the installation have changed to make NeurOn-Line a component of the G2 bundle, rather than vice versa.

  For example, to start G2 NeurOn-Line, you choose Start > Programs > Gensym G2 8.3r1 > G2 NeurOn-Line > Start G2 NeurOn-Line Server.

  And the `nol` directory is a subdirectory of the `g2-8.3r1` directory.

- Statistical tools in G2 for online productivity analysis:

  - Principle Component Analysis (PCA) models can be developed in both NOL Studio and G2.

  - G2 provides a powerful visualization environment for process monitoring.

  - Informational charts for PCA models.

  - Monitoring charts for process monitoring.

  - Additional information for online model training.

  - Model parameter exchange in XML format.

- Additional Information for GNNE Online Model Training:

  - A status flag that indicates the training is in progress.

  - A status flag that indicates that an error has occurred during training.

  - More informative error messages for errors during the training process.

- NOL Studio models can be export into XML format:

  - The model types include Predictive Model, BPN, AAN, RBFN.

  - XML file can be imported into GNNE predictive model.

- Predictive model in GNNE:

  - Contains input and output variable information.

  - Contains input delay information.

  - Can be loaded and exported with XML format.

  - Model execution is inside G2.

  - Can be retrained online.

  - Predictive Model in Nol Studio and the Predictive Model in GNNE can exchange parameters online with Nol Studio connected to G2.

A

Acrobat Reader, recommended version for   27
AIX
    component availability for   36
    support for   151
    supported version   14
API
    See system procedures
attributes
    date and time formats for   91
attributes, generating a list of system-
  defined   131
authorizing G2 bundle components   15

B

bridges
    See G2 bridges
bug fixes
    G2
        Version 8.3 Rev. 0   45
        Version 8.3 Rev. 1   43
    G2 ActiveXLink   185
    G2 CORBALink   189
    G2 Database Bridges   191
    G2 Developer's Utilities   167
    G2 HLA   205
    G2 Java Mail Bridge   209
    G2 Java Socket Manager   213
    G2 JavaLink   207
    G2 JMSLink   217
    G2-PI Bridge   221
    G2 utilities   181
    G2 WebLink   225
    G2-OPC Client Bridge   219
    G2-SNMP Bridge   223
    Telewindows
        Version 8.3 Rev. 0   157
        Version 8.3 Rev. 1   155
Business Process Management System (BPMS)
    Version 2.3 Rev. 0   176

Business Rules Management System (BRMS)
    Version 2.3 Rev. 0   176
bytes, reading and writing from and to a
  file   138

C

calendar control
    Version 8.3 Rev. 0   53
change log
    enhancements   93
    system procedures   94
    viewing   93
chart views
    Version 8.3 Rev. 0   73
color-picker control
    Version 8.3 Rev. 0   53
combo-box control
    Version 8.3 Rev. 0   54
computational capabilities
    confirming run state changes   124
    date and time formats for attributes   91
    error handling   124
    locking mechanism for objects   109
    referencing a time interval ending with the
      collection time   123
    referring to the current system time   124
    summing values in histories   122
    Version 8.3 Rev. 0   91
    version control enhancements   93
connections
    interactively changing vertices   90
context-sensitive help capability, support for
  through GOLD   34
controls, dialog
    Version 8.3 Rev. 0   53
    Version 8.3 Rev. 1   44
custom Windows dialogs
    See dialog views
customer support services   xxiii

HP-UX
    component availability for 36
    Itanium
        G2 Gateway 202
        Version 8.3 Rev. 0 151
        Version 8.3 Rev. 1 44
    supported version 14
HTML
    G2 online documentation 26
HTML views
    Version 8.3 Rev. 0 78
HTTP
    See Also G2 WebLink
    Web client 143
    Web server 144

I

image control
    Version 8.3 Rev. 0 58
Inspect commands
    version control 108
installation instructions 15
integration, Web
    HTTP 143
    network connection management 147
    SOAP 142
    Web services 139
IP address, getting for G2 server 129
Itanium
    G2 Gateway 202
    support for 151
item configurations, upgrading 17

J

Java
    See G2 JavaLink, G2 Java Mail Bridge, G2
      JMSLink, and G2 Java Socket Manager
jmail-disconnect-without-kill 211
jmail-ping 210
`jsockman.kb` 214

K

KBs
    loading when upgrading 16
    recompiling 16
    text-stripped
        See text-stripped KBs
    upgrading
        general 15
        standard user interface 17
known bugs
    G2
        Version 8.3 Rev. 0 45
        Version 8.3 Rev. 1 43
    G2 ActiveXLink 185
    G2 CORBALink 189
    G2 Database Bridges 191
    G2 Developer's Utilities 167
    G2 Diagnostic Assistant (GDA) 229
    G2 e-SCOR 237
    G2 Gateway 199
    G2 HLA 205
    G2 Integrity 245
    G2 Java Mail Bridge 209
    G2 Java Socket Manager 213
    G2 JavaLink 207
    G2 JMSLink 217
    G2 NeurOn-Line 253
    G2 OPCLink 219
    G2 Optegrity 241
    G2 ReThink 239
    G2 SymCure 249
    G2 utilities 181
    G2 WebLink 225
    G2-PI Bridge 221
    G2-SNMP Bridge 223
    Telewindows
        Version 8.3 Rev. 0 157
        Version 8.3 Rev. 1 155

L

language, window specific 23
license key 15

R

S

T

Windows user interface   *(continued)*
    tree view
        Version 8.3 Rev. 0   86
workspace view control
    NMS popup menus with   71
workspaces
    visible grid on   89
World Wide Web (WWW)
    See G2 WebLink

X

XML
    integrating with Web services, using   139